

Generator maintenance scheduling based on the expected capability of satisfying energy demand

J Eygelaar



UNIVERSITEIT
iYUNIVESITHI
STELLENBOSCH
UNIVERSITY

100
1918 - 2018

Dissertation presented for the degree of
Doctor of Philosophy
in the Faculty of Engineering at Stellenbosch University

Declaration

By submitting this dissertation electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: March 1, 2018

Abstract

The integrity of an electric power system is significantly threatened by unexpected downtimes of *power generating units* (PGUs). In order to minimise the occurrence of such unexpected PGU downtimes, planned preventative maintenance is routinely performed on the PGUs of the system. The effective scheduling of these planned maintenance PGU outages is a considerable challenge for any power utility.

The celebrated *generator maintenance scheduling* (GMS) problem involves finding a set of planned preventative maintenance outages of PGUs in a power system. A feasible solution to this problem is typically a list of dates indicating the commencement of planned maintenance for each PGU in the power system. Solutions to the GMS problem are typically subjected to a wide variety of power system constraints and the problem is considered to be a hard combinatorial optimisation problem.

Two novel GMS criteria are introduced in this dissertation. The first criterion involves minimisation of the probability that any PGU in the system will fail during a scheduling window of pre-specified length. This scheduling criterion is weighted according to the rated capacity of each PGU so as to give some preference, in terms of maintenance commencement times, to PGUs that contribute considerably to the overall system capacity. This criterion draws from basic notions in reliability theory which may be used to estimate the failure probability of a system. The second criterion involves maximisation of the expected energy produced during the scheduling window. In this case, PGU failures are modelled by random variables. Two mixed integer programming models are formulated for the GMS problem with the newly proposed scheduling criteria as objective functions. One of these models is linear and the other one is nonlinear. The nonlinear model is linearised by piecewise linear approximation in order to be able to solve it exactly. Both models incorporate a number of constraints, including energy demand satisfaction constraints, earliest and latest maintenance window constraints, maintenance resource constraints and maintenance exclusion constraints.

Two GMS test problems are modelled in this fashion. The resulting four GMS model instances are each solved by two different solution approaches — exactly and approximately (by employing a metaheuristic). The exact solution approach involves use of IBM ILOG's well-known optimisation suite CPLEX which employs a branch-and-cut method, while the metaheuristic of *simulated annealing* is implemented in the programming language R as approximate model solution methodology. After computing optimal solutions for the four GMS model instances mentioned above, a sensitivity analysis is performed in order to determine the feasibility of an exact solution approach in respect of small to medium-sized GMS problem instances. An extensive parameter optimisation experiment is also conducted in order to obtain a suitable set of simulated annealing parameter values for use in the context of the four GMS model instances. The GMS solutions obtained when incorporating these suitable parameter values within the simulated annealing algorithm for the four GMS model instances are compared to the corresponding exact solutions. The approximate solution methodology is found to be a viable alternative solution approach to

the exact solution approach, capable of obtaining solutions within 3% of optimality for all four GMS model instances — often requiring considerably shorter computation times.

The efficacy of the two proposed scheduling criteria are also analysed in terms of a real-world case study based on the power grid of the national power utility in South Africa. The 157-unit Eskom test problem contains a large number of PGUs, some of which require maintenance multiple times during the scheduling window. The aforementioned approximate solution approach is adopted to solve this large problem instance with respect to both proposed scheduling criteria and is found to be a viable approach from a practical point of view.

A computerised *decision support system* (DSS) is also proposed which is aimed at facilitating effective GMS decision making with respect to the two proposed scheduling criteria. The DSS is implemented in the programming language *Shiny*, which is an R package for creating user-friendly interfaces. The DSS is equipped with an intuitive graphical user interface and the system is able to solve user-provided problem instances of the GMS problem.

Uittreksel

Die integriteit van 'n elektriese kragstelsel word beduidend deur onverwagte onderbrekings in die werking van die stelsel se *kragopwekkingseenhede* (KOEe) bedreig. Beplande voorkomende onderhoud word roetine-gewys op die KOEe van kragstelsels uitgevoer om voorkomste van sulke onverwagte onderbrekings te minimeer. Die doeltreffende skedulering van KOE diensonderbrekings vir voorkomende onderhoud is 'n aansienlike uitdaging vir enige kragvoorsiener.

Die gevierde *opwekker-onderhoudskeduleringsprobleem* (OOS-probleem) behels die soeke na 'n versameling beplande onderhoudsonderbrekings vir die KOEe in 'n kragstelsel. 'n Toelaatbare oplossing vir hierdie probleem neem tipies die vorm aan van 'n lys datums wat begintye vir die beplande onderhoud van elke KOE in die stelsel aandui. Oplossings van die OOS-probleem word tipies aan 'n wye verskeidenheid kragstelselbeperkings onderwerp en die probleem word as 'n moeilike kombinatoriese optimeringsprobleem beskou.

Twee nuwe OOS-kriteria word in hierdie proefskrif daargestel. Die eerste kriterium behels minimering van die waarskynlikheid dat enige KOE in die stelsel gedurende 'n vooraf-gespesifiseerde skeduleringsvenster faal. Hierdie skeduleringskriterium word ook volgens die kapasiteitstempo van elke KOE geweeg om sodoende in terme van onderhoudbegintye voorkeur te gee aan KOEe wat noemenswaardige kapasiteit tot die stelsel bydra. Hierdie kriterium bou op basiese konsepte in betroubaarheidsteorie wat gebruik kan word om die falingswaarskynlikheid van 'n stelsel af te skat. Die tweede kriterium behels maksimering van die verwagte hoeveelheid energie wat gedurende die skeduleringsvenster opgewek sal word. In hierdie geval word KOE falings deur kansveranderlikes gemodelleer. Twee gemengde heeltallige programmeringsmodelle word vir die OOS-probleem geformuleer, met die nuut-voorgestelde skeduleringskriteria as doelfunksies. Een van hierdie modelle is lineêr en die ander een is nie-lineêr. Die nie-lineêre model word deur middel van stuksgewys-lineêre benaderings gelineariseer sodat dit eksak opgelos kan word. Beide modelle sluit 'n aantal beperkings in, naamlik energie-vraagbeperkings, vroegste en laatste onderhoudvenster-beperkings, skeduleringshulpbron-beperkings en onderhouduitsluitingsbeperkings.

Twee OOS-toetsprobleme word op hierdie wyse gemodelleer. Die gevolglike vier OOS-modelgevalle word elk op twee verskillende maniere opgelos — eksak en benaderd (deur middel van 'n metaheuristiek). Die eksakte oplossingsbenadering behels die gebruik van IBM ILOG se bekende optimeringsuite CPLEX wat 'n vertak-en-snit metode toepas, terwyl die metaheuristiek *gesimuleerde tempering* as benaderde oplossingsmetodologie in die programmeringstaal R geïmplementeer word. Nadat optimale oplossings vir al vier OOS-probleemgevalle bereken word, word 'n sensitiviteitsanalise uitgevoer om die haalbaarheid van die eksakte metode in die konteks van klein en medium OOS-probleemgevalle te toets. 'n Uitgebreide parameter-optimeringseksperiment word ook uitgevoer om 'n sinvolle versameling parameterwaardes vir die gesimuleerde temperingsalgoritme in die konteks van die bogenoemde vier probleemgevalle te bepaal. Die OOS-oplossings wat vir die vier probleemgevalle deur gebruikmaking van hierdie versameling parameterwaardes in die gesimuleerde temperingsalgoritme gevind word, word met die ooreenstem-

mende eksakte oplossings vergelyk. Daar word bevind dat die benaderde oplossingsbenadering 'n haalbare alternatief is tot die eksakte benadering, wat oplossings binne 3% van optimaliteit vir al vier probleemgevalle kan vind — dikwels ook binne aansienlik korter berekeningstye.

Die gepastheid van die twee voorgestelde skeduleringskriteria word ook in die konteks van 'n realistiese gevallestudie ondersoek, wat gebaseer is op die kragnetwerk van die Suid-Afrikaanse kragvoorsiener, Eskom. Die 157-eenheid Eskom toetsprobleem bevat 'n groot getal KOEe, sommige waarvan veelvuldige onderhoudsonderbrekings gedurende die skeduleringsvenster vereis. Die bogenoemde benaderde oplossingsbenadering word, onderworpe aan beide skeduleringskriteria, op hierdie groot toetsprobleem toegepas, en daar word bevind dat die oplossingsbenadering prakties haalbaar is.

'n Gerekenariseerde *besluitsteunstelsel* (BSS) word ook daargestel wat daarop gemik is om doeltreffende OOS-besluitneming met betrekking tot beide voorgestelde skeduleringskriteria te fasiliteer. Die BSS word in die programmeringstaal Shiny geïmplementeer. Shiny is 'n R-pakket waarmee gebruikersvriendelike koppelvlakke daargestel kan word. Die BSS word van 'n gebruikersvriendelike koppelvlak voorsien en die stelsel is daartoe instaat om gebruikers-gespesifiseerde gevalle van die OOS-probleem op te los.

Acknowledgements

The author wishes to acknowledge the following people and institutions for their various contributions towards the completion of this work:

- My promoter, Prof JH van Vuuren, for the guidance and support throughout the duration of my research, not only on a professional level, but also on a more personal level. He assisted in developing me as a young researcher as well as a person. I am eternally grateful for his friendship, support and guidance throughout the completion of this dissertation.
- Dr BG Lindner for introducing me to the field of study in this dissertation and assisting me in overcoming some technical difficulties during the early stages of programming in a new programming language. The quality of his work and technical ability is not only an inspiration to me, but also to the other members of SUnORE.
- My fellow SUnORE colleagues for their friendship, support and endless technical assistance over the past three years during the completion of this research. The memories, laughter and other experiences will be cherished for many years to come.
- Other friends and family for their encouragement and moral support during the completion of this dissertation. Their understanding during times of unavailability and great pressure, especially toward the end of this work, is much appreciated.
- The Department of Industrial Engineering at Stellenbosch University, and SUnORE in particular, for the use of its great office space and computing facilities.
- Finally, SUnORE for the financial support that made this research possible.

Table of Contents

Abstract	iii
Uittreksel	v
Acknowledgements	vii
List of Reserved Symbols	xv
List of Acronyms	xvii
List of Figures	xix
List of Tables	xxiii
List of Algorithms	xxvii
1 Introduction	1
1.1 Background	1
1.2 Informal problem description	4
1.3 Dissertation scope and objectives	5
1.4 Dissertation organisation	6
I Literature review	11
2 Generator maintenance scheduling	13
2.1 Model considerations	13
2.1.1 The scheduling window	14
2.1.2 The scheduling resolution	14
2.1.3 The objective function	14
2.1.4 The model constraints	15

2.1.5	Related energy problems	16
2.2	GMS model formulations in literature	17
2.2.1	Objective function formulation	18
2.2.2	Constraint formulation	22
2.3	GMS model solution approaches	26
2.3.1	Mathematical programming techniques	26
2.3.2	Expert systems	30
2.3.3	Fuzzy logic approaches	31
2.3.4	Heuristics	32
2.3.5	Metaheuristics	33
2.3.6	Recent developments	35
2.4	The method of simulated annealing	36
2.5	Chapter summary	39
3	Reliability theory	41
3.1	General considerations	41
3.2	Basic mathematical notions	42
3.3	Lifetime distribution models for non-repairable systems	44
3.3.1	The exponential model	44
3.3.2	The Weibull model	46
3.3.3	The normal model	47
3.3.4	The lognormal model	48
3.3.5	The gamma model	49
3.4	Repairable systems	50
3.4.1	The Homogeneous poisson process	50
3.4.2	The Non-homogeneous poisson process following an exponential law	51
3.4.3	The Non-homogeneous poisson process following a power law	52
3.5	Life data classification	53
3.5.1	Complete data	53
3.5.2	Right-censored data	54
3.5.3	Interval-censored data	54
3.5.4	Left-censored data	54
3.6	Trend tests	55
3.6.1	The reverse arrangement test	55
3.6.2	The military handbook test	56
3.6.3	The Laplace test	57

3.7	Model parameter estimation methods	58
3.7.1	The maximum likelihood method	58
3.7.2	The least squares method	59
3.7.3	The Bayesian parameter estimation method	59
3.8	Acceleration models	60
3.8.1	The Arrhenius model	61
3.8.2	The Eyring model	61
3.9	Chapter summary	62
II Mathematical modelling framework		63
4	Mathematical model formulations	65
4.1	The GMS objective functions selected	65
4.1.1	Minimising probability of unit failure	65
4.1.2	Maximising expected energy production	66
4.2	Model assumptions	67
4.3	GMS models	69
4.3.1	The model variables	70
4.3.2	The objective functions	70
4.3.3	The model constraints	76
4.4	Chapter summary	77
5	Model solution approaches	79
5.1	Linearisation of the model of §4.3	79
5.1.1	Piecewise linear function approximation	79
5.1.2	Breakpoint selection during piecewise linear function approximation . . .	81
5.2	Exact model solution approach	83
5.2.1	Motivation for the choice of optimisation platform	83
5.2.2	CPLEX implementation	84
5.3	Approximate model solution approach	86
5.3.1	Motivation for the selected approximate solution methodology	89
5.3.2	Simulated annealing implementation	89
5.3.3	Experimental design	95
5.4	Chapter summary	97

III Academic benchmark results	99
6 Academic benchmark system data	101
6.1 The 21-unit system	101
6.2 The 32-unit IEEE-RTS	103
6.3 Chapter summary	109
7 Minimising probability of unit failure	111
7.1 Exact solution results	111
7.1.1 The 21-unit system	111
7.1.2 The IEEE-RTS	118
7.2 Approximate solution results	124
7.2.1 The 21-unit system	125
7.2.2 The IEEE-RTS	138
7.3 Chapter summary	153
8 Maximising expected energy production	157
8.1 Piecewise linear approximation results	157
8.1.1 The 21-unit system	157
8.1.2 The IEEE-RTS system	167
8.2 Metaheuristic approximate solution approach	177
8.2.1 The 21-unit system	177
8.2.2 The IEEE-RTS	190
8.3 Chapter summary	204
IV Real-world model application	207
9 Case study data	209
9.1 Background	209
9.2 Specifications	209
9.3 Objective function extensions	215
9.3.1 Minimising of probability of unit failure	215
9.3.2 Maximising expected energy production	220
9.4 Chapter summary	224
10 Numerical results	227
10.1 Parameter optimisation experiments	227

Table of Contents	xiii
10.1.1 Minimising the probability of unit failure	227
10.1.2 Maximising expected energy production	228
10.2 Approximate solutions	229
10.2.1 Minimising the probability of unit failure	229
10.2.2 Maximising expected energy production	234
10.3 Chapter summary	239
11 Decision support framework	241
11.1 General considerations in decision support systems	241
11.1.1 The database	241
11.1.2 The graphical user interface	242
11.1.3 The model base	242
11.2 Detailed process description	242
11.3 System development	243
11.3.1 Data preparation	243
11.3.2 System walk-through	244
11.4 Chapter summary	258
V Conclusion	259
12 Dissertation summary	261
12.1 Dissertation contents	261
12.2 Appraisal of dissertation contributions	264
13 Possible future work	269
13.1 Incorporating PGU generation capability variation	269
13.2 Enforcing a maintenance interval constraint	270
13.3 Failure frequency generalisation	270
13.4 Incorporating PGUs with increasing failure rates	270
13.5 The possibility of failures being dependent on one another	271
13.6 Reliability accuracy improvement in the linear GMS model	271
13.7 Adopting constraint programming as solution approach	271
13.8 Adopting a multi-objective GMS paradigm	272
References	273

A Eskom case study: Parameter optimisation experimental results	287
A.1 Minimising the probability of unit failure	287
A.1.1 Initial acceptance ratio and soft constraint violation severity factor	287
A.1.2 Cooling parameter, reheating parameter and epoch parameter	288
A.2 Maximising expected energy production	289
A.2.1 Initial acceptance ratio and soft constraint violation severity factor	289
A.2.2 Cooling parameter, reheating parameter and epoch parameter	290
B Contents of the accompanying disc	293

List of Reserved Symbols

GMS model

Symbol	Meaning
a	Function value of a piecewise linear break point
b	Piecewise linear break point
C_u	Rated capacity of PGU u
C_{\max}	Rated capacity of largest PGU in the system
d_u	Duration of planned maintenance to be performed on PGU u
E	Expected energy produced
e_u	Earliest maintenance commencement date for PGU u
F_u	Number of resources required for the maintenance of PGU u
$f_{u,p,v}$	Resource requirement parameter for maintenance during time period p if planned maintenance were to be scheduled to start during time period v on PGU u
I	Maximum number of power generating units that may be scheduled for simultaneous maintenance
\mathcal{J}	Set of PGUs which may not be in maintenance simultaneously
k_u	Time period during which maintenance of PGU u is scheduled to commence
ℓ_u	Latest maintenance commencement date of PGU u
M_p	Number of available resources during planning period p
m	Number of planning periods within the scheduling window
n	Number of power generating units in the system
\mathcal{P}	Set of planning periods in the scheduling window
s	Number of soft constraints
\mathcal{U}	Set of power generating units in the system
w_u	Number of maintenance intervals of PGU u within a given scheduling window
X_u	Random variable denoting the timing of the next failure of PGU u
x'_u	The time period during which PGU u entered into operation after its previous maintenance operation
$x_{u,p}$	GMS binary decision variable specifying the commencement date of maintenance of PGU u during time period p
$y_{u,p}$	GMS binary auxiliary variable specifying that PGU u is in maintenance during time period p
ϵ	Earliest starting time of the maintenance interval of a PGU
λ_u	Failure rate of PGU u
τ	Scheduling window

Indices

Symbol	Meaning
p	Planning period index
u	Power generating unit index
v	Scheduled planned maintenance commencement starting time

Simulated annealing

Symbol	Meaning
A_{min}	Minimum number of move acceptances
$\overline{\Delta E}$	Average deterioration in objective function values
G	Limiting values for the soft constraints
L	Termination length of an epoch
T_0	Initial temperature
α	Cooling parameter
γ	Soft constraint severity factor
ξ	Reheating parameter
ϕ	Multiplicative penalty function
χ_0	Initial acceptance ratio
ψ	Epoch management parameter

List of Acronyms

Acronym	Meaning
ACO	Ant Colony Optimisation
B&B	Branch-and-Bound
CDF	Cumulative Distribution Function
CFR	Constant Failure Rate
DFR	Decreasing Failure Rate
DPSA	Dynamic Programming Successive Approximation
DSS	Decision Support System
EAF	Energy Availability Factor
ED	Economic Dispatch
EFS	Energy Flow Simulator
ES	Expert System
EUE	Expected Unsupplied Energy
FOM	Force Of Mortality
GA	Genetic Algorithm
GENCO	GENERation COmpany
GMS	Generator Maintenance Scheduling
GUI	Graphical User Interface
HCI	Human-Computer Interaction
HPP	Homogeneous Poisson Process
IEEE	Institute of Electrical and Electronics Engineers
IFR	Increasing Failure Rate
IP	Integer Programming
IRP	Integrated Resource Plan
ISO	Independent System Operator
LOLE	Loss Of Load Expectation
LOLP	Loss Of Load Probability
LP	Linear Program
MHT	Military Handbook Test
MILP	Mixed Integer Linear Programming
MLE	Maximum Likelihood Estimation
MTBF	Mean Time Between Failures
MTTF	Mean Time To Failure
NHPP	Non-Homogeneous Poisson Process
OCFLF	Other Capability Loss Factors
PCLF	Planned Capability Loss Factors
PDF	Probability Density Function
PGU	Power Generating Unit

Acronym	Meaning
PSO	Particle Swarm Optimisation
RAT	Reverse Arrangement Test
RT	Reliability Theory
RTS	Reliability Test System
SA	Simulated Annealing
TS	Tabu Search
UC	Unit Commitment
UCLF	Unplanned Capability Loss Factors

List of Figures

1.1	Energy production and consumption in South Africa	2
2.1	Benders' decomposition flow diagram	29
3.1	The well-known bathtub curve	42
3.2	A graphical representation of complete data	53
3.3	A graphical representation of right censored data	54
3.4	A graphical representation of interval censored data	55
3.5	A graphical representation of left censored data	55
3.6	A graphical representation of Q_{end}	57
3.7	<i>Military handbook test</i> outcomes	57
3.8	<i>Laplace trend test</i> outcomes	58
4.1	A graphical representation of variables and parameters	70
4.2	The reliability of a PGU	71
4.3	The expected energy produced in three different cases	75
5.1	A piecewise linear function	80
5.2	Parameters and variables in CPLEX	85
5.3	Parameter inputs required for implementation of the GMS models	86
5.4	Objective function and constraint implementations in CPLEX	87
5.5	An alternative objective function implementation in CPLEX	88
5.6	Ejection chain neighbourhood move operator example	93
6.1	The true value and piecewise linear approximation (21-unit system)	104
6.2	IEEE-RTS demand data	106
6.3	The true value and piecewise linear approximation (IEEE-RTS)	108
7.1	An optimal maintenance schedule for the 21-unit system (linear)	112
7.2	The manpower required by and system capacity of the 21-unit system (optimal)	113

7.3	Probability of failure for each PGU of the 21-unit system	114
7.4	Comparison between two maintenance schedules for the 21-unit system (linear) .	116
7.5	An optimal maintenance schedule for the IEEE-RTS (linear)	119
7.6	The manpower required by and system capacity of the IEEE-RTS (optimal) . . .	120
7.7	Comparison between two maintenance schedules for the IEEE-RTS (linear) . . .	122
7.8	The manpower required by and system capacity of the IEEE-RTS (linear, SSR) .	123
7.9	Linear model optimality gap box plots for the 21-unit system (Phase 1)	127
7.10	Linear model computation time for the 21-unit system (Phase 1)	128
7.11	The 21-unit system infeasible incumbents of the linear model (Phase 1)	129
7.12	Linear model optimality gap box plots for the 21-unit system (Phase 2)	133
7.13	Linear model computation time for the 21-unit system (Phase 2)	134
7.14	The 21-unit system infeasible incumbents of the linear model (Phase 2)	134
7.15	The incumbent returned by the SA algorithm for the 21-unit system (linear) . .	136
7.16	The manpower required by and system capacity of the 21-unit system (incumbent)	137
7.17	Maintenance schedule comparison for the 21-unit system (linear)	138
7.18	The manpower required by and system capacity of the 21-unit system (linear) . .	139
7.19	Linear model optimality gap box plots for the IEEE-RTS (Phase 1)	141
7.20	Linear model computation time for the IEEE-RTS (Phase 1)	142
7.21	The IEEE-RTS infeasible incumbents of the linear model (Phase 1)	144
7.22	Linear model optimality gap box plots for the IEEE-RTS (Phase 2)	147
7.23	Linear model computation time for the IEEE-RTS (Phase 2)	148
7.24	The IEEE-RTS infeasible incumbents of the linear model (Phase 2)	148
7.25	The incumbent returned by the SA algorithm for the IEEE-RTS (linear)	151
7.26	The manpower required by and system capacity of the IEEE-RTS (incumbent) .	152
7.27	Maintenance schedule comparison for the IEEE-RTS (exact)	153
7.28	The manpower required by and system capacity of the IEEE-RTS (exact)	154
8.1	An optimal maintenance schedule for the 21-unit system (nonlinear)	158
8.2	The manpower required by and system capacity of the 21-unit system (nonlinear)	159
8.3	Expected energy production for each PGU of the 21-unit system	160
8.4	Comparison between two maintenance schedules for the 21-unit system (SSR) . .	162
8.5	Comparison between two maintenance schedules for the 21-unit system (probability)	164
8.6	An optimal maintenance schedule for the IEEE-RTS (nonlinear)	168
8.7	The manpower required by and system capacity of the IEEE-RTS (nonlinear) . .	169
8.8	Comparison between two maintenance schedules for the IEEE-RTS (SSR)	170
8.9	The manpower required by and system capacity of the IEEE-RTS solution (SSR)	171

8.10	Comparison between two maintenance schedules for the IEEE-RTS (probability)	173
8.11	The manpower required by and system capacity of the IEEE-RTS (probability)	174
8.12	Nonlinear model optimality gap box plots for the 21-unit system (Phase 1)	179
8.13	Nonlinear model computation time for the 21-unit system (Phase 1)	180
8.14	The 21-unit system infeasible incumbents of the nonlinear model (Phase 1)	181
8.15	Nonlinear model optimality gap box plots for the 21-unit system (Phase 2)	184
8.16	Nonlinear model computation time for the 21-unit system (Phase 2)	186
8.17	The 21-unit system infeasible incumbents of the nonlinear model (Phase 2)	186
8.18	The incumbent returned by the SA algorithm for the 21-unit system (nonlinear)	188
8.19	The manpower required by and system capacity of the 21-unit system (incumbent)	189
8.20	Maintenance schedule comparison for the 21-unit system (nonlinear)	190
8.21	The manpower required by and system capacity of the 21-unit system (comparison)	191
8.22	Nonlinear model optimality gap box plots for the IEEE-RTS (Phase 1)	193
8.23	Nonlinear model computation time for the IEEE-RTS (Phase 1)	194
8.24	The IEEE-RTS infeasible incumbents of the nonlinear model (Phase 1)	195
8.25	Nonlinear model optimality gap box plots for the IEEE-RTS (Phase 2)	198
8.26	Nonlinear model computation time for the IEEE-RTS (Phase 2)	200
8.27	The IEEE-RTS infeasible incumbents of the nonlinear model (Phase 2)	201
8.28	The incumbent returned by the SA algorithm for the IEEE-RTS (nonlinear)	203
8.29	The manpower required by and system capacity of the IEEE-RTS (incumbent)	204
8.30	Maintenance schedule comparison for the IEEE-RTS (nonlinear)	205
8.31	The manpower required by and system capacity of the IEEE-RTS (comparison)	206
9.1	157-unit Eskom case study demand data	213
9.2	The reliability of a PGU requiring multiple maintenance procedures	221
9.3	Failure in Case I	222
9.4	Failure in Case II	223
9.5	Failure in Case III	223
9.6	Expected energy produced by PGU 105	225
10.1	Incumbent returned by the linear SA model for the case study (capacity)	231
10.2	Incumbent returned by the linear SA model for the case study (failure rate)	232
10.3	System capacity for the 157-unit Eskom case study for the linear model	233
10.4	Incumbent returned by the nonlinear SA model for the case study (capacity)	236
10.5	Incumbent returned by the nonlinear SA model for the case study (failure rate)	237
10.6	System capacity for the 157-unit Eskom case study for the nonlinear model	238

11.1 Graphical DDS overview	243
11.2 Demand specification format	244
11.3 PGU specifications format	245
11.4 Home scree of the DSS	246
11.5 File input GUI	247
11.6 Summary of system tab (eight percent safety margin)	248
11.7 Summary of system tab (fifteen percent safety margin)	249
11.8 PGU specifications tab	250
11.9 Demand specifications tab	251
11.10 Algorithm specifications window	252
11.11 Algorithm overview tab	253
11.12 Maintenance schedule tab	254
11.13 Maintenance starting times	255
11.14 Available capacity tab	256
11.15 Required manpower tab	257

List of Tables

1.1	Plant mix of Eskom	2
3.1	Reverse arrangement test critical values	56
5.1	Triangular matrix containing the SSR	82
5.2	Parameter values of the parameter experiment for the 21-unit system (Phase 1)	95
5.3	Parameter values of the parameter experiment for the 21-unit system (Phase 2)	95
6.1	The 21-unit test system specifications	102
6.2	Time elapsed since previous PGU maintenance for the 21-unit system	102
6.3	Piecewise linearisation breakpoints for the 21-unit system	103
6.4	The IEEE-RTS specifications	105
6.5	Exclusion sets for IEEE-RTS	105
6.6	The IEEE-RTS demand per week	106
6.7	Time elapsed since previous PGU maintenance for the IEEE-RTS	107
6.8	Piecewise linearisation breakpoints for the IEEE-RTS	107
7.1	Objective function value comparison for the 21-unit system (linear)	115
7.2	Statistics pertaining to the optimal solutions of the 21-unit system (linear)	117
7.3	Groups of PGUs for the IEEE-RTS with similar specifications	121
7.4	Objective function value comparison for the IEEE-RTS (linear)	122
7.5	Statistics pertaining to the optimal solutions of the IEEE-RTS (linear)	124
7.6	Linear model optimality gaps for the 21-unit system (Phase 1)	125
7.7	Linear model computation times for the 21-unit system (Phase 1)	126
7.8	The 21-unit system infeasible incumbents of the linear model (Phase 1)	126
7.9	The 21-unit system final parameter values of linear model (Phase 1)	130
7.10	Linear model optimality gaps for the 21-unit system (Phase 2)	130
7.11	Linear model computation times for the 21-unit system (Phase 2)	131
7.12	The 21-unit system infeasible incumbents of the linear model (Phase 2)	131

7.13	The parameters finally selected for the 21-unit system (linear)	135
7.14	Linear model optimality gaps for the IEEE-RTS (Phase 1)	140
7.15	Linear model computation times for the IEEE-RTS (Phase 1)	140
7.16	The IEEE-RTS infeasible incumbents of the linear model (Phase 1)	140
7.17	The IEEE-RTS final parameter values of the linear model (Phase 1)	143
7.18	Linear model optimality gaps for the IEEE-RTS (Phase 2)	145
7.19	Linear model computation times for the IEEE-RTS (Phase 2)	145
7.20	The IEEE-RTS infeasible incumbents of the linear model (Phase 2)	146
7.21	The parameters finally selected for the IEEE-RTS (linear)	150
8.1	Objective function value comparison for the 21-unit system (SSR)	161
8.2	Objective function value comparison for the 21-unit system (probability)	163
8.3	Statistics pertaining to the optimal solutions of the 21-unit system (nonlinear)	166
8.4	Objective function value comparison for the IEEE-RTS (SSR)	172
8.5	Objective function value comparison for the IEEE-RTS (probability)	175
8.6	Statistics pertaining to the optimal solutions of the IEEE-RTS (nonlinear)	176
8.7	Nonlinear model optimality gaps for the 21-unit system (Phase 1)	178
8.8	Nonlinear model computation times for the 21-unit system (Phase 1)	178
8.9	The 21-unit system infeasible incumbents of the nonlinear model (Phase 1)	178
8.10	The 21-unit system final parameter values of the nonlinear model (Phase 1)	182
8.11	Nonlinear model optimality gaps for the 21-unit system (Phase 2)	183
8.12	Nonlinear model computation times for the 21-unit system (Phase 2)	183
8.13	The 21-unit system infeasible incumbents of the nonlinear model (Phase 2)	183
8.14	The parameters finally selected for the 21-unit system (nonlinear)	187
8.15	Nonlinear model optimality gaps for the IEEE-RTS (Phase 1)	192
8.16	Nonlinear model computation times for the IEEE-RTS (Phase 1)	192
8.17	The IEEE-RTS infeasible incumbents of the nonlinear model (Phase 1)	192
8.18	The IEEE-RTS final parameter values of the nonlinear model (Phase 1)	196
8.19	Nonlinear model optimality gaps for the IEEE-RTS (Phase 2)	197
8.20	Nonlinear model computation times for the IEEE-RTS (Phase 2)	197
8.21	The IEEE-RTS infeasible incumbents of the nonlinear model (Phase 2)	197
8.22	The parameters finally selected for the IEEE-RTS (nonlinear)	201
9.1	Description of the PGUs in the Eskom case study	210
9.2	The actual PGU specifications for the 157-unit Eskom case study	210
9.3	Demand required per day for the 157-unit Eskom case study	214

9.4	The 157-unit Eskom case study specifications	216
10.1	Parameters finally selected for the 157-unit Eskom case study (linear model) . . .	228
10.2	Parameters finally selected for the 157-unit Eskom case study (nonlinear model)	229
10.3	Decision variable values for the 157-unit Eskom case study (linear model)	230
10.4	Decision variable values for the 157-unit Eskom case study (nonlinear model) . .	235
A.1	Objective function values in the parameter experiment (Phase 1, linear model) .	287
A.2	Computation times of the parameter experiment (Phase 1, linear model)	288
A.3	The infeasible incumbents of the parameter experiment (Phase 1, linear model) .	288
A.4	Objective function values in the parameter experiment (Phase 2, linear model) .	288
A.5	Computation times of the parameter experiment (Phase 2, linear model)	289
A.6	The infeasible incumbents of the parameter experiment (Phase 2, linear model) .	289
A.7	Objective function values in parameter experiment (Phase 1, nonlinear model) .	290
A.8	Computation times of the parameter experiment (Phase 1, nonlinear model) . . .	290
A.9	The infeasible incumbents of parameter experiment (Phase 1, nonlinear model) .	290
A.10	Objective function values in parameter experiment (Phase 2, nonlinear model) .	291
A.11	Computation times of the parameter experiment (Phase 2, nonlinear model) . . .	291
A.12	The infeasible incumbents of parameter experiment (Phase 2, nonlinear model) .	291

List of Algorithms

5.1	Determining an SA initial temperature	90
5.2	Ejection chain SA move operator	94
5.3	GMS simulated annealing	96

CHAPTER 1

Introduction

Contents

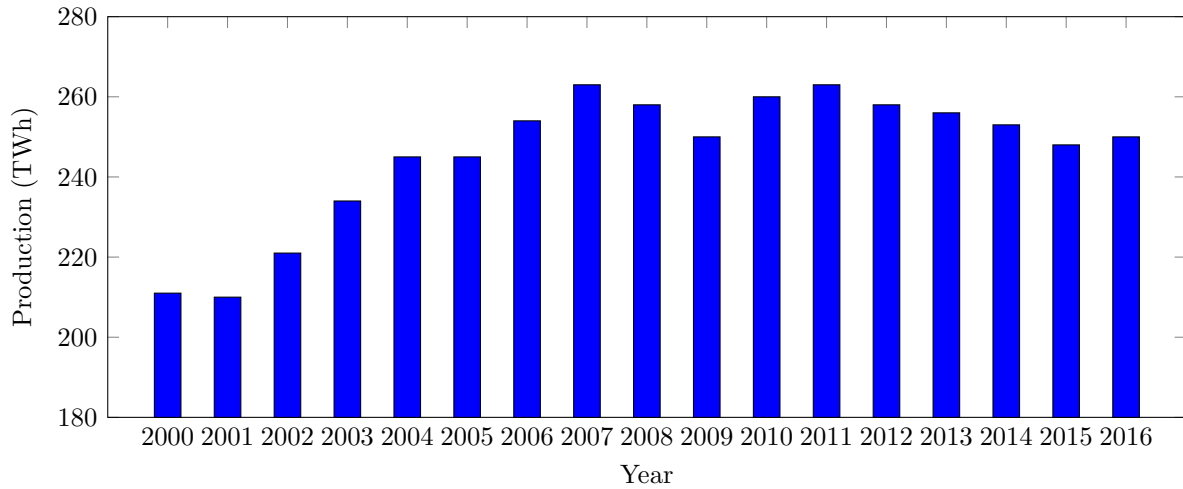
1.1	Background	1
1.2	Informal problem description	4
1.3	Dissertation scope and objectives	5
1.4	Dissertation organisation	6

1.1 Background

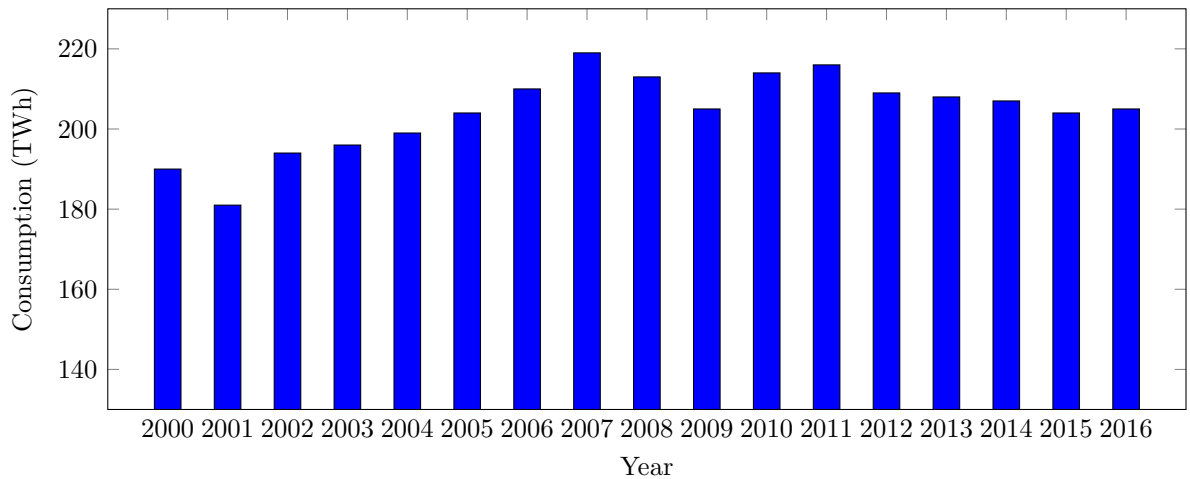
The sustainability of modern society is heavily dependent on a secure and accessible supply of energy [63]. Power utilities have become one of the most crucial resources in any nation's economy and therefore the operations planning of these utilities is of utmost importance [104]. Such operations planning is particularly challenging in the context of developing countries, because the electricity demands in these countries typically increase rapidly due to fast evolving economies [12]. Not only are electricity utilities pressured to meet the ever-changing demands in such countries, they are also pressured to remain current in respect of global energy policies that are currently pushing for "greener energy." This places additional strain on the capital reserves of developing countries, because cleaner energy sources typically still come at a much greater cost than the more traditional methods of electricity generation [63].

In South Africa, which is classified as a developing country [119], Eskom is the sole power utility and was established in 1923 as the Electricity Supply Commission. It was then converted into a public, limited-liability company in 2002 that is wholly owned by the government of South Africa [75]. Today Eskom is recognised as one of the top twenty power utilities in the world, based on generation capacity, with a net maximum self-generated capacity of approximately 44 184 MW. Eskom supplies more than 45% of the electricity consumed in Africa and supplies approximately 96% of South Africa's electricity [76]. The production and consumption of power in South Africa over the past 17 years are presented in Figures 1.1(a) and 1.1(b), respectively.

Although almost 86% of the total installed generation capacity in South Africa is coal-fire based, alternative forms of renewable energy sources are constantly researched by the utility. The combination of the various technologies used to generate electricity is called the *plant mix* which, for Eskom, is shown in Table 1.1 [76]. As may be seen in the table, Eskom utilises six technologies in its electricity generation operations. Power stations are either classified as base load stations or as peak demand stations. Base load stations are stations that operate every day



(a) The power production in South Africa over the past 17 years [72]



(b) The power consumption in South Africa over the past 17 years [72]

FIGURE 1.1: The energy production and consumption of South Africa over the past 17 years.

of the year, whereas peak demand stations only operate when the demand exceeds the supply capability of the base load stations.

Coal-fired power stations use coal as their primary fuel source. These stations are the workhorses of the South African power utility and are required to operate 24 hours a day in order to meet the country's energy demand, with the exception of being offline when scheduled for planned maintenance or when failures occur.

TABLE 1.1: The plant mix of Eskom's electricity generating technologies as on 4 October 2017 [76].

Type	# Power Plants	Type	Plant mix
Coal-fired stations	13	Base Load	85.43%
Nuclear stations	1	Base Load	4.32%
Hydro stations	2	Peak Demand	1.36%
Pumped storage schemes	2	Peak Demand	3.17%
Gas fired stations	4	Peak Demand	5.49%
Wind farms	2	Base Load	0.23%

Africa's first and only nuclear power station, Koeberg, has a total installed capacity of 1 910 MW and is located in Melkbosstrand within the Western Cape [76].

The third generation technology that Eskom employs is hydro energy power stations. These peak demand stations capture the energy of moving water from dams or in rivers and convert the kinetic energy to electrical energy. The two hydro power stations of Eskom are located in the Gariep dam near Norvalspont and in the Vanderkloof dam near Petrusville, respectively [76]. Another technology which also uses the kinetic energy of moving water is a pump storage scheme. This type of facility is also classified as a peak demand station. These stations work on the same principle as hydro power stations, but reuse the water that was used to generate electricity by pumping it back to a storage reservoir up-river during off-peak times to be reused during subsequent peak demand times [77].

The plant mix also contains four gas turbine power stations which have very quick start-up times, but also very high operating costs due to their use of kerosene and diesel as primary fuel sources. These stations are therefore only used during peak demand periods and during emergencies.

Eskom has additionally invested in two wind farms, one of which has already been in operation since 2002 and another which came into full operation early in 2015. The first wind farm consists of only three wind turbines with an installed capacity of merely 3 MW. The second wind farm, which is situated near Vredendal in the Western Cape, consists of 46 wind turbines with a combined installed capacity of 100 MW [76].

During 2015, the electricity system of South Africa was severely challenged and experienced a tightly constrained demand/supply balance, which put the power system at risk in respect of both its adequacy and reliability. This situation may be attributed to the fact that during the 1980s there was an excess supply of electricity due to an electricity generation expansion programme launched by Eskom during the late 1970s. As a result, little or no investments were made in the generation expansion of Eskom during the 1990s and early 2000s [125].

Also contributing to the highly constrained South African energy system, was the higher than expected demand since 2008. This caused nationwide blackouts and since then strain on the system has only recently decreased. Following this, Eskom introduced "load shedding" to the nation. Load shedding involves a series of planned rolling blackouts that follows a rotating schedule. It is used to decrease the demand during periods when Eskom cannot meet the required demand and the short supply threatens the integrity of the nation's electricity grid [103].

Another reason for the situation that Eskom found itself in during 2015, is inadequate maintenance planning. It was recognised at the quarterly *State of the System* briefing held on January 15th, 2015 that a major reason for the previous South African energy situation is that Eskom had not performed the necessary maintenance on the *power generating units* (PGUs) of its power plants [89]. The required maintenance downtimes of PGUs had been postponed on a continual basis due to the high energy demand experienced. Although Eskom had a specific maintenance philosophy in place, it had not remained completely faithful to this philosophy and was therefore faced with massive challenges [89].

A power utility's ability to satisfy energy demand can be influenced significantly by unexpected breakdowns of PGUs. In most cases, such unexpected failures are also much more expensive to repair than taking planned preventative maintenance action. Maintenance of ageing PGUs is, however, often neglected due to high energy demand and low system capacity, as seen in the case of Eskom. The typical objectives pursued in the design of PGU maintenance schedules do not take these difficulties into account. Two new scheduling criteria are therefore proposed in

this dissertation in order to explore to what extent *generator maintenance scheduling* (GMS) optimisation can contribute to the reliability of a power utility's operations.

1.2 Informal problem description

An electricity utility, such as Eskom, typically strives to maintain a reserve margin for generation capacity of 15%. During 2015, the capacity required by the South African grid exceeded supply by approximately 7.5%. This meant that the reserve margin for generation capacity dropped to approximately 7.5%, which is half of the desired reserve margin [49]. This low percentage can cause blackouts when sudden fluctuations in electricity demand are experienced and is therefore a serious problem for any power utility. In the case of Eskom, this drop in reserve margin is partially attributable to the utility not remaining entirely faithful to its own maintenance philosophy pre 2014/2015, as mentioned in the previous section. After falling behind on maintaining the PGUs of power stations, Eskom was faced with a number of unexpected downtimes at power stations and was therefore often unable to achieve the desired reserve margin, resulting in the implementation of load shedding.

Currently, the capability of providing maintenance scheduling recommendations by determining a power utility's expected capacity to satisfy energy demand is not specifically incorporated in the GMS designs in the literature and therefore research is conducted in this dissertation to enable such a capability. Maintenance schedules may be determined by solving incarnations of the celebrated GMS problem, which aims to find a good planned maintenance schedule for the PGUs in a power system in such a way that energy demand is satisfied effectively and efficiently. Recent and current developments in the GMS literature do not, however, take into account the reliability of the PGUs in a power system when scheduling maintenance. In a power system that has been in operation for a considerable number of years, such as in the case of Eskom, the reliability of the PGUs in the system may be compromised. Hence there exists a need for the capability of scheduling planned preventative maintenance procedures based on the probability that a PGU will fail, which is very dependent on the PGU's age. The expected energy produced by a system over a given scheduling window may also be determined based on the probability of PGU failure, which may be employed as an alternative scheduling criterion.

Two novel scheduling criteria are consequently proposed as objective functions to the GMS problem in this dissertation, the first of which takes into account the probability of PGUs failing. This function is proposed so as to be able to quantify an entire power system's reliability in terms of PGU failures. Another novel scheduling criterion is also proposed as an alternative objective function for the GMS problem which aims to quantify an entire power system's reliability in terms of the expected energy production over a given scheduling window, taking into account the probability of PGU failure. Both these scheduling criteria are included in the constrained framework of standard GMS models in order to analyse the effectiveness of scheduling planned maintenance according to such criteria. The proposed scheduling criteria are expected to facilitate reduction of the chance of load shedding having to be implemented by scheduling planned maintenance in pursuit of avoiding PGU failures — events that may cause sudden drops in available energy capacity.

The aforementioned two GMS model considerations may then be employed to provide good maintenance schedules for power systems and may eventually be incorporated into a decision support framework in aid of power utilities with respect to daily GMS decision making.

1.3 Dissertation scope and objectives

The following twelve objectives are pursued in this dissertation:

- I To *conduct* a thorough survey of the literature related to:
 - (a) the elements typically taken into consideration in GMS decisions in general,
 - (b) the derivation of model formulations for the GMS problem,
 - (c) appropriate solution approaches designed for instances of the GMS problem,
 - (d) the constituent models and methodologies of reliability theory in general,
 - (e) trend tests related to identifying repairable and non-repairable systems,
 - (f) lifetime models related to repairable and non-repairable systems in reliability theory,
 - (g) estimation methods for repairable and non-repairable lifetime model parameters,
 - (h) acceleration models for failure analysis.
- II To *establish* a mathematical modelling framework capable of quantifying the reliability of an electric power system in terms of:
 - (a) power generating unit failure, and
 - (b) expected energy productionover a specified maintenance planning horizon, which may be used in optimisation approaches toward solving instances of the GMS problem. The framework should include the techniques researched in pursuit of Objectives I(c)–(h).
- III To *formulate* novel single-objective combinatorial optimisation models for the GMS problem in the context of maximising power system reliability. The models should be based on the modelling frameworks of Objectives II(a) and II(b).
- IV To *design* a generic *decision support system* (DSS) capable of providing good planned maintenance schedules for the PGUs in a power system aimed at maximising the system's reliability (by either minimising probability of unit failure or maximising expected energy production). The DSS should draw on the techniques researched in pursuit of Objective I and should contain the models formulated in pursuit of Objective III.
- V To *implement* a concept demonstrator of the DSS of Objective IV in a suitable application software platform. The concept demonstrator should be able to take as input the demand and system specifications of a user-specified power generating system, as well as related user-specified parameters for the GMS problem, and it should produce as output a high-quality maintenance schedule for the PGUs in the system.
- VI To *verify* and *validate* the system designed in pursuit of Objective IV according to generally accepted modelling guidelines. An exact solution approach should be employed in the context of small problem instances and an approximate solution approach in the context of larger problem instances to aid in the verification and validation process.
- VII To *apply* the system designed in pursuit of Objective IV in special case studies involving the celebrated 21-unit GMS test system [54] and the IEEE-RTS [188] benchmark GMS system in the literature.

- VIII To *evaluate* the effectiveness of the system's outputs in the context of the case studies of Objective VII.
- IX To *establish* a real-world case study which is based on the power system of the national power utility of South Africa, Eskom.
- X To *apply* the system designed in pursuit of Objective IV to the real-world case study of Objective IX.
- XI To *evaluate* the effectiveness of the system's outputs in the context of the real-world case study of Objective IX.
- XII To *recommend* sensible follow-up work related to the work in this dissertation which may be pursued in the future.

The scope of this dissertation is such that other related energy problems, such as the unit commitment problem [186, 192, 207], the economic dispatch problem [16, 82] or the transmission scheduling problem [67, 68, 152], are not considered. The research is also limited by only taking into account the risk associated with unexpected failures of PGUs and not the risk associated with any lost opportunity cost as a result of planned maintenance being performed.

1.4 Dissertation organisation

This dissertation comprises twelve further chapters which are organised into five parts, as well as two appendices. In §2, which is the first chapter in a two-chapter part dedicated to a review of the literature, the reader is introduced to the GMS problem and a number of considerations generally addressed when formulating GMS models, including energy problems that are related to the GMS problem. A review of the literature on GMS model formulations is presented next, including mathematical programming formulations containing popular objective functions as well as constraint sets typically included. This is followed by a thorough review of solution approaches that have been adopted in the literature to solve instances of the GMS problem, including mathematical programming techniques, expert systems, fuzzy logic approaches, heuristics and metaheuristics. A separate, more detailed review section is dedicated to the method of *simulated annealing* (SA), as this method is applied as the approximate model solution methodology in this dissertation.

The second chapter in Part I of this dissertation, §3, is devoted to reliability theory. The reader is introduced to various fundamental concepts in the theory of reliability. This is followed by a review of the basic mathematical notations required to represent various ideas within reliability theory. Two types of systems are typically considered in reliability theory, namely non-repairable systems and repairable systems, both of which are described in §3, along with popular distribution models used in each of these cases. The section is followed by a description of typical failure data employed to approximate failure models for these systems. A section on trend tests is also included. These tests may be employed to determine whether or not there exists a trend in the failure times of a data set. The next section is devoted to the estimation of failure model parameters, including the maximum likelihood method, the least squares method and the Bayesian parameter estimation method. In the final section of the chapter, the reader is introduced to the notation of acceleration models which describe ways of modelling systems that operate under high stress.

Part II of this dissertation also contains two chapters and is concerned with establishing a mathematical framework to newly propose GMS models. In §4, mathematical models are

formulated for two incarnations of the GMS problem considered in this dissertation. The first section of the chapter provides a motivation for the design the two newly proposed objective functions as scheduling criteria, one of which is linear and one which is nonlinear. This is followed by a discussion on the assumptions made in order to facilitate derivation of both mathematical models. The next section contains a description of the actual GMS models adopted in this dissertation. The section includes a detailed derivation of the two proposed objective functions and a mathematical representation of the model constraints. The constraint sets included in the GMS models are energy demand satisfaction constraints, maintenance window constraints, maintenance resource constraints and maintenance exclusion constraints.

Two methodologies are adopted in this dissertation to solve the GMS models of §4. These methodologies are described in some detail in §5, which is the second chapter in Part II. One of these methodologies is an exact solution approach and the other is an approximate solution approach. Both methodologies are particularly tailored to instances of the GMS models proposed in §4. The first section of the chapter contains a description of a method which may be employed to linearise the nonlinear GMS model proposed in §4.3. A piecewise linear approximation method is proposed in which the optimal positions for breakpoints are determined by employing dynamic programming. The second section of the chapter contains a description of the exact solution approach adopted, a motivation for the choice of optimisation platform (CPLEX) and a description of the implementations within this platform of both the linear and nonlinear models. The third section contains a general introduction to the approximate solution approach, namely the method of SA. The section also includes a motivation for the choice of this solution methodology as well as a detailed discussion on the implementation of the method of SA. The discussion on the implementation of the technique covers a method for determining initial solutions and the initial temperature for the algorithm, the cooling and reheating schedules employed, the constraint handling technique implemented, the epoch management protocol adopted, the neighbourhood move operator selected and the termination criteria enforced.

The third part of this dissertation contains three chapters and is dedicated to a documentation of and discussion on the results returned by the various solution methods for the proposed GMS models with respect to two academic benchmark systems from the literature. The two benchmark systems considered in order to test the effectiveness of the newly proposed GMS objectives and the two solution approaches are described in the first chapter of Part III, §6. This description includes reference to the input data and parameters pertaining to these two systems. These two systems are a 21-unit test system and the celebrated 32-unit IEEE-RTS. The more basic 21-unit test system is presented in the first section of the chapter, contains 21 PGUs and exhibits a constant peak demand over a scheduling horizon of 52 one-week planning periods. The second section contains a description of the larger IEEE-RTS, which contains 32 PGUs and exhibits a varying peak demand, attributed to seasonal demand requirements.

In the second chapter of Part III, the seventh chapter of the dissertation, the numerical results obtained when employing the minimisation of the probability of unit failure scheduling criterion are presented. The first section of the chapter contains a presentation of the results returned by the exact solution approach adopted for both the 21-unit test system and the IEEE-RTS. The results of this section represent optimal maintenance schedules in respect of the newly proposed GMS objective function as well as an analysis of the manpower required and available system capacity associated with these solutions. The optimal solutions obtained are also compared to solutions from the literature for an alternative GMS model involving the well-known minimisation of the sum of squared reserve margins as scheduling criterion. Sensitivity analyses are also performed in order to analyse the feasibility of the exact solution approach for small systems such as the 21-unit test system and the IEEE-RTS. In the second section of the chapter, the

approximate solution approach results are presented for both the 21-unit test system and the IEEE-RTS. In this section, the results of the experimental design followed to determine the best combination of parameters for use in the method of SA are presented for both benchmark systems. These parameters include an initial SA acceptance ratio, a constraint violation severity factor, a cooling parameter, a reheating parameter and an epoch termination parameter. The presentation of the results obtained from the parameter optimisation experiment include comparisons between the different combinations of parameters in order to obtain the best combination. Thereafter, the section also contains the solutions obtained when solving both the 21-unit test system and the IEEE-RTS by the method of SA as well as comparisons of the solutions thus obtained with the corresponding exact solutions.

The final chapter of Part III, which is the eighth chapter in the dissertation, follows the same structure as that of §7, but the numerical results obtained by employing the maximisation of the expected energy production scheduling criterion are presented instead. The first section of the chapter contains a presentation of the results returned by the exact solution approach after employing the piecewise linearisation approach described in §5. This presentation includes results for both the 21-unit test system and the IEEE-RTS. An optimal maintenance schedule for the piecewise linear approximation is provided in respect of the newly proposed GMS objective function as is an analysis of the manpower required and available system capacity associated with these solutions. The optimal solutions obtained are also compared with solutions from the literature for a GMS model involving the well-known minimisation of the sum of squared reserve margins as scheduling criterion. Sensitivity analyses are also performed in order to analyse the feasibility of the exact solution approach for small systems, such as the 21-unit test system and the IEEE-RTS, as was performed for the minimisation of the probability of unit failure scheduling criterion. In the second section of this chapter, the approximate solution approach results obtained when employing the maximisation of the expected energy production scheduling criterion are presented for both the 21-unit test system and the IEEE-RTS. In this section, the results of the experimental design followed to determine the best combination of parameters for use in the method of SA are presented for both benchmark systems are presented. The same parameter ranges are considered for the maximisation of the expected energy production scheduling criterion as was the case for the minimisation of the probability of unit failure scheduling criterion in order to obtain the best combination. Thereafter, solutions are presented for the 21-unit test system and the IEEE-RTS by applying the SA solution approach upon adoption of the best parameter value combination returned by the experimental design. The section closes with comparisons of the solutions thus obtained with the exact solutions.

The penultimate part of the dissertation, Part IV, is aimed at facilitating application of the proposed GMS to a real-world scenario. This part consists of three chapters, the first of which contains a description of the real-world case study considered in some detail in this dissertation. The case study is called the 157-unit Eskom case study and is based on the energy grid of the national power utility in South Africa. The first section of §9 provides some background on the case study and this is followed by detailed specifications of the power system. In the final section of the chapter, some extensions are proposed to the linear and nonlinear GMS objective functions of §4.3 in order to accommodate the situation where PGUs have to be scheduled for maintenance multiple times within a given scheduling window.

The results obtained by employing the approximate solution approach adopted in this dissertation to the real-world case study of §9, are presented and analysed in §10, the second chapter in Part IV, within the context of both the linear and nonlinear GMS models proposed in §4. The first section of the chapter is dedicated to the results obtained by employing the minimisation of the probability of unit failure scheduling criterion as well as an analysis of the available

system capacity associated with this solution. The second section follows the same structure, but is dedicated to the results obtained by employing the maximisation of the expected energy production GMS objective function.

Part IV finally closes in §11 with the proposal of a computerised DSS, designed by the author, which is aimed at facilitating effective GMS decision making. In the first section of the chapter, some general consideration is given to typical DSS development. This includes a description of the three main components of a typical DSS and is followed by a detailed process description of the GMS DSS proposed. Finally, the system is described in a comprehensive system walk-through fashion aimed at informing potential users how to utilise the DSS to its full potential.

The final part of this dissertation, Part V, contains two final chapters. The penultimate chapter of the dissertation, §12, contains a summary of the research performed, as documented in this dissertation, as well as an appraisal of the contributions made within this dissertation.

The final chapter, §13, contains an elaboration on seven suggestions for future work in the field of GMS that may follow naturally on the research performed in this dissertation.

Part I

Literature review

CHAPTER 2

Generator maintenance scheduling

Contents

2.1 Model considerations	13
2.2 GMS model formulations in literature	17
2.3 GMS model solution approaches	26
2.4 The method of simulated annealing	36
2.5 Chapter summary	39

This chapter contains a literature review on the *generator maintenance scheduling* (GMS) problem. Various generic GMS modelling considerations are considered in §2.1. In §2.2, a brief overview is given of the most commonly adopted GMS model formulations. This is followed in §2.4 by a brief description of the celebrated method of simulated annealing as applicable to GMS problem instances. Other model solution approaches are also described in §2.3, including various mathematical programming techniques, as well as the use of expert systems, fuzzy logic, heuristics and metaheuristics other than simulated annealing.

2.1 Model considerations

Scheduling the maintenance of the PGUs of power utilities is of utmost importance in energy system design, planning and operations management [188]. The planning involved the scheduling of such units for maintenance increases in complexity as the number of PGUs increases and as the reserve margin of the system decreases (*i.e.* the demand increases) [188]. GMS typically involves scheduling PGUs for preventative maintenance over a certain scheduling window in pursuit of a set of objectives and subject to a number of operational and system constraints in order to ensure schedule feasibility. Preventative maintenance performed on PGUs is directly related to a power utility's ability to supply the required energy demand [135]. Three main goals are typically pursued in the design of good maintenance schedules for the PGUs of a power utility [5], namely:

- (a) to increase the economic benefits and the reliability of the entire electrical system,
- (b) to extend the lifetime of the individual PGUs, and
- (c) to avoid the installation of new PGUs.

Before the GMS problem may be formulated mathematically, four planning considerations have to be taken into account, namely the length of the scheduling window, the scheduling resolution, the particular constraints included in the model and the scheduling objectives to be pursued.

2.1.1 The scheduling window

The GMS problem is formulated over a so-called *scheduling window* which is an interval of planning periods over which the user, typically a power utility, aims to schedule preventative maintenance of PGUs into the future. The scheduling window of the GMS problem typically varies from a number of weeks to a number of years in the literature, although one year is the most common scheduling window adopted [5, 134], since PGUs are generally serviced on an annual basis [137].

Another reason why a one-year scheduling window is such a popular choice may be that the load demand is typically forecast by power utilities over a period of some multiple of years, because power utilities typically attempt to include all four seasons in their forecast planning [188]. There are, however, cases in the literature where the length of the scheduling window is anything between eight weeks [66] to five years [166]. Generally, a power utility decides *a priori* on the scheduling window to be employed based on the nature of the data that it possesses. Burke *et al.* [36] considered a case where the scheduling window has an influence on the next scheduling window in the hope of obtaining some sort of periodicity.

2.1.2 The scheduling resolution

The scheduling resolution in the GMS problem refers to the length of a single planning period within the scheduling window. The most common scheduling resolution is typically weekly [134], since preventative maintenance of PGUs often requires a minimum of one week of downtime. In the literature there are, however, instances involving higher scheduling resolutions, such as daily or even hourly resolutions [86, 107].

Once again, there is no universally adopted scheduling resolution for the GMS problem — the user typically decides *a priori* on a suitable scheduling resolution based on the system requirements as well as the system data that it is able to collect.

2.1.3 The objective function

It is common in GMS models to pursue only a single scheduling objective. Various lesser important scheduling objectives are typically incorporated in GMS model formulations as constraints. In the literature, three classes of GMS scheduling criteria prevail, namely *economic criteria*, *reliability criteria* and *convenience criteria* [53, 135, 227]. Of these three classes of criteria, economic and reliability criteria are most commonly employed for modelling purposes [46, 53, 71, 227]. Economic scheduling criteria in the context of the GMS problem usually entail the minimisation of two main cost components, namely maintenance cost and energy production cost [53]. In many cases, the maintenance cost is, however, ignored in favour of considering only the energy production cost when scheduling maintenance. The reason for this omission is that production cost is typically much larger than maintenance cost [107, 133]. There are nevertheless also cases in the literature where only maintenance cost is adopted as scheduling criterion [168]. In some cases, energy replacement cost is also employed as scheduling criterion [227]. Energy replacement cost refers to the change in maintenance cost associated with

swapping the maintenance slot of a PGU, due to a change in the initial maintenance schedule, with that of a generally more expensive PGU [227]. The maintenance cost component of an economic scheduling criterion is especially important when the duration of the planned outages of the PGUs are allowed to vary within given limits. Maintenance cost is typically linked with the additional appointment of staff members tasked with conducting maintenance operations during a certain time interval or the overtime of staff members already appointed [53]. Recent competition between power utilities in certain countries has also resulted in scheduling criteria which aim to maximise the profit of a power utility instead of minimising its cost [128, 225]. Combinations of various criteria from one or more of the above-mentioned classes of GMS criteria have also been adopted, which yield multi-objective GMS formulations within the class of economic scheduling criteria GMS models [162].

Reliability scheduling criteria is seen as the most important class of scheduling criteria in the context of the GMS problem [227]. A reliability scheduling criterion may be one of a number of alternatives of which the most common is levelling the reserve generation margin over the entire scheduling window [53]. This objective aims to minimise the sum of the differences between the reserve margin and the demand per scheduling period. Other popular reliability-related GMS criteria include minimising the *loss of load probability* (LOLP), minimising the *expected unsupplied energy* (EUE) and minimising the *loss of load expectation* (LOLE) [166]. The LOLP is calculated as the expected probability that the system load will exceed the generation capability of the system. This probability is obtained from a probability distribution function which takes into account the daily load variations [227]. The EUE scheduling criterion specifies the expected unsupplied energy as a monetary value which represents the purchase of unsupplied energy from other power utilities at premium cost. The LOLE specifies the probability of losing a certain amount of capacity due to maintenance or failures of PGUs in the power system.

A class of scheduling criteria that is not adopted as often in GMS as the aforementioned criteria is the class of convenience criteria, also known as *maintenance schedule deviation* criteria. This type of scheduling criterion is formulated in terms of four main elements, namely ideal preventative maintenance frequency, maintenance urgency, changes in previously established maintenance schedules and the ideal maintenance sequences of various PGUs [227]. It may also involve minimising the violation of soft constraints in the model. Members of the class of convenience scheduling criteria are seldom used as main scheduling criteria, but are often used in conjunction with other scheduling criteria [134, 227, 131].

When modelling the GMS problem as a multi-objective problem, any combination of the above-mentioned criteria from the three classes of economic, reliability and convenience criteria may, of course, be included in the formulation. Examples of different combinations of members of the three classes of criteria employed within the context of GMS may be found in [107, 134, 141].

2.1.4 The model constraints

The GMS problem naturally includes a number of model constraints which may either be incorporated as soft or hard constraints. If soft constraints are violated by candidate solutions, these solutions are still considered feasible, although the objective function is typically penalised by the number of such constraint violations and by the degree of each violation. Hard constraints, on the other hand, may not be violated by candidate solutions to the GMS problem in order to be considered feasible [108].

The *demand satisfaction constraint* is perhaps the most important constraint in model formulations of the GMS problem and may be included either as a soft or a hard constraint. This constraint is aimed at ensuring that the demand for power that has to be provided by the system

of PGUs is met during each planning period in a discretisation of the scheduling window under consideration. The capacity of the power system will vary since PGUs are scheduled for planned maintenance and their temporary unavailability therefore has to be taken into account when calculating the capacity of the system. If this constraint is modelled as a soft constraint, then the objective function may be penalised by assigning a monetary value to not being able to satisfy demand. In the case where the demand satisfaction constraint is modelled as a hard constraint, solutions are only considered feasible if, during each planning period of the scheduling window, the demand for power is satisfied completely [55].

Maintenance window constraints specify earliest and latest planning periods during which each PGU should be scheduled for planned maintenance and are typically modelled as hard constraints. Determining suitable values for these earliest and latest planning periods may prove difficult from a practical point of view — they are often merely subjective rules of thumb specified by the management of a power system or by the suppliers of some of the large components of the PGUs [65].

Resource availability constraints involve ensuring that adequate amounts of resources are available to perform maintenance on PGUs during all time periods during which they are scheduled to undergo maintenance. An example of a constraint in this class involves ensuring that an adequate number of maintenance crew members are available during the scheduling window to perform the actual maintenance on the PGUs scheduled for maintenance at any time during the scheduling window. This type of constraint can also be modelled either as a soft or a hard constraint. The advantage of modelling this type of constraint as a soft constraint is that an actual value may be assigned to hiring more personnel at a certain time during the scheduling window, by which the objective function may be penalised while still obtaining feasible solutions [65].

The class of scheduling *duration constraints* specify the duration, relative to the scheduling resolution, measured in consecutive time periods, required to perform planned maintenance on each PGU. Each PGU will therefore typically have a duration constraint which specifies how long the PGU will be offline when it is scheduled for planned maintenance [55].

Other examples of hard constraints are *service contiguity constraints*, *exclusion constraints* and *precedence constraints*. Service contiguity constraints specify that the planned downtime of a PGU for maintenance purposes should be a contiguous period of time (*i.e.* should not contain interruptions) [55]. Exclusion constraints specify sets of PGUs that are not allowed to be scheduled for maintenance simultaneously. Finally, precedence constraints specify that certain PGUs within the power system should be scheduled for maintenance before certain other PGUs may undergo maintenance.

2.1.5 Related energy problems

Apart from the GMS problem, other scheduling problems also exist in the literature on power systems. The three main scheduling problems, excluding GMS, in this domain are the *unit commitment* (UC) problem, the *economic dispatch* (ED) problem, and the *transmission line maintenance scheduling* problem. Brief overviews of these three problems and their relations to the GMS problem are provided in this section.

The unit commitment problem

The UC problem is concerned with determining which PGUs should be in service in order to meet the demand required by the power system during each time interval of the scheduling window [186]. In order to accurately plan which PGUs should be in service, a small scheduling window is

typically chosen since more accurate demand predictions may be obtained over a shorter interval than over a longer one. The scheduling resolution for the UC problem is typically taken as daily or weekly [192]. The UC problem is similar to the GMS problem, but differs from it as result of the constraints that are imposed. The GMS problem determines which PGUs should be scheduled for planned maintenance, whereas the UC problem determines which PGUs that are not scheduled for planned maintenance, should be scheduled to be in service in order to meet the required demand. A typical objective in the UC problem is to schedule PGUs to be in service by minimising the production cost over the scheduling window. These two problems may also be modelled as a combined problem in which the GMS problem's long-term solution serves as an input to the UC problem over a short-term scheduling window [207]. This may cause a small amount of inaccuracy in demand prediction due to the GMS problem potentially being based on less accurate long-term demand prediction, whereas the UC problem requires more accurate short-term demand prediction. A different approach may also be followed where the GMS problem utilises accurate short-term demand predictions, but this often causes dimensionality problems which may render the process of finding a good solution very difficult [207].

The economic dispatch problem

The ED problem determines how the load should be distributed among the PGUs that are determined to be in service by the UC problem. As in the UC problem, the objective according to which demand satisfaction is distributed among the PGUs that are in service typically involves the minimisation of the overall power generation cost subject to operational and transmission constraints [16]. Those PGUs that are in service and incur the smallest energy production and operational cost will typically be used first to meet the demand required [82]. The input to the ED problem depends on the output of the UC problem and these two problems are sometimes modelled in conjunction with one another in practice. This also implies that an instance specification of the GMS problem is indirectly influenced by the solution of the ED problem, and may therefore be included in a model where the GMS, UC and ED problems are all combined.

The transmission line maintenance scheduling problem

Other important components of the power system that require maintenance are the substations and the transmission lines connecting the PGUs and the substations. These transmission lines and substations can only be maintained within very specific time windows, since it is required that the PGU which supplies a substation and connecting transmission lines being serviced, should itself also be out of service. Since transmission lines cannot be scheduled for planned maintenance if the supplying PGU is in service, the GMS problem and the transmission line maintenance scheduling problem are often modelled together [67, 68, 152].

2.2 GMS model formulations in literature

The GMS problem differs from typical scheduling problems in the literature. Models for many scheduling problems, such as *job scheduling*¹ or *nurse scheduling*², are formulated as one of the

¹The allocation of resources in a system to perform different tasks. A number of jobs typically await processing and the decision maker has to determine which jobs should be prioritised and the amount of time that should be allocated to these prioritised jobs [202].

²A scheduling problem in which an optimal schedule of working shifts has to be constructed for nurses employed by a hospital. Also referred to as the *nurse rostering problem* [61].

classical optimisation problems found in operations research literature, such as the *assignment problem*³ or the *vehicle routing problem*⁴. The GMS problem has, however, seldom been formulated as one of these optimisation problems. Mromlinski [168] nevertheless modelled the GMS problem as a binary transportation problem in 1985.

According to Schlünz [188], one possible reason why the GMS problem is seldom formulated as a classical optimisation problem is the large degree of variability across instances of the GMS problem. GMS problem instances may involve optimising a single objective or optimising multiple conflicting objectives simultaneously in which trade-offs are sought. The objective function(s) adopted in the GMS problem may also be either linear or nonlinear, which provides for even greater variability in GMS problem formulations.

In the remainder of this section, the most popular model formulations for the GMS problem in the literature are discussed.

2.2.1 Objective function formulation

Suppose a power system contains n PGUs denoted by the set $\mathcal{U} = \{1, \dots, n\}$ and that the scheduling window for the power system is discretised into m planning periods of equal length denoted by the set $\mathcal{P} = \{1, \dots, m\}$. Let $x_{u,p}$ be a binary decision variable taking the value 1 if planned maintenance of PGU $u \in \mathcal{U}$ is scheduled to start during planning period $p \in \mathcal{P}$, or zero otherwise, and let $y_{u,p}$ be a binary auxiliary variable taking the value 1 if planned maintenance of PGU $u \in \mathcal{U}$ is scheduled during planning period $p \in \mathcal{P}$, or zero otherwise.

Furthermore, let $c_{u,p}^{Pr}$ denote the energy production cost of unit $u \in \mathcal{U}$ during planning period $p \in \mathcal{P}$ and let $c_{u,p}^{Ma}$ denote the maintenance cost incurred if unit $u \in \mathcal{U}$ is scheduled for maintenance during planning period $p \in \mathcal{P}$. Then the most commonly adopted economic scheduling criterion adopted in the literature is to

$$\text{minimise} \quad \sum_{u \in \mathcal{U}} \sum_{p \in \mathcal{P}} [c_{u,p}^{Pr}(1 - y_{u,p}) + c_{u,p}^{Ma}y_{u,p}]. \quad (2.1)$$

A number of variations on the objective function (2.1) also exist in the literature. One such variation is due to Edwin and Curtius [64], who only considered the energy production cost incurred by the power system as scheduling criterion. This objective function is therefore to

$$\text{minimise} \quad \sum_{u \in \mathcal{U}} \sum_{p \in \mathcal{P}} c_{u,p}^{Pr}(1 - y_{u,p}). \quad (2.2)$$

Another variation on (2.1) was proposed by Mromlinski [168] in 1985, who only considered maintenance cost as scheduling criterion. A further adaptation due to Mromlinski [168] involved weighting the production cost by the rated generating capacities of the various PGUs as scheduling criterion. Let C_u denote the rated generating capacity of PGU $u \in \mathcal{U}$. Then the scheduling objective in the formulation of Mromlinski [168] is to

$$\text{minimise} \quad \sum_{u \in \mathcal{U}} \sum_{p \in \mathcal{P}} c_{u,p}^{Pr}(1 - y_{u,p})C_u. \quad (2.3)$$

By multiplying the generating capacity by the maintenance cost of each PGU, a larger cost coefficient is achieved for PGUs in the system with a larger generating capacity. This adds some

³A well-known special case of the transportation problem. A job can only be assigned to one resource and a resource can only be assigned one job [218].

⁴A celebrated combinatorial optimisation problem which aims to find an optimal set of routes for a fixed fleet of vehicles which has to service a certain number of customers [139].

realism to the GMS problem in the sense that larger PGUs may typically require more expensive parts and may take longer to repair [168].

Another economic scheduling criterion involves minimising the fuel cost associated with generating electricity [57]. This is due to the fact that the production cost of electricity mainly depends on the fuel consumption of the PGUs. An adaptation of (2.2) may be adopted to accommodate fuel consumption cost. Let $c_{u,p}^{Fu}$ denote the cost of fuel for PGU u during planning period p . Then the scheduling objective is to

$$\text{minimise} \quad \sum_{u \in \mathcal{U}} \sum_{p \in \mathcal{P}} c_{u,p}^{Fu} (1 - y_{u,p}). \quad (2.4)$$

A variation on the theme involves taking into account the fuel cost as part of the production cost in the objective function. This was the approach of Digalakis and Margaritis [57], who in 2002 introduced a variable $o_{u,p}$ denoting the output level of PGU $u \in \mathcal{U}$ during planning period $p \in \mathcal{P}$. Their objective was to

$$\text{minimise} \quad \sum_{u \in \mathcal{U}} \sum_{p \in \mathcal{P}} (c_p^{Fu} o_{u,p} + c_{u,p}^{Ma} y_{u,p}), \quad (2.5)$$

where c_p^{Fu} denotes the fuel cost per unit in a homogeneous network of PGUs during planning period $p \in \mathcal{P}$. The objective function (2.5) is based on the fact that the fuel cost typically only depends on the output level of the PGU — the cost of fuel remains constant for all the PGUs in the system during any planning period $p \in \mathcal{P}$. If maintenance is performed on PGU $u \in \mathcal{U}$, however, the variable $o_{u,p}$ is equal to zero as no power is generated during maintenance.

GMS model formulations may also be found in the literature where an economic scheduling criterion includes the start-up cost associated with putting a PGU back into operation after having performed planned maintenance on that unit [39, 166]. Canto [39] found that the maintenance cost is insignificant if the scheduling objective includes the start-up cost of the PGUs and that maintenance cost may in this case therefore be omitted from the objective function. It has, in fact, been found that the maintenance cost can be up to one million times smaller than the start-up cost of a specific PGU. This led to an adaptation of (2.1) in which the maintenance cost is replaced by the start-up cost of a PGU. Let $c_{u,p}^{St}$ denote the start-up cost associated with PGU $u \in \mathcal{U}$ during planning period $p \in \mathcal{P}$. Then the objective adopted by Canto [39] is to

$$\text{minimise} \quad \sum_{u \in \mathcal{U}} \sum_{p \in \mathcal{P}} \left[c_{u,p}^{Pr} (1 - y_{u,p}) + c_{u,(x_u+d_u)}^{St} \right], \quad (2.6)$$

where x_u denotes an integer value representing the starting period for planned maintenance on PGU u (that is, $x_u = \sum_{p \in \mathcal{P}} p x_{u,p}$) and where d_u denotes the duration of maintenance on PGU $u \in \mathcal{U}$. Hence, $c_{u,(x_u+d_u)}^{St}$ is the start-up cost for PGU $u \in \mathcal{U}$ given that it is returned to operation during planning period $x_u + d_u$.

One of the most recently proposed economic scheduling criteria involves not only the maintenance cost of the PGUs, but also the maintenance cost of the transmission lines that form part of the power system. This objective yields a more realistic estimation of the total maintenance cost since the generating capability of a PGU depends on both the maintenance of the PGUs in the system as well as the maintenance of the transmission lines in the system that convey power from the PGU. Marwali and Shahidehpour [152], as well as Silva *et al.* [193] initially included maintenance cost of the transmission lines as a set of constraints in their GMS model formulations. In this case, the resulting scheduling problem is sometimes referred to as a *global generator/transmission scheduling problem* [152].

Some examples of objectives from the class of economic scheduling criteria include transmission line maintenance cost as part of the maintenance and production cost [67, 68, 93, 152]. The inclusion of the maintenance cost of the transmission lines introduces a new set of constraints to the problem which are discussed later in this chapter. Let \mathcal{L} denote the set of transmission lines in the power system and let $c_{\ell,p}^{Tr}$ denote the cost of transmission line maintenance if planned maintenance were to be performed on transmission line $\ell \in \mathcal{L}$ during planning period $p \in \mathcal{P}$. Furthermore, let $T_{\ell,p}$ be a binary decision variable taking the value 1 if transmission line $\ell \in \mathcal{L}$ is in maintenance during planning period $p \in \mathcal{P}$, or zero otherwise. Then the objective in this case is to

$$\text{minimise} \quad \sum_{p \in \mathcal{P}} \left[\sum_{u \in \mathcal{U}} c_{u,p}^{Pr} (1 - y_{u,p}) + \sum_{u \in \mathcal{U}} c_{u,p}^{Ma} y_{u,p} + \sum_{\ell \in \mathcal{L}} c_{\ell,p}^{Tr} T_{\ell,p} \right]. \quad (2.7)$$

Although economic GMS criteria are found in abundance in the literature, it is often claimed that the class of reliability criteria gives rise to the most important GMS objective functions [227]. These reliability criteria may be deterministic or stochastic in nature [227]. A large number of reliability GMS criteria appear in the literature — most of them involving the so-called reserve load in some way. The *reserve load* of a system is the excess generating capability that the system has at a given period (*i.e.* after having satisfied demand for that period).

One of the most popular reliability-related GMS objective functions in the literature involves minimising the *sum of squares of the reserve* (SSR) load. This objective is a member of a more general set of objective functions concerned with levelling the reserve load of the system over the entire scheduling window. Let $C_{u,p}$ denote the generating capacity of PGU $u \in \mathcal{U}$ during planning period $p \in \mathcal{P}$ and let D_p denote the expected demand of the system during planning period $p \in \mathcal{P}$ [53, 87]. The demand D_p of the system during period $p \in \mathcal{P}$ may include a certain safety margin required by the user. Then the objective is to

$$\text{minimise} \quad \sum_{p \in \mathcal{P}} r_p^2, \quad (2.8)$$

where

$$r_p = \sum_{u \in \mathcal{U}} C_{u,p} (1 - y_{u,p}) - D_p \quad (2.9)$$

denotes the reserve level of the system during planning period $p \in \mathcal{P}$.

A slightly simpler approach to adopting the SSR load in the objective function of (2.8) is to consider the generating capacity of PGU $u \in \mathcal{U}$ to be constant for all planning periods in \mathcal{P} . In this instance the parameter $C_{u,p}$ in (2.8) reduces to the previously defined parameter C_u (as in (2.3)).

Another GMS objective function, also concerned with levelling the reserve margin, involves minimising the sum of the differences between the average reserve load of the system and the actual reserve load of the system over the scheduling period [22, 66]. In this case the objective is to

$$\text{minimise} \quad \sum_{p \in \mathcal{P}} (\bar{r} - r_p), \quad (2.10)$$

where

$$\bar{r} = \frac{1}{m} \sum_{p \in \mathcal{P}} r_p \quad (2.11)$$

denotes the average reserve load and r_p is as defined in (2.9) for all $p \in \mathcal{P}$.

The scheduling criterion in (2.10) is used less often in the literature due to its inferiority relative to the SSR load in (2.8). This inferiority stems from the fact that the terms in (2.10) may be

positive or negative and may therefore cancel each other out to some extent, which does not entirely represent the capability of the system in terms of meeting demand. A second weakness of the objective function (2.10) is that outliers are penalised more heavily than reserve levels close to the average load in the objective function (2.8) due to the presence of the square in (2.8), which may provide improved solutions in terms of levelling the reserve load.

Another reliability-based GMS criterion found in the literature involves maximising the smallest reserve load over the scheduling period [59, 169, 172], that is, to

$$\text{minimise} \quad \min_{p \in \mathcal{P}} r_p. \quad (2.12)$$

This is also not a very popular objective function, although some instances of its use occur in the literature [59, 169, 172]. The objective function (2.12) may be useful to power utilities which are concerned that they might fail to be able to satisfy demand for energy.

A reliability-related scheduling criterion included in some stochastic models of the GMS problem involves minimising the risk of not satisfying energy demand with the expected available capacity per time period [214]. This LOLP objective aims to minimise the probability of a power system not meeting the load demand with the available capacity, and may be formulated as

$$\text{minimise} \quad LOLP = P(X > R), \quad (2.13)$$

where X denotes a stochastic variable which represents the outage capacity during a certain planning period and where R denotes the power system's reserve capacity, which is the effective capacity of the system C less the maximum load L (also a stochastic variable). Typically, the expected values for outages and loads are used instead of the probability in (2.13) [214]. This is achieved by calculating the number of planning periods for which the maximum load of the power system is expected to exceed the available capacity of the system. The LOLE may therefore be calculated as

$$LOLE = LOLP \times P, \quad (2.14)$$

where P denotes the expected number of planning periods over the scheduling window during which the demand is expected to exceed the available capacity, and the objective is then to minimise the quantity in (2.14).

Another commonly used reliability scheduling criterion is the *expected energy not served* (EENS). This function calculates the expected energy that will not be supplied under conditions when the load of the system exceeds the available generation capacity and is typically measured in units of kWh [45, 122]. A logical way to express the EENS is

$$\text{EENS} = L \frac{D}{3600}, \quad (2.15)$$

where L is the average annual power load of the system in kW and D is the duration of the unavailability of load in seconds [122]. This function can also be expressed in terms of a so-called energy index of reliability,

$$\text{EIR} = 1 - \frac{\text{EENS}}{E_0}, \quad (2.16)$$

where E_0 is the total energy demand of the system over the duration of the scheduling window in kWh [122, 147]. A more complex approach to solving the EENS may be to incorporate the net demand forecast error as a continuous random variable and/or including some generator uncertainties regarding actual generation capability. In such an approach, the generator uncertainties are usually incorporated as a set of binary random variables [147].

The final class of GMS criteria, namely convenience criteria, is seldom employed in single-objective modelling approaches for the GMS problem. In most instances, this criterion is incorporated into multi-objective GMS models where the convenience objective might be to minimise the number of constraints that are violated in the system or to minimise the expected number of changes to the currently planned maintenance schedule of the user [227].

Any of the aforementioned objective functions may, of course, be employed in conjunction with one another to formulate a multi-objective model for the GMS problem. A typical occurrence in the GMS literature is to employ an economic scheduling criterion in combination with a reliability scheduling criterion [66, 133, 166, 224].

2.2.2 Constraint formulation

GMS model instances in the literature are typically formulated in terms of integer decision variables or binary decision variables as exemplified in the discussion on scheduling objectives of the previous section. The adoption of each of these types of decision variables hold both advantages and disadvantages. In this section, both typical GMS integer decision variable constraint sets and typical binary decision variable constraint sets are described. In the case where integer decision variables are used, x_u denotes the starting period for planned maintenance on PGU $u \in \mathcal{U}$ during the scheduling window. In this case, an auxiliary variable $y_{u,p}$ is also employed which takes the value 1 if planned maintenance is scheduled for PGU $u \in \mathcal{U}$ during planning period $p \in \mathcal{P}$, or zero otherwise, as already introduced in §2.2.1. If, however, binary decision variables are adopted, the decision variable $x_{u,p}$ takes the value 1 if planned maintenance is scheduled for PGU $u \in \mathcal{U}$ during planning period $p \in \mathcal{P}$, or zero otherwise, and the same auxiliary variable $y_{u,p}$ is also adopted in this case.

As mentioned in §2.1.4, the demand satisfaction constraint is one of the most important constraint sets in models of the GMS problem. The most basic way of imposing demand satisfaction involves requiring that

$$\sum_{u \in \mathcal{U}} C_{u,p}(1 - y_{u,p}) \geq D_p, \quad p \in \mathcal{P} \quad (2.17)$$

in order to ensure that the demand D_p for energy in the power system during any planning period is met. A slightly more complicated formulation of the demand satisfaction constraint may be found in [87] where the generating capacity C_u of PGU $u \in \mathcal{U}$ is allowed to vary over the planning periods of the scheduling window. In this formulation, the parameter C_u in constraint set (2.17) is replaced by the parameter $C_{u,p}$ which denotes the generating capacity of PGU u during planning period $p \in \mathcal{P}$.

Another variation on the demand satisfaction constraint set (2.17) may be found in [66, 87], where the demand satisfaction constraint set takes into account the required reserve or safety margin in the system. This is achieved by adding a parameter R_p to the right-hand side of constraint set (2.17) which denotes the reserve or safety margin required during planning period $p \in \mathcal{P}$. In this case, the demand constraint is formulated as

$$\sum_{u \in \mathcal{U}} C_{u,p}(1 - y_{u,p}) \geq D_p + R_p, \quad p \in \mathcal{P} \quad (2.18)$$

instead of (2.17). In order to impose demand satisfaction in the case of using binary decision variables, two additional parameter sets are required. The first set \mathcal{U}'_p is the set of PGUs that are allowed to be in a state of planned maintenance during planning period $p \in \mathcal{P}$. Hence, $\mathcal{U}'_p = \{u \mid p \in \mathcal{P}_u\}$, where $\mathcal{P}_u = \{p \in \mathcal{P} \mid e_u \leq p \leq \ell_u\}$ denotes the set of planning periods during which planned maintenance may start for PGU $u \in \mathcal{U}$, with e_u denoting the earliest starting

planning period for planned maintenance on PGU $u \in \mathcal{U}$ and ℓ_u denoting the latest starting planning period for planned maintenance on PGU $u \in \mathcal{U}$. The second set $\mathcal{S}'_{u,p}$ is the set of starting planning periods such that if planned maintenance were to start on PGU $u \in \mathcal{U}$ during such a planning period, then PGU u will be in a state of planned maintenance during planning period $p \in \mathcal{P}$. Therefore, $\mathcal{S}'_{u,p} = \{j \in \mathcal{P}_u \mid p - d_u + 1 \leq j \leq p\}$ [53]. The demand satisfaction constraint set, when using binary decision variables, may then be formulated as

$$\sum_{u \in \mathcal{U}} C_{u,p} - \sum_{u \in \mathcal{U}'} \sum_{j \in \mathcal{S}'_{u,p}} C_{u,j} x_{u,j} \geq D_p + R_p, \quad p \in \mathcal{P}. \quad (2.19)$$

Burke *et al.* [36], as well as Digalakis and Margaritis [57], took a slightly different approach in formulating the demand satisfaction constraint set. In their modelling approaches, the output level of a PGU is not considered a fixed value representing the capacity of the unit. A variable $o_{u,p}$ is rather employed (as introduced in the previous section) which denotes the output level of PGU $u \in \mathcal{U}$ during planning period $p \in \mathcal{P}$. Since this modelling approach takes into account variable output levels of the PGUs, the demand of the system is required to be equal to the output of the fleet of PGUs. The demand satisfaction constraint set of Burke *et al.* [36] and of Digalakis and Margaritis [57] is therefore

$$\sum_{u \in \mathcal{U}} o_{u,p} = D_p, \quad p \in \mathcal{P}. \quad (2.20)$$

An additional constraint set

$$0 \leq o_{u,p} \leq C_{u,p}(1 - y_{u,p}), \quad u \in \mathcal{U}, p \in \mathcal{P} \quad (2.21)$$

is also enforced, which specifies output limits for each PGU. Constraint set (2.21) specifies that the output level of a PGU $u \in \mathcal{U}$ generating $o_{u,p}$ units of power should be between zero and the generating capacity $C_{u,p}$ of the unit during planning period $p \in \mathcal{P}$. The value should, however, be equal to zero if the PGU is in a state of planned maintenance.

Another basic class of constraints in GMS models are maintenance window constraints. This type of constraint set specifies earliest and latest starting times during which planned maintenance may commence on every PGU. If integer variables are employed in the model formulation, then the maintenance window constraints may be formulated as

$$e_u \leq x_u \leq \ell_u, \quad u \in \mathcal{U} \quad (2.22)$$

[4, 133]. If, however, binary variables are employed in the formulation, then the more complex maintenance window constraint set

$$\sum_{p \in \mathcal{P}_u} x_{u,p} = 1, \quad u \in \mathcal{U} \quad (2.23)$$

is required, in addition to the constraint sets

$$x_{u,p} = 0, \quad u \in \mathcal{U}, p \notin \mathcal{P}_u, \quad (2.24)$$

$$y_{u,p} = 0, \quad u \in \mathcal{U}, p < e_u \text{ or } p > \ell_u + d_u - 1. \quad (2.25)$$

In (2.25), d_u denotes the duration of planned maintenance required for PGU $u \in \mathcal{U}$, as before.

The duration of planned maintenance is furthermore required to be contiguous. In order to ensure that this requirement is met, the constraint set

$$y_{u,p} = \begin{cases} 1, & \text{if } x_u \leq p \leq x_u + d_u - 1, \\ 0, & \text{for all other } p \end{cases} \quad (2.26)$$

is included in the model formulation when integer decision variables are used [57, 67, 68]. Once again, two more complicated sets of constraints are required in the case where binary variables are used in the formulation. The first is the maintenance duration constraint set and the second ensures contiguity of the planned maintenance period. The maintenance duration constraint set may be formulated as

$$\sum_{p \in \mathcal{P}} y_{u,p} = d_u, \quad u \in \mathcal{U} \quad (2.27)$$

[39, 185], while the constraint set ensuring contiguity of the planned maintenance period of duration d_u may be formulated as

$$y_{u,p} - y_{u,p-1} \leq x_{u,p}, \quad u \in \mathcal{U}, p \in \mathcal{P} \setminus \{1\}, \quad (2.28)$$

$$y_{u,1} \leq x_{u,1}, \quad u \in \mathcal{U} \quad (2.29)$$

[39, 185]. There are, however, some cases in the literature [135, 152, 168] where the maintenance duration constraint set (2.27) and the maintenance contiguity constraint set (2.28) are combined into a single constraint set. This combined constraint set is

$$\sum_{p=x_u}^{x_u+d_u-1} y_{u,p} = d_u, \quad u \in \mathcal{U}. \quad (2.30)$$

Instances may also be found in the literature where an undesirable nonlinear construct is employed in the formulation of the maintenance duration and contiguity constraints [133]. In such cases, the duration constraint set is the same as (2.27), whereas the contiguity constraint set is formulated as

$$\prod_{j=x_u}^{x_u+d_u-1} y_{u,p} = 1, \quad u \in \mathcal{U}. \quad (2.31)$$

Resource constraints may also be included in GMS model formulations. These constraints specify the maximum number of resources available during a given planning period, of which some pre-specified minimum amount of resources are required to schedule planned maintenance for any PGU. These constraints ensure that the maximum amount of resources available during a planning period is not exceeded by the amount of resources required to carry out planned maintenance during that period. The resources available in GMS problem instances are typically the number of maintenance personnel available to perform planned maintenance. The most basic formulation of a resource constraint set of this kind was formulated for the case where integer decision variables are employed in the model formulation and assumes that during each period of planned maintenance, the same amount of resources is required for any given PGU [7, 68, 87]. Let $f_{u,p}$ denote the amount of resources required in order to perform planned maintenance on PGU $u \in \mathcal{U}$ during planning period $p \in \mathcal{P}$. Then the resource constraint set may be formulated as

$$\sum_{u \in \mathcal{U}} f_{u,p} y_{u,p} \leq M_p, \quad p \in \mathcal{P}, \quad (2.32)$$

where M_p denotes the maximum available amount of resources during planning period $p \in \mathcal{P}$. In the case where binary decision variables are employed in the formulation, a constraint set similar to (2.32) may be included in the model formulation [51, 53]. Following the same notation as in (2.19), the constraint set may be formulated as

$$\sum_{u \in \mathcal{U}'_p} \sum_{j \in \mathcal{S}'_{u,p}} f_{u,j} x_{u,j} \leq M_p, \quad p \in \mathcal{P}. \quad (2.33)$$

In more complex incarnations of the resource constraint set it is assumed that the resources required during each period of planned maintenance are not the same for some of the PGUs. This approach assumes that during the i -th planning period of planned maintenance, the PGU may require a specific amount of resources. Let f_u^i denote the amount of resources required for planned maintenance on PGU $u \in \mathcal{U}$ during its i -th period of its planned maintenance and let $f_{u,p,v}$ denote the resources required for planned maintenance on PGU $u \in \mathcal{U}$ during planning period $p \in \mathcal{P}$ if the maintenance were scheduled to start during planning period $v \in \mathcal{P}$. The resources required may then be calculated as

$$f_{u,p,v} = \begin{cases} f_u^{p-v+1}, & \text{if } p - v < d_u, \\ 0, & \text{otherwise.} \end{cases} \quad (2.34)$$

From (2.34), a more general resource constraint set may be formulated as

$$\sum_{u \in \mathcal{U}} \sum_{v \in \mathcal{P}} f_{u,p,v} x_{u,v} \leq M_p, \quad p \in \mathcal{P}. \quad (2.35)$$

There are also formulations of the GMS problem in the literature that contain so-called exclusion constraints. These constraints specify sets of PGUs which are not allowed to be in a state of simultaneous planned maintenance. The reason for this type of constraint is that PGUs from the same power station are often not allowed to be in maintenance during the same period. It may, for example, also be required that certain PGUs of the same class (*e.g.* coal, nuclear, wind, sun, *etc.*) should not be in a state of simultaneous planned maintenance. In order to achieve this type of constraint, let $\mathcal{J}_1, \dots, \mathcal{J}_w$ be sets of PGUs which may not all be scheduled simultaneously for planned maintenance and let I_i denote the maximum number of PGUs that are allowed to be scheduled for planned maintenance simultaneously during any planning period within exclusion set \mathcal{J}_i . Then the exclusion constraint set may be formulated as

$$\sum_{u \in \mathcal{J}_i} y_{u,p} \leq I_i, \quad p \in \mathcal{P}, \quad i \in \{1, \dots, w\} \quad (2.36)$$

[39, 51, 133]. In practice, it is also sometimes advantageous for a power utility to schedule planned maintenance of certain PGUs before certain other PGUs. This may be achieved by the incorporation of so-called *precedence constraints* which may either specify that some PGU u_1 has to be scheduled for planned maintenance and return to full operation before a certain PGU u_2 can be scheduled for planned maintenance, or that PGU u_2 is only allowed to enter a state of planned maintenance after planned maintenance has commenced on PGU u_1 . In the case where integer decision variables are employed in the model formulation, the former type of precedence constraint may be formulated as

$$x_{u_1} + d_{u_1} \leq x_{u_2}. \quad (2.37)$$

In the latter case, the precedence constraints may simply be formulated as

$$x_{u_1} < x_{u_2}. \quad (2.38)$$

In the case where binary decision variables are, however, employed in the model formulation, the former type of precedence constraint may be formulated as

$$\sum_{v \in \mathcal{P}} x_{i_1,v} - x_{i_2,p} \geq 0, \quad p \in \mathcal{P} \quad (2.39)$$

[39], while the latter type of precedence constraint may be formulated as

$$x_{i_1,p} + x_{i_2,p} \leq 1, \quad p \in \mathcal{P}. \quad (2.40)$$

2.3 GMS model solution approaches

Suitable GMS model solution techniques should be able to obtain good or, in some cases, optimal solutions to realistically sized model instances within a reasonable amount of computation time. The power systems of modern power utilities typically consist of large numbers of PGUs, sometimes exceeding 100 PGUs, of which each have to be scheduled for maintenance over planning periods varying in number from 52 (weeks) to 365 (days) within an annual scheduling window [166]. The number of planning periods and the number of PGUs both contribute to the complexity of the GMS model instance, which is typically NP-hard⁵ [81, 90]. This section contains a brief survey of the solution approaches adopted in the literature to solve GMS models.

2.3.1 Mathematical programming techniques

The GMS problem is of such a nature that it may easily be formulated as a mathematical problem, as demonstrated in §2.2. Many mathematical programming-related solution approaches may therefore be employed to solve *integer programming* (IP) or *mixed integer linear programming* (MILP) models of the GMS problem. Examples of such solution approaches include IP or MILP techniques based on the celebrated *branch-and-bound* (B&B) method, *dynamic programming* (DP) and a variety of *decomposition approaches* [5, 90]. IP techniques, such as Balas' algorithm⁶, are designed to find optimal solutions to IP problems, although a major disadvantage associated with the adoption of IP and MILP-related techniques is that they can only be used to solve relatively small problem instances to optimality within reasonable timeframes due to their high computational burdens [59]. Decomposition approaches have therefore been adopted to solve larger GMS problem instances [5, 90]. In the remainder of this section, the working of the B&B method, a decomposition method, and the method of dynamic programming are discussed in some detail.

The branch-and-bound method

The *branch-and-bound* (B&B) method was first introduced by Land and Doig [138] in 1960. This method finds an optimal solution to a combinatorial optimisation problem instance by a systematic enumeration of the candidate solutions in the feasible region to suitably selected subproblems. Each set of candidate solutions forms a rooted subtree of a tree-like data structure (called a *search tree*) representing the progression of the search for an optimal solution. The branches of these subtrees represent different, mutually exclusive subsets of the solution space. The algorithm compares solutions uncovered in the various branches of the search tree to global estimates of upper and lower bounds on the optimal solution objective function value. Branches are stored or discarded based on whether or not they contain improved solutions in respect of the best solution found so far in the tree [214].

This method is guaranteed to find optimal solutions to combinatorial optimisation problems. Let $f(\boldsymbol{\eta})$ be the objective function of a minimisation problem, where $\boldsymbol{\eta}$ is the vector of decision

⁵A problem is referred to as NP-hard (non-deterministic polynomial-time hard) if all other problems in the complexity class NP can be reduced to it in polynomial-time [130, 210]. Loosely speaking, this means that no polynomial time exact solution methodology is known for solving the problem, hence justifying the use of approximate techniques, such as heuristics or metaheuristics, for large instances of the problem.

⁶Balas' algorithm is used to solve constrained binary programs. It involves a systematic procedure which initialises with all the variables assigned the value 0 and then the algorithm assigns the value 1 successively to certain selected variables. The variables are selected in such a manner that either an optimal solution is found or evidence is found that no feasible solution exists [17].

variables which assume combinations of values within a set \mathcal{S} of candidate solutions, also referred to as the *feasible region* or the *search space*. As the name of the method suggests, there are two main procedures that are applied iteratively throughout the execution of the B&B method, namely *branching* and *bounding*. The branching procedure typically returns two or more mutually disjoint subsets $\mathcal{S}'_1, \mathcal{S}'_2, \dots$ of \mathcal{S}' , where \mathcal{S}' is a given subset of \mathcal{S} and where the union of the subsets $\mathcal{S}'_1, \mathcal{S}'_2, \dots$ form \mathcal{S}' . The minimum objective function value is then obtained for each subset in \mathcal{S}' . The second procedure in the method is called *bounding* which provides a method of calculating both lower and upper bounds on the objective function value $f(\boldsymbol{\eta})$ for each subset $\mathcal{S}' \subseteq \mathcal{S}$.

The function of the bounding procedure is to compare the lower and upper bounds of the various subsets of the feasible region. The B&B method works in such a manner that a subset may be discarded (removed from the search) if its lower bound is found to be greater than the upper bound of any other subset in the tree. The process of discarding subsets in the B&B method is referred to as *pruning* and its function is to decrease the number of candidate solutions which have to be considered explicitly during the search. The branching procedure is terminated based on two criteria. The first criterion is when a subset \mathcal{S}'_i is reduced to contain a single element, while the second criterion is when the upper and lower bound on $f(\boldsymbol{\eta})$ are equal over the search space \mathcal{S} .

The B&B method is often used to solve IPs or MILPs such as the GMS problem presented in §2.2. In such a case, the B&B method is initiated by solving the *linear programming* (LP) relaxation⁷ of the original IP or MILP. When solving the LP relaxation of the IP or MILP, two outcomes are possible. The first outcome may yield an optimal solution comprising integer values for all the variables restricted to be integer (these variables are called *integer variables*) in the IP or MILP problem, in which case the LP relaxation optimal solution is also an IP or MILP optimal solution. The second outcome may yield an optimal solution comprising non-integer values for one or more of the variables constrained to be integer in the IP or MILP problem. If this is the case, the branching procedure is performed on the integer variables with non-integer values in the LP relaxation optimal solution. This process is repeated until the optimal solution comprises integer values for all the integer variables.

Each time the branching procedure is carried out, the problem is partitioned into a number of smaller subproblems, each with a smaller feasible solution space represented by the nodes in the search tree. Consider an integer variable η which has a non-integer value q . The branching procedure leads to two separate subsets of the decision space where the one node (*i.e.* subset) represents that part of the decision space where $\eta \leq \lfloor q \rfloor$ and the other node that part where $\eta \geq \lceil q \rceil$. The inequalities introduced by the branching procedure are included as constraints in each of the respective subproblems.

During the bounding procedure, a lower bound on the objective function value $f(\boldsymbol{\eta})$ is computed for the IP or MILP, by solving the LP relaxation of the particular subset of the solution space \mathcal{S}'_i (*i.e.* the subproblem). The protocol in which the next subproblem is selected for consideration is important, because it has to take into account the number of nodes already explored during the search process as well as the memory capacity of the computer on which the method is executed. Three main methods may be employed to select the next subproblem, namely a *best-first search strategy*, a *breadth-first search strategy* and a *depth-first search strategy* [118]. According to the best-first search strategy, the untermiated node with the largest lower bound on the objective function value is selected to branch from. According to the breadth-first search strategy, branching is executed from all the nodes on one level of the search tree before considering

⁷The LP relaxation of an IP or MILP is obtained upon the removal of any binary or integer restrictions on the decision variables in the formulation [218].

any nodes on the next level in the search tree. The depth-first search strategy is similar to the breadth-first search strategy, but nodes in the tree with the largest level are instead branched from first. The best-first search strategy is typically used since it often minimises the number of redundant nodes that are visited during the search [118].

Branching from a node is terminated once the solution to the LP relaxation of that node contains only integer values for the integer variables, since this solution is a feasible solution to the IP or MILP as well. Branching from a node in the search tree is also terminated when it results in a lower bound on the objective function that is no better than the best feasible solution found up to that point during the search. Nodes that are terminated for this reason are called *fathomed* nodes [118], since any solutions uncovered by branching from such nodes will not lead to nodes with better objective function values than the best uncovered objective function value up to that point during the search. The branching procedure terminates once all the nodes are fathomed or once the best lower bound on the objective function value is no better than the best feasible solution found up to that point.

In the literature, the B&B method has been used in a number of cases to solve GMS problem instances in which a single objective function is considered [13, 21, 59, 64, 88, 132, 163]. Some attempts have also been made to solve GMS problem instances involving more than one objective simultaneously using the B&B method [134, 144, 166].

A decomposition approach

Benders decomposition is a method by which a large problem is decomposed (partitioned) into a number of smaller problems of similar structure. After the problem has thus been decomposed, the smaller problems are easier to solve (require much shorter times to solve). This property renders Benders decomposition a good solution approach when attempting to solve large combinatorial optimisation problems [39]. Benders decomposition is therefore a good solution approach for large instances of the GMS problem.

In order to employ a Benders' decomposition approach, the optimisation problem under consideration has to be formulated in a certain way. In this formulation the objective is to

$$\left. \begin{array}{ll} \text{minimise} & \mathbf{c}^T \boldsymbol{\eta} + \mathbf{f}^T \boldsymbol{\omega} \\ \text{subject to} & \mathbf{A}\boldsymbol{\eta} + \mathbf{B}\boldsymbol{\omega} = \mathbf{b}, \\ & \boldsymbol{\eta} \geq \mathbf{0}, \\ & \boldsymbol{\omega} \in \mathbf{Y} \subseteq \mathbb{R}^q, \end{array} \right\} \quad (2.41)$$

where q denotes the number of variables in the decision variable vector $\boldsymbol{\omega}$ and \mathbb{R}^q denotes the subset of vectors containing q real-valued elements [153]. For an MILP, \mathbb{R}^q is replaced by \mathbb{Z}^q , denoting the subset of vectors containing q integer-valued elements. When a Benders decomposition approach is employed, the variables are employed to partition the problem into multiple smaller problems. In these subproblems, the presence of \mathbf{y} -variables increase the level of solution difficulty. The problem therefore becomes easier to solve if these variables are fixed. The original problem is subsequently partitioned into one master problem, which contains the $\boldsymbol{\omega}$ -variables together with a small number of constraints, and a number of subproblems, each containing the $\boldsymbol{\eta}$ -variables. The master problem may be considered a relaxation of the original problem as it contains fewer constraints. The master problem is solved (*i.e.* obtaining values for the $\boldsymbol{\omega}$ -variables) which are, in turn, used to solve the subproblems (LPs containing only $\boldsymbol{\eta}$ -

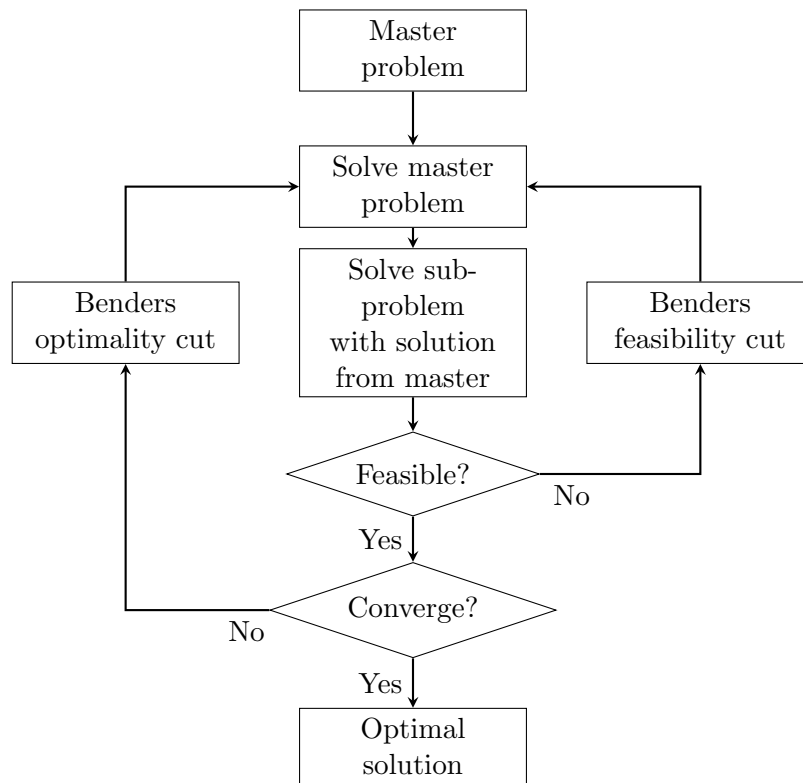


FIGURE 2.1: The process followed in a Benders decomposition solution approach (adapted from [93]).

variables). The preliminary solution to the master problem yields a lower bound on the original objective function value of the problem [200].

The solution to the master problem may, however, result in an infeasible solution when solving the subproblems. In such a case, a constraint is added to the master problem, referred to as a *Benders feasibility cut*. If the solution to the master problem provides a feasible solution, but the solution is not significantly close to the current lower bound of the objective function value, a constraint is again added to the master problem, which is referred to as *Benders optimality cut*. An iterative approach is adopted, following the method described above, to solve the master problem which satisfies all the constraints and is satisfactorily close to the lower bound [153]. A finite number of iterations are required for this method to converge to an optimal solution as there are only a finite number of Benders cuts that can be performed and these cuts are generated during each iteration of the algorithm [200]. A flow diagram illustrating the process of Benders decomposition is presented in Figure 2.1.

In the literature, Benders decomposition has been used in a number of cases to solve instances of GMS models incorporating a single, linear objective function [39, 95, 149, 152, 153, 221]. The approach may, however, also be used in a slightly adapted form in cases where the master problem is nonlinear and this alternative has indeed also been employed to solve model instances of the GMS problem in which nonlinear objective functions occur [51].

Dynamic programming

DP is a methodology according to which a large, unwieldy, temporal decision problem is partitioned into a reasonable number of smaller, overlapping subproblems which are easier to solve. These smaller subproblems are solved to optimality and the solutions to these subproblems are

then used to construct a solution to the larger problem. The number of subproblems should be enough to break the larger problem down sufficiently so that each subproblem can be solved individually within a reasonable time frame when compared to the time required to solve the original, large problem [29]. The method reduces the computation time by storing the solutions to the subproblems and recalling these solutions once a similar subproblem is investigated.

The GMS problem has to be formulated in a slightly different manner than the models in §2.2 in order to be solvable by DP. More specifically, the problem has to be formulated so that it contains a control vector and state vector. Define the control vector $\mathbf{U}(j) = [u_1(j) \ u_2(j) \ \dots \ u_n(j)]$ where $u_i(j)$ takes the value 1 if PGU $i \in \mathcal{I}$ is scheduled for planned maintenance during period $j \in \mathcal{J}$, or 0 otherwise. Furthermore, define the state vector $\mathbf{X}(j) = [x_1(j) \ x_2(j) \ \dots \ x_i(j)]$ where $x_i(j)$ denotes the completion degree of maintenance of PGU $i \in \mathcal{I}$ at the beginning of period $j \in \mathcal{J}$. In addition, let the recursion relation used to express the maintenance scheduling process of the GMS problem be

$$\mathbf{X}(j+1) = \mathbf{U}(j) + \mathbf{X}(j), \quad j \in \mathcal{J}. \quad (2.42)$$

In order for this recursive relation to be valid, boundary conditions of the form

$$\mathbf{X}(1) = \mathbf{0} \quad \text{and} \quad \mathbf{X}(m+1) = [d_1 \ d_2 \ \dots \ d_i] \quad (2.43)$$

have to be defined, where m again denotes the number of maintenance periods within the planning window and d_i denotes duration of planned maintenance of PGU $i \in \mathcal{I}$.

For large problems, the method of DP may be computationally very expensive as it contains a large number of control and state variables. The standard DP method is seldom applied directly to solve a realistically sized GMS model instances. This has led to an adaptation of the DP method, called *dynamic programming with successive approximations* (DPSA) [5], which reduces the dimensionality of an instance. The disadvantage of using the adapted method, however, is that while it may converge to a solution faster than the original DP approach (*i.e.* lead to a reduction in computation time), it may not guarantee a globally optimal solution.

The manner in which the DPSA approach is employed to solve GMS problem instances involves obtaining a trade-off between scheduling PGUs for maintenance simultaneously and scheduling the PGUs sequentially according to a predefined priority list⁸. During each iteration of the method, a sequence of DPs is solved for some subset of the PGUs. The reason for considering a subset of PGUs during each iteration is that the number of state variables is thus reduced for each subproblem, which results in easier subproblems which may be solved faster.

The method is initialised by providing it with an initial maintenance schedule. During each iteration, a small subset of PGUs is considered and the problem is solved for this subset. The PGUs that are not present in the subset do not experience any changes in their current maintenance schedules. The algorithm is terminated once two consecutive solutions achieve objective function values that are not significantly different from each other [135].

In the literature, the DPSA method has been employed in a number of studies to solve single-objective GMS model instances [144, 219, 226].

2.3.2 Expert systems

An *expert system* (ES) may be defined as a database of gathered expert knowledge in respect of a certain application field which is used by a software program to make informed decisions or assist in decision making [5]. By making use of knowledge gained by GMS experts (who

⁸A list of PGUs stating the priority in terms of when planned maintenance should be performed for each PGU.

have typically worked in the field for a long time and have experienced these problems first hand), an ES can sometimes be used to solve instances of the GMS problem and account for certain uncertainties and complicated scheduling factors that other approaches are not able to accomplish. In order to ensure that an ES is valid, however, accurate and precise expert rules have to be set in place to form a knowledge base. In practice, most of these rules are based on heuristic assumptions and are typically used to inform decisions when scheduling planned maintenance for PGUs [5, 107].

The ES approach is not a popular approach toward solving GMS model instances on its own. It is mostly implemented in conjunction with other methods in order to improve the quality of the final solutions thus obtained. An example of such a combined approach was proposed by Lin *et al.* [144], where an ES was combined with the B&B method and DP. An indexing system was developed with the valuable input from field experts which specifies which objective function (*i.e.* to minimise production cost or to maximise the minimum reserve margin) should be adopted in order to find good maintenance schedules. The index takes into account the load required during a certain scheduling window as well as the available capacity in order to specify which objective function should be optimised [144]. The index can specify one of three cases which correspond to an objective function that should be adopted in order to solve the problem. The objective of maximising the minimum reserve margin is pursued using the B&B method, whereas the objective of minimising the production cost is pursued by DP.

2.3.3 Fuzzy logic approaches

Fuzzy solution approaches reside within a paradigm of many-valued logic which deals with approximates. Rather than producing binary (*i.e.* true or false) answers, these methods provide answers that represent a certain degree of truth. A degree of truth is typically assigned to an answer based on a *membership function* which ranges from zero to one rather than adopting a binary membership function [222]. Solutions to GMS model instances may be obtained by the interaction between fuzzy sets which represent the objective function and the constraints of the model [164]. Fuzzy solution approaches have been shown to be capable of providing good solutions to GMS model instances within acceptable computation time frames [52, 110].

The degree of truth involved in a fuzzy logic approach is used to categorise the outcomes of a fuzzy set. This categorisation is performed based on the membership function, which is essential to fuzzy logic. Fuzzy logic may therefore be seen as an extension of the more conventional *boolean logic* where an element either belongs to a certain set or not. Let \mathcal{Y}_S be a function that characterises the set S in a boolean logic paradigm. Then

$$\mathcal{Y}_S = \begin{cases} 1, & \text{if } x \in S, \\ 0, & \text{if } x \notin S, \end{cases} \quad (2.44)$$

where the element x either belongs to the set S (*i.e.* true) or not (*i.e.* false). In the case of fuzzy logic, however, the element x can belong to a given set to a certain extent or according to a degree of truth. The degree of truth is determined by a fuzzy set's membership function, denoted by μ_S . The membership function μ_S may then be defined as

$$0 \leq \mu_S \leq 1, \quad \text{for any } x, \quad (2.45)$$

where μ_S represents the degree of truth associated with the statement that the element x belongs to the set S . The outcome of the membership function is then analysed and the set to which the element should eventually be assigned is selected subjectively by human referees [148]. It

may, for example, be agreed that a certain percentage of human referees have to agree on a set before the element is finally assigned to that set.

A fuzzy number may be defined as a one-dimensional fuzzy set where there is at least one value of μ_S that is equal to one for $x \in P(s)$. Another requirement is that there should exist an α -level subset (containing only definitely true elements of $P(s)$) associated with the fuzzy set S such that $\mu_S(x) = \alpha$. The intervals of these α -level subsets should decrease monotonically as the value α tends towards 1 [148].

The approach taken to solve a GMS model instance according to a fuzzy logic approach is that membership functions are obtained for the objectives and the soft constraints of the model instance. The GMS problem contains a fair amount of uncertainty and adopting a fuzzy logic approach alleviates the difficulties associated with the mathematical treatment of some of these uncertainties. A fuzzified GMS model instance is then typically solved by means of a fuzzy DP technique or with the help of a metaheuristic. In the literature, the fuzzy logic approach has been employed in a number of approaches toward solving instances of the GMS problem based on either a single objective function [52, 69, 161] or on multiple objective functions [110, 111].

2.3.4 Heuristics

The word *heuristic* derives from the Greek word *heuristikein*, which means to *find* or to *uncover* [217]. Heuristic searches aim to uncover not necessarily optimal solutions, but at least solutions of acceptable quality, by iteratively applying a trial-and-error solution approach over the course of a certain time interval. These methods are relatively simple to implement and require fairly short computation times, but they typically yield solutions of relatively poor quality. Heuristic approaches toward solving GMS models usually involve considering each PGU separately and scheduling these PGUs sequentially as per a pre-specified scheduling order within a given scheduling window. The most commonly applied mechanism in these heuristics involves equalising the system's risk and/or its net reserves [94].

More generally, a heuristic approach typically requires a number of rules by which the PGUs have to be scheduled sequentially. The constraints of the problem instance are ranked in order of nonincreasing importance and during the scheduling of PGUs the aim is to ensure that no constraints are violated. If this is not possible, a schedule is developed which violates a small number of the least important constraints. The manner in which the PGUs are ranked (*i.e.* the priority list) depends on the requirements of the power system [94]. In the most basic cases, the PGUs are only ranked according to their rated capacities and are scheduled accordingly. In some cases, however, PGUs are first ranked according to the type of unit (*i.e.* coal, hydro, wind, *etc.*) and then, within each group type, the PGUs are ranked according to their maintenance duration. If any of the PGUs require the same maintenance durations, they are ranked according to their rated capacities [46]. This ranked list is then used to schedule planned maintenance sequentially by violating the smallest number of constraints, if any.

In the literature, heuristic solution approaches, such as those described above, have been adopted in a number of cases to solve single-objective GMS model instances [27, 42, 46]. Heuristic solution approaches have also been combined with other solution methodologies in a number of cases in the GMS literature [32, 35, 43, 121].

2.3.5 Metaheuristics

The prefix *meta* derives from the Greek *above* or *beyond*. As the name suggests, metaheuristics are superior to heuristics in terms of their flexibility and their ability to yield better quality solutions than searches based on fixed heuristic rules. The class of metaheuristics may broadly be partitioned into two groups, namely *trajectory-based metaheuristics* and *population-based metaheuristics*. Trajectory-based metaheuristics trace out a succession of iteratively modified or transformed solutions to an optimisation problem over a number of iterations of the algorithm. These metaheuristics move from a single solution to another single solution through the search space in an iterative manner, based on a carefully selected move operator. Examples of trajectory-based metaheuristics include algorithms such as local search, the method of *simulated annealing* (SA), and *tabu search* (TS) [30]. Population-based metaheuristics, on the other hand, employ a population of solutions which are transformed simultaneously in an iterative fashion [199]. An initial population is created after which a new population is generated from that first population. The two populations are then integrated according to some selection procedure. This combined population is used to form a subsequent population, from where the procedure is repeated until some stopping criterion is met. Examples of population-based metaheuristics include scatter searches, evolutionary algorithms, and swarm optimisation techniques.

Metaheuristics most commonly employed to solve GMS model instances are *genetic algorithms* (GAs), SA, *particle swarm optimisation* (PSO), TS and *Ant colony optimisation* (ACO). A brief description follows in the remainder of this section of each of these algorithms in general as well as a description of these methods as they have been implemented to solve GMS model instances. Since the method of SA is applied later in this dissertation to solve two novel GMS model instances approximately, this method is described in more detail in the next section.

Genetic algorithms

A GA is a population-based metaheuristic which attempts to mimic biological evolution and natural selection processes in order to uncover good solutions [105]. Each candidate solution in the search space is referred to as an individual and each individual has a fitness value associated with it which is determined by the objective function. The fitness level of an individual is typically used in the selection of individuals for reproduction or replacement in the population. A *population* of individuals (solutions) is generated during each iteration, consisting of a subset of individuals that will be considered during that iteration of the algorithm. The population at any fixed iteration of a GA is referred to as a *generation*. These generations evolve over the iterations of the algorithm until the termination criteria of the algorithms are met [176].

A number of operators are employed during each iteration to produce a new population for the next generation. One such operator is the so-called *selection* operator, which determines which individuals of the population will be selected for reproduction. A *crossover* operator represents the sexual reproduction process according to which the characteristics of two individuals selected from the population for reproduction are combined to generate one or more offspring individuals. In order to steer the search through unexplored regions of the solution space, the GA also makes use of a *variation* or *mutation* operator which applies slight modifications to the individuals in the population in order to promote diversity among solutions. Typically, the selection process will occur first, followed by the crossover process and finally the mutation process (during each iteration of the algorithm).

In the literature, the GAs have been used in a number of studies to solve both single-objective GMS model instances [55, 213, 215] and model instances involving multiple objective functions

[22, 74]. GAs have also been combined with other solution methodologies within the GMS context in a number of cases in the literature [37, 52, 53, 69, 111, 141, 162].

Particle swarm optimisation

Another population-based optimisation technique is PSO, which mimics the social behaviour of a school of fish or a flock of birds [106]. The PSO algorithm is typically initialised by a random initial population of solutions. During each iteration of the algorithm, the population is updated based on candidate solutions to some optimisation problem instance found by other individuals in the population in their search for optimal solutions. Each solution at which an individual currently finds itself is referred to as a *particle*. The particles “move” through the solution space, but are biased in the direction of the currently best particles in the system. Each particle keeps track of its individual coordinates within the solution space as well as its associated fitness, as determined by the objective function. The particles also keep track of the coordinates associated with the best solution found by the entire swarm or flock of particles so far during the search and the direction of search is generally focussed in that direction.

Each individual particle is attributed its own velocity which is updated during each iteration so as to produce acceleration toward the location of good solutions. The change in the velocity of a particle is typically weighted according to its current location as well as relative to the best solution’s current location in the solution space. Hu [106] argued that PSO is able to achieve improved results in less time and requires less memory when compared to other population-based optimisation techniques, such as GAs or ACO, for certain problems. PSO also involves only a small number of parameters which require adjustment.

In the literature, the PSO approach has been used in a number of cases to solve single-objective GMS model instances [65, 131]. The PSO approach has also been combined with other solution methodologies within the GMS context in a number of cases in the literature [198, 220].

The method of tabu search

TS is a trajectory-based local search metaheuristic method developed by Glover in 1986 [97]. The TS approach focusses on avoiding the phenomenon of *cycling*. Cycling refers to the process where a certain algorithm keeps returning to a recently visited area within the solution space of an optimisation problem [176]. In TS, a short-term memory of recently visited areas within the search space is introduced. This short-term memory is maintained dynamically during the search and is referred to as the *tabu list* — areas or solutions that are part of the tabu list may not be revisited for a certain number of iterations in the TS algorithm after having visited them last [34, 96]. The TS approach considers all the neighbouring solutions of a current solution. This neighbourhood is found by perturbing the current solution locally. The best neighbouring solution is then selected as the new current solution (worsening solutions may be accepted if there are no improving solutions within the current solution’s neighbourhood). The search is, however, not allowed to return to any solution in the tabu list for a predetermined number of iterations (called the *tabu tenure*) so as to ensure that the algorithm does not become trapped at the locally optimal solution. This process is repeated iteratively until some stopping criterion is met. The number of iterations that must elapse before a solution is removed from the tabu list can be altered to suit the requirements of the algorithm. Keeping a solution in the tabu list for a small number of iterations promotes intensification of the search, where the focus will be on a certain area within the search space in which a good solution has been found, whereas keeping a solution in the tabu list for a large number of iterations promotes diversification of the search where new areas in the search space are explored which may contain good solutions [96].

In the literature, the method of TS has been used in a number of studies to solve single-objective GMS model instances [36, 66]. The TS approach has also been combined with other solution methodologies within the GMS context in a number of cases in the literature [37, 44, 127].

Ant colony optimisation

ACO is a population-based metaheuristic that was proposed by Dorigo [60] in 1992. The method is based on the natural behaviour of and communication between ants when foraging for food. When ants search for food, they initially wander around seemingly randomly with no coordination between themselves and the rest of the colony, until a good food source is found. The ant that finds the food source then returns to the ant colony and releases a pheromone⁹ along the path from the food source to the ant colony. The pheromone level along the path depletes over time, which means that if the distance from the ant colony to the food source is long, the pheromone level will, on expectation, be low once other ants sense it. If, on the other hand, the distance between the ant colony and the food source is short, the pheromone level will be high when other ants sense it. These ants will then start following the same path to gather food. While other ants travel along this path they also release pheromones, thus strengthening the pheromone level along the path. Since ants select paths randomly with a bias toward paths with high pheromone levels, the search process results in a situation where the ant colony successfully discovers shorter paths to food sources from their nest. For this reason, the approach was initially developed to find shortest paths in graphs, but has since been adapted to solve various other optimisation problems such as assignment problems and scheduling problems [60, 87].

In the literature, the ACO approach has been applied in a number of studies to solve single-objective GMS model instances [83, 87, 170]. The ACO approach has also been combined with other solution methodologies within the GMS context in a number of cases in the literature [44, 81].

2.3.6 Recent developments

Recently, some of the focus of GMS research has shifted from regulated systems (a single power utility providing energy to an entire country and the government of that country ensuring that the power utility does not take advantage of the end user) to deregulated systems (many power utilities are able to produce and distribute energy and hence competition replaces monopolies) [90]. The majority of GMS problems in deregulated systems either involve cost-based or profit-based scheduling criteria as objective function.

Bisanovix *et al.* [28] presented a comprehensive approach for maintenance scheduling of PGUs in a competitive market. The model takes into account long-term contracts with predefined power profiles and energy prices are forecasted weekly. Similar approaches have been adopted elsewhere with the aim of modelling coordination procedures for an independent system operator. This approach usually employs a game theoretic framework for the GMS problem [160]. Some recent papers have also incorporated the loss of power utility reputation and consumer loyalty towards a power utility as costs in the optimisation model. This is typically achieved by employing the Analytical Hierarchy Process to incorporate these costs as part of the opportunity cost in the model [56]. In 2017, Mazidi *et al.* [155] proposed a bi-level model which seeks to maximise the profit of the power utility, taking into account the reliability of the system as well as the cost limits of a society.

⁹A pheromone is a chemical substance that an ant produces and releases in order to communicate indirectly with other ants in an ant colony [120].

Although the majority of deregulated systems are focused on cost-based and profit-based GMS criteria, there have also been a number of papers considering reliability-based scheduling objectives [90]. For instance, Elyas *et al.* [70] argued that a power utility can be more competitive by being more reliable and therefore proposed two GMS objective functions. The first objective seeks to maximise annual social welfare while the second maximises producers' benefits and reliability of the power grid. In some cases, cost-based, profit-based and reliability-based GMS criteria have been combined to propose multi-objective models for finding good maintenance schedules, as proposed, for example, by Zhan *et al.* [224]. The objective functions proposed as part of these models typically maximise producer profit, maximise system reliability and/or minimise generating cost.

2.4 The method of simulated annealing

The method of SA has been employed successfully in the literature to solve various classes of complex optimisation problems. The SA approach has specifically also been used in a number of cases to solve GMS model instances based on a single objective function. In 1991, Satoh and Nara [187], implemented the method of SA to solve a GMS model instance with the objective of minimising the sum total of production cost and maintenance cost. Saraiva *et al.* [185] formulated an MILP GMS model with the aim of minimising the operation cost over all the planning periods, and solved this model by means of SA by penalising unsupplied energy by a monetary value. Schlünz and van Vuuren [188, 189] presented an adapted benchmark system, called the IEEE-RTS, to which SA was applied in order to find good solutions in terms of minimising the sum of squared reserves. The SA approach has also been combined with other solution methodologies within the context of GMS, such as evolutionary techniques [37, 53, 141, 162] and constraint programming [98].

The method of SA was first introduced in 1983 by Kirkpatrick *et al.* [129] as a trajectory-based metaheuristic for solving optimisation problems and is based on the physical process of annealing in metallurgy. It was initially introduced as an approximate solution methodology for single-objective optimisation problems but has since been adapted for use in the context of multi-objective optimisation problems as well [18, 50, 197]. The method is described here in the original single-objective context. The search consists of iterative application of a modification process called the Metropolis algorithm and requires the following inputs:

- an initial solution,
- an initial temperature,
- an appropriate cooling schedule,
- an epoch¹⁰ management strategy,
- a neighbourhood move operator, and
- end-of-search termination criteria.

The algorithm is initialised with an initial temperature T_0 and an initial solution $\boldsymbol{\eta}_0$. During each iteration of the search, the current solution $\boldsymbol{\eta}_i$ is modified by a selection

$$\boldsymbol{\eta}' \in N(\boldsymbol{\eta}_i), \quad (2.46)$$

¹⁰A pre-specified number of iterations which together form a search stage.

where $N(\boldsymbol{\eta}_i)$ denotes the *neighbourhood* of $\boldsymbol{\eta}_i$. This neighbourhood of a solution $\boldsymbol{\eta}$ is populated by exhaustively applying a so-called *move operator* to $\boldsymbol{\eta}$ which entails perturbing $\boldsymbol{\eta}$ locally. A randomly selected neighbouring solution $\boldsymbol{\eta}'$ is then compared to the current solution $\boldsymbol{\eta}_i$. If $\boldsymbol{\eta}'$ achieves an improvement in the objective function, which is also called the *energy* of the system, when compared to that of the corresponding value of $\boldsymbol{\eta}_i$, the neighbouring solution is accepted with a probability of 1 (*i.e.* with certainty) and becomes the current new solution, *i.e.* $\boldsymbol{\eta}_{i+1} \leftarrow \boldsymbol{\eta}'$. If, however, the neighbouring solution's objective function value does not achieve an improvement over that of the current solution, the neighbouring solution is accepted with probability $\exp(-\Delta E(\boldsymbol{\eta}_i, \boldsymbol{\eta}')/T_k)$; *i.e.* $\boldsymbol{\eta}_{i+1} \leftarrow \boldsymbol{\eta}'$ if $q < \exp(-\Delta E(\boldsymbol{\eta}_i, \boldsymbol{\eta}')/T_k)$, where q is a uniformly distributed random variable on the unit interval, ΔE is the change in the energy and T_k is the temperature of the search during the current epoch, k . The reason behind allowing the algorithm to accept a worsening solution with a certain probability is that it enables the algorithm to escape local optima. At a high temperature, the algorithm will accept the majority of worsening solutions, because then $\exp(-\Delta E(\boldsymbol{\eta}, \boldsymbol{\eta}')/T_k)$ is close to 1. As the temperature decreases, however, $\exp(-\Delta E(\boldsymbol{\eta}, \boldsymbol{\eta}')/T_k)$ tends to zero, which is expected to result in fewer worsening solutions being accepted. For this reason, the SA algorithm should have a high initial temperature which will enable the algorithm initially to explore as many solutions in the solution space as possible, whereafter the algorithm is expected to converge to a locally optimal solution, or possibly a globally optimal solution, as the temperature decreases according to the cooling schedule in use.

The two most commonly adopted cooling schedules in the literature on the GMS problem is an adaptive schedule proposed by Van Laarhoven and Aarts [211] and the more rigid *geometric schedule*. In both these schedules, a constant temperature is maintained for the entire epoch. The temperature of the search is decreased at the end of each epoch, and the durations of these epochs are determined dynamically during the search so as to promote metaheuristic flexibility. According to Dreo *et al.* [62], a general rule of thumb which may be used to terminate an epoch is when $12N$ solutions have been accepted or when $100N$ solutions have been attempted during the epoch, where N denotes a measure of the number of degrees of freedom of the search. The maximum length of an epoch is typically proportional to the number of possible neighbouring solutions, which is a measure of the number of degrees of freedom [189].

According to the cooling schedule proposed by Van Laarhoven and Aarts [211],

$$T_{k+1} = T_k \left(\frac{1}{1 + \frac{\ln(1+\delta)}{3\sigma_k} T_k} \right), \quad k = 0, 1, 2, \dots, \quad (2.47)$$

where δ is a small real number and σ_k is the standard deviation of the change in objective function at the end of epoch k . This schedule has proved to be very successful in the context of GMS problems, but is computationally expensive and time consuming to implement [1, 189].

The second cooling schedule is the geometric schedule which is more commonly used in practice due to the simplicity of implementing it and the rapid convergence to good solutions often obtained [2, 189, 205]. According to the geometric schedule,

$$T_{k+1} = \alpha T_k, \quad k = 0, 1, 2, \dots, \quad (2.48)$$

where α is called the *cooling parameter* and is typically assigned a value between 0.5 and 0.99 [126]. Apart from the two cooling schedules mentioned above, there are also other cooling schedules in the literature which are not used as often. One such cooling schedule is that of Huang *et al.* [109] according to which

$$T_{k+1} = T_k \exp\left(-\frac{\lambda T_k}{\sigma_k}\right), \quad k = 0, 1, 2, \dots, \quad (2.49)$$

where λ is a parameter that is typically assigned the value 0.7 [109, 189]. Another cooling schedule is the schedule proposed by Triki *et al.* [205] in 2005 according to which

$$T_{k+1} = T_k \left(1 - T_k \frac{\Delta_k}{\sigma_k^2} \right), \quad k = 0, 1, 2, \dots, \quad (2.50)$$

where Δ_k denotes the decrease in the mean objective function value during epoch k and σ_k is again the standard deviation of the change in objective function during epoch k .

In 1987, van Laarhoven *et al.* [211] presented a method for calculating a problem-specific initial temperature T_0 for the method of SA. The initial temperature is defined as $T_0 = -\overline{\Delta E^{(+)}}/\chi_0$, where $\overline{\Delta E^{(+)}}$ is the average increase in the objective function value (*i.e.* energy) of the system for a minimisation problem (for maximisation it would be $\overline{\Delta E^{(-)}}$, the average decrease in the objective function value) and where χ_0 is the so-called *acceptance ratio*. This ratio is defined as the number of accepted solutions which exhibit a deterioration in the objective function value divided by the number of attempted solutions which exhibit a deterioration in the objective function value. The ratio is typically set to a value between 0.5 and 0.8, as suggested by Buseti [38]. The average increase in the objective function value $\overline{\Delta E^{(+)}}$ is determined during a random walk over a certain pre-specified length in the solution space, using the same initial solution as the starting point of the random walk as would be used during execution of the SA algorithm.

Buseti [38] stated that the number of iterations spent in a single epoch k should be problem-dependent, rather than being a function of the epoch. A Markov chain of length L_k may be used to determine the number of iterations within epoch k . Ideally, one would require a pre-specified parameter A_{min} which specifies the minimum number of move acceptances during epoch k before lowering the temperature and moving on to the next epoch. As T_k approaches zero, the probability of accepting non-improving solutions decreases which, in turn, increases the number of iterations required to achieve the minimum number A_{min} of move acceptances per epoch. Therefore, an epoch will be terminated once a certain number of moves L have been attempted by reheating the temperature of the system to the initial temperature or decreasing the temperature according to the cooling schedule once A_{min} moves have been accepted, where A_{min} and L are pre-specified parameters satisfying $A_{min} < L$. Dreo *et al.* [62] proposed that $A_{min} = 12N$ and $L = 100N$, where N is a measure of the number of degrees of freedom of the optimisation problem under consideration.

During execution of the SA algorithm, the nondominated solution¹¹ is archived. The initial solution is the first entry in the archive. If a neighbouring solution of the current solution is not dominated by the current archived solution, the neighbouring solution is archived and the previously archived solution is removed from the archive. The algorithm is typically initialised by an initial solution to a GMS problem instance by selecting a random starting time, according to a uniform distribution, for planned maintenance on PGU $u \in \mathcal{U}$ within the scheduling window of the problem. When selecting such a random starting time, the algorithm takes into account the service duration required to complete maintenance on PGU $u \in \mathcal{U}$, but may violate the other soft constraints. This procedure is repeated for all PGUs in the power system.

The SA algorithm typically terminates when either one of two pre-specified states are reached within the system. The first is when the algorithm does not accept the minimum required acceptance number of $12N$ candidate solutions for three consecutive epochs. This state is referred to as the *frozen* state, since no more significant improvements can be made in the system. The second termination criterion for the SA algorithm is when a maximum number of iterations I_{max} have been performed. This number is usually selected as a large value so as

¹¹A solution obtained during the search for which no improving solution has yet been found [62].

to ensure that the system terminates due to a frozen state rather than reaching the maximum number of iterations which may terminate the algorithm prematurely. This second termination criterion is introduced to limit the operating time of the system so as to ensure that the overall computation time is acceptable to the user.

2.5 Chapter summary

A review of the literature on GMS problems was provided in this chapter. In §2.1, a general description of the GMS problem was given in some detail, explaining the notions of a scheduling window and of scheduling resolution, as well as referring the typical scheduling criteria pertaining to GMS model formulations in the literature. Constraints applicable to GMS model formulations and other related energy problems found in the literature were also described. This was followed, in §2.2, by a discussion on the most popular GMS model formulations in the literature, citing typical objective functions for and constraints included in model incarnations of the GMS problem. GMS model solution approaches that have been adopted in the literature were described in §2.3, including mathematical programming techniques, expert systems, fuzzy solution approaches, heuristics and finally metaheuristics. The approximate solution methodology that is employed later in this dissertation, namely the method of SA, was finally described in more detail in §2.4.

CHAPTER 3

Reliability theory

Contents

3.1	General considerations	41
3.2	Basic mathematical notions	42
3.3	Lifetime distribution models for non-repairable systems	44
3.4	Repairable systems	50
3.5	Life data classification	53
3.6	Trend tests	55
3.7	Model parameter estimation methods	58
3.8	Acceleration models	60
3.9	Chapter summary	62

This chapter contains a literature review on the research area of *reliability theory*. A number of general considerations in respect of reliability theory are presented, with a focus on the mathematical formulation of the notation of reliability. This is followed by a description of the most popular trend tests employed in practice to determine whether a system exhibits non-repairable or repairable characteristics. The chapter also includes mathematical treatments of the notion of reliability in the contexts of repairable systems and non-repairable systems, as well as popular models for describing the reliabilities of both types of systems.

3.1 General considerations

Survival theory is a subfield of general statistics concerned with the analysis of the durations between the occurrences of successive events in a system. The nature of these events may vary according to the type of system being analysed, and may include mechanical failures of machinery or illnesses of biological organisms [159]. The intersection between survival theory and engineering is known as *reliability theory* or *reliability analysis*, which is a general theory about system failures. Its constituent components are ideas, mathematical models and methods that can be used to estimate, predict, understand and optimise the lifespans of components as well as of systems as a whole [19]. The objective in reliability theory is usually to quantify a suitable trade-off between wasting a system's residual life and running the risk of unexpected failure of the system. The two main branches of reliability theory are theories that have been developed for repairable systems and for non-repairable systems [181]. A brief overview is given in this section of these two subtheories.

A system is classified as a *non-repairable system* when the population of items contained as components in the system is one for which individual items that fail are permanently removed from the population and the system is repaired by replacing these items with items from either the same or from a different population. A *repairable system*, on the other hand, may be defined as a system that can be restored to fully operational performance by any maintenance action other than replacing the entire system after failing to perform some of its intended functions [11].

The effective reliability of a system over a certain time frame during the system's lifetime can be determined based on historical failure data of the system. Historical failure data may exhibit an *increasing failure rate* (IFR), a *constant failure rate* (CFR) or a *decreasing failure rate* (DFR). A system exhibiting an IFR, also called an *ageing system*, generally comprises components that wear out over the lifetime of the system, which causes the time between consecutive failures to decrease. Systems with IFRs generally require the most attention with respect to planned maintenance, since failures in the system are observed more frequently towards the end of the system's lifetime [85]. Some systems exhibit CFRs, where the failures in the system are observed to be random with inter-failure rates that are exponentially distributed. These systems require less attention when planning maintenance due to the randomness between consecutive failures [23]. In the case where a system exhibits a DFR, the time between consecutive failures is increasing — hence fewer failures are observed towards the end of the system's lifetime — and so the system may be described as an *improving system* [3]. For improving systems, it might sometimes be harmful to conduct planned maintenance as the system naturally increases in reliability over time.

The *well-known bathtub curve*, presented in Figure 3.1, may be used to represent the failure rate of a system graphically as a function of the lifetime of the system. Typically, from the start of the lifetime of a system up to a certain time t_1 , the system exhibits an DFR. Between times t_1 and $t_2 > t_1$, the system is in its useful stage and exhibits a CFR. The final part of the graph, from t_2 onwards, represents the final part of the system's lifetime, which exhibits an IFR.

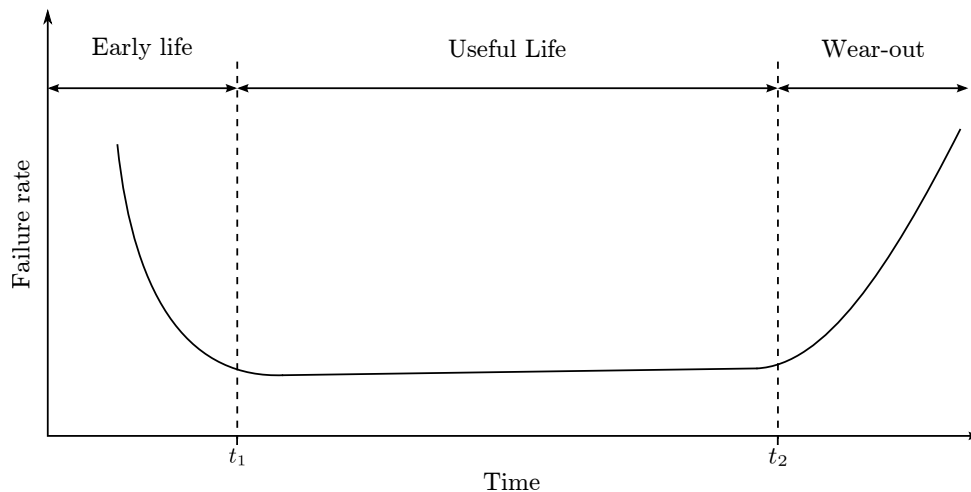


FIGURE 3.1: The failure rate of a system as a function of time, represented by the well-known bathtub curve [214].

3.2 Basic mathematical notions

Suppose $\{T_1, \dots, T_{n-1}\}$ is a set of $n - 1$ past failure times observed in a system, where $T_1 \leq T_2 \leq \dots \leq T_{n-1}$ are measured in units of global time, denoted by t . Then the reliability of the

system is defined as

$$R(t) = P(T_n > t), \quad (3.1)$$

where T_n is a continuous random variable denoting the n -th failure time of the system, also measured in units of global time. The system reliability at time t is therefore the probability that the n -th system failure will occur after time t [20]. It is clear that $R(t) = 1$ for all $t \leq T_{n-1}$. The complement of the reliability function is called the *lifetime distribution model*, or the *cumulative distribution function* (CDF) of T_n , also sometimes referred to as the *theoretical population model* describing the lifetime of a component or system [167, 173]. The lifetime distribution function of T_n is denoted by

$$F(t) = P(T_n \leq t) = 1 - R(t). \quad (3.2)$$

From the lifetime distribution function, the *probability density function* (PDF) of T_n , denoted by f , may be calculated by taking the derivative of (3.2). Therefore, the CDF is given by

$$F(t) = \int_0^t f(\tau) d\tau. \quad (3.3)$$

The *mean time to failure* (MTTF) between the $(n-1)$ -th and n -th failures may be computed as the expected value

$$\mu = \int_{T_{n-1}}^{\infty} f(\tau) \cdot \tau d\tau.$$

The PDF $f(t)$ and the CDF $F(t)$ may be estimated based on the distribution of the past failure times T_1, \dots, T_{n-1} under the assumption that the past failure rate is an indication of the future failure rate. Substituting (3.3) into (3.2), it follows that

$$R(t) = 1 - \int_0^t f(\tau) d\tau = \int_t^{\infty} f(\tau) d\tau. \quad (3.4)$$

If the PDF $f(t)$ of T_n and therefore the reliability function $R(t)$ of the system is known, the *conditional failure rate* of the system, also known as its *hazard rate* or *force of mortality* (FOM) [204], is defined as

$$h(t) = \frac{f(t)}{R(t)} = \frac{f(t)}{1 - F(t)} \quad (3.5)$$

for all $t > T_{n-1}$. The hazard rate in (3.5) is referred to as a conditional failure rate, because $h(t)$ may be interpreted as the failure rate, given that the components in the system survive past time t [171]. A justification of this interpretation of the function $h(t)$ may be obtained by computing the conditional probability that the system will fail before a certain time $t + t_e$, given that it survives past time t [142]. This conditional probability is

$$\begin{aligned} P(T_n \leq t + t_e \mid T_n > t) &= \frac{P((T_n \leq t + t_e) \cap (T_n > t))}{P(T_n > t)} \\ &= \frac{1}{R(t)} \int_t^{t+t_e} f(\tau) d\tau. \end{aligned}$$

For a small, positive value of t_e , it follows by the continuity of $f(t)$ that

$$\frac{1}{R(t)} \int_t^{t+t_e} f(\tau) d\tau \approx \frac{t_e f(t)}{R(t)}. \quad (3.6)$$

During an interval of time, the number of failures expected to be observed in the system is

$$\text{interval duration} \times \text{failure rate} = t_e h(t). \quad (3.7)$$

Equating (3.6) and (3.7), it follows that

$$h(t) = \frac{f(t)}{R(t)},$$

as is noted in (3.5). The reliability of the system may be expressed in terms of the failure rate of the system by noting that

$$\begin{aligned} h(t) &= \frac{f(t)}{R(t)} \\ &= \frac{\frac{dF(t)}{dt}}{1 - F(t)} \\ &= -\frac{d}{dt}[\ln(1 - F(t))]. \end{aligned} \quad (3.8)$$

Integrating both sides of (3.8) over the interval 0 to t , it follows that

$$\begin{aligned} -\int_0^t h(\tau) d\tau &= \int_0^t \frac{d}{d\tau}[\ln(1 - F(\tau))] d\tau \\ &= \ln(1 - F(t)) \\ &= \ln R(t) \end{aligned} \quad (3.9)$$

by the fundamental theorem of the calculus. Finally, in order to express the system reliability in terms of the failure rate of the system, the logarithm may be removed by exponentiating both sides of (3.9) to obtain

$$R(t) = e^{-\int_0^t h(\tau) d\tau}. \quad (3.10)$$

If the function $h(t)$ in (3.10) is integrated, the *cumulative hazard function* $H(t) = \int_0^t h(\tau) d\tau$ is obtained, which describes the number of failures expected to have occurred by a certain time during the lifetime of the system [204].

3.3 Lifetime distribution models for non-repairable systems

The choice of an appropriate lifetime distribution model for a non-repairable system is usually determined by three main factors, namely (i) whether statistical or physical evidence exist that relates the lifetime distribution model to the failure mechanism on a theoretical basis, (ii) whether the lifetime distribution model has previously been used to represent a similar failure mechanism and has proved successful in that context, and (iii) whether the lifetime distribution model is flexible and convenient enough to fit the failure data empirically [167, 171]. Five of the most popular lifetime distribution models for non-repairable systems are described in this section. In each case, the corresponding PDF, CDF and hazard rate are described. The models reviewed are the exponential model, the Weibull model, the normal model, the lognormal model and the gamma model.

3.3.1 The exponential model

One of the most useful and widely used lifetime distribution models is the *exponential model* [204]. This is a very simple model with only one unknown parameter λ that has to be estimated. The PDF of the exponential lifetime distribution is

$$f(t) = \lambda e^{-\lambda t}, \quad (3.11)$$

where λ is a constant [171, 204].

The CDF of the exponential lifetime distribution model, derived using (3.3), is

$$F(t) = \int_0^t \lambda e^{-\lambda x} dx = 1 - e^{-\lambda t}.$$

The reliability of a system can now be calculated according to this model, using (3.4), as

$$R(t) = 1 - (1 - e^{-\lambda t}) = e^{-\lambda t}. \quad (3.12)$$

From (3.12), the system reliability may be calculated at any time instant t . With both $f(t)$ and $R(t)$ known, (3.5) may be used to calculate the hazard rate of the system as

$$h(t) = \frac{\lambda e^{-\lambda t}}{e^{-\lambda t}} = \lambda, \quad (3.13)$$

which results in an exponential failure rate that is reduced to the value λ for all time values t . The exponential distribution is the only lifetime distribution with a constant failure rate. The *mean time to failure* (MTTF) M for the exponential lifetime distribution may be calculated by integrating (3.4) by parts to obtain

$$M = \frac{1}{\lambda},$$

the reciprocal of the hazard rate λ . By (3.13), the expected number of failures to have occurred by a certain time t may be calculated as

$$H(t) = \int_0^t h(x) dx = \lambda t,$$

where t is the continuous global time of the system in question. In order to associate a confidence level with the expected failure time T_n , a confidence interval may be calculated. This is achieved by calculating confidence bands for the MTTF, using the chi-square distribution, as

$$\frac{2(n-1)M}{\chi_{1-\beta/2, 2(n-1)}^2} \leq \text{MTTF} \leq \frac{2(n-1)M}{\chi_{\beta/2, 2(n-1)}^2}, \quad (3.14)$$

where $\chi_{\beta/2, 2(n-1)}^2$ represents the chi-square statistic at a confidence level of β , with $2(n-1)$ degrees of freedom, where $n-1$ is the number of observed past failures. The confidence bands for the timing of the n -th failure may now be calculated using the upper and lower confidence limits for the MTTF calculated in (3.14), which yields the interval

$$M_\ell n \leq T_n \leq M_u n,$$

where M_ℓ and M_u are the lower and upper confidence limits for the MTTF, respectively, and T_n is the expected observation time of the n -th failure at a confidence level of β .

The exponential model is widely used in industry for the modelling of the flat part of the bathtub curve, shown in Figure 3.1, between the time instants t_1 and t_2 , due to the model having a constant failure rate [167, 173, 214]. This model is only applicable when early decreasing failures and late wear-out failures are not taken into account [171]. The exponential model is sometimes also used as proxy for other failure models by approximating an exponential piecewise function for the model under consideration.

3.3.2 The Weibull model

The *Weibull model* is another flexible lifetime distribution model and can be used in a wide range of reliability problems. The model has two parameters, namely a *scale parameter* α and a *shape parameter* γ . The PDF of the Weibull lifetime distribution is

$$f(t) = \frac{\gamma}{\alpha} \left(\frac{t}{\alpha}\right)^{\gamma-1} \exp\left[-\left(\frac{t}{\alpha}\right)^\gamma\right]. \quad (3.15)$$

The CDF of the Weibull lifetime distribution model, derived using (3.3), is

$$F(t) = \int_0^t \frac{\gamma}{\alpha} \left(\frac{x}{\alpha}\right)^{\gamma-1} \exp\left[-\left(\frac{x}{\alpha}\right)^\gamma\right] dx = 1 - \exp\left[-\left(\frac{t}{\alpha}\right)^\gamma\right].$$

The reliability of a system may be calculated according to the Weibull lifetime distribution model, using (3.4), as

$$R(t) = 1 - (1 - \exp\left[-\left(\frac{t}{\alpha}\right)^\gamma\right]) = \exp\left[-\left(\frac{t}{\alpha}\right)^\gamma\right]. \quad (3.16)$$

With both $f(t)$ and $R(t)$ known, (3.5) may be used to calculate the hazard rate of the system as

$$h(t) = \frac{\frac{\gamma}{\alpha} \left(\frac{t}{\alpha}\right)^{\gamma-1} \exp\left[-\left(\frac{t}{\alpha}\right)^\gamma\right]}{\exp\left[-\left(\frac{t}{\alpha}\right)^\gamma\right]} = \frac{\gamma}{\alpha} \left(\frac{t}{\alpha}\right)^{\gamma-1}. \quad (3.17)$$

The MTTF for the Weibull lifetime distribution model is

$$M = \alpha \Gamma\left(\frac{1}{\gamma} + 1\right),$$

where $\Gamma(\frac{1}{\gamma} + 1)$ is the well-known gamma function

$$\Gamma(c) = \int_0^\infty e^{-x} x^{c-1} dx \quad (3.18)$$

evaluated at $(\frac{1}{\gamma} + 1)$. The expected number of failures to have occurred by a certain time t may be calculated by (3.13) as

$$H(t) = \int_0^t h(x) dx = \left(\frac{t}{\alpha}\right)^\gamma.$$

In order to calculate a confidence band for the expected failure time T_n , confidence intervals must again be calculated. This is achieved in the exact same manner as for the exponential model, as explained in §3.3.1 — see (3.14) and (3.15).

A three-parameter Weibull model is also used in some cases in the literature, where the third parameter is called the *waiting parameter* and is denoted by μ . The set of equations for the three-parameter Weibull model is the same as for the two-parameter Weibull model, except that t is replaced by $t - \mu$ in every equation. The waiting parameter provides for the possibility of no failure occurring in the system for a certain duration μ .

The flexibility of the Weibull model has resulted in the model being applied successfully to a wide range of problems [173, 181]. The Weibull model has also been used as a form of extreme value analysis where the earliest failure time of many competing failures are determined in order to determine system failure.

3.3.3 The normal model

The *Gaussian distribution*, more commonly known as the *normal distribution*, is probably the most widely-used distribution in statistics. The normal distribution is therefore also adopted as a lifetime distribution model in reliability theory. This model, however, has a left-hand limit extending to negative infinity which is not realistic when modelling failure time data (as time values are normally restricted to non-negative values). For this reason the normal model has been argued by some to be inappropriate for use as a lifetime distribution model [180]. The negative left-hand limit may, however, largely be avoided provided that the distribution has a large mean and a relatively small standard deviation. The PDF for the normal model is given by

$$f(t) = \frac{1}{\sigma\sqrt{2\pi}} \exp \left[-\frac{1}{2} \left(\frac{t-\mu}{\sigma} \right)^2 \right], \quad (3.19)$$

where μ denotes the mean of the times to failure and σ denotes the standard deviation of the times to failure. The CDF of the normal lifetime distribution model is given by

$$F(t) = \int_{-\infty}^t \frac{1}{\sigma\sqrt{2\pi}} \exp \left[-\frac{1}{2} \left(\frac{x-\mu}{\sigma} \right)^2 \right] dx, \quad (3.20)$$

which may be approximated by

$$F(t) \approx \int_0^t \frac{1}{\sigma\sqrt{2\pi}} \exp \left[-\frac{1}{2} \left(\frac{x-\mu}{\sigma} \right)^2 \right] dx = \Phi(t).$$

This integral is the standard normal CDF, which may also be written as

$$\Phi(x) = \frac{1}{2} \left[1 + \operatorname{erf} \left(\frac{x-\mu}{\sigma\sqrt{2}} \right) \right]. \quad (3.21)$$

In (3.21), $\operatorname{erf}(x)$ is the well-known *error function* which represents the probability that a random variable with normal distribution of variance $\frac{1}{2}$ and mean 0 will fall in the range $[-x, x]$. This function may be evaluated as

$$\operatorname{erf}(x) = \frac{1}{\sqrt{\pi}} \int_{-x}^x e^{-y^2} dy. \quad (3.22)$$

The reliability of a system may now be calculated according to the normal lifetime distribution model, using (3.4), as

$$R(t) = 1 - \Phi \left(\frac{t-\mu}{\sigma} \right) = \int_t^{\infty} \frac{1}{\sigma\sqrt{2\pi}} \exp \left[-\frac{1}{2} \left(\frac{x-\mu}{\sigma} \right)^2 \right] dx. \quad (3.23)$$

With both $f(t)$ and $R(t)$ known, (3.5) may be used to calculate the hazard rate of the system as

$$h(t) = \frac{\frac{1}{\sigma\sqrt{2\pi}} \exp \left[-\frac{1}{2} \left(\frac{t-\mu}{\sigma} \right)^2 \right]}{\int_t^{\infty} \frac{1}{\sigma\sqrt{2\pi}} \exp \left[-\frac{1}{2} \left(\frac{x-\mu}{\sigma} \right)^2 \right] dx}. \quad (3.24)$$

The MTTF for the normal lifetime distribution model is simply the mean of the normal distribution, namely $M = \mu$. The expected number of failures to have occurred by a certain time t may be calculated by taking the integral of (3.24), *i.e.*

$$H(t) = \int_0^t h(t) dt = \int_0^t \frac{\frac{1}{\sigma\sqrt{2\pi}} \exp \left[-\frac{1}{2} \left(\frac{t-\mu}{\sigma} \right)^2 \right]}{\int_t^{\infty} \frac{1}{\sigma\sqrt{2\pi}} \exp \left[-\frac{1}{2} \left(\frac{x-\mu}{\sigma} \right)^2 \right] dx} dt.$$

In order to calculate a confidence band for the expected failure time T_n , confidence intervals must again be calculated. This is achieved in the exact same manner as for the exponential model, as described in §3.3.1 — see (3.14) and (3.15).

3.3.4 The lognormal model

As with the Weibull model, the lognormal model is mainly applied in two-parameter form. The lognormal distribution's two parameters are a *shape parameter* α , and a *median* T_{50} , which fulfil the role of a scale parameter. If a system exhibits a time to failure that follows a lognormal distribution, the natural logarithm of such a time to failure has a normal distribution. Therefore, if the natural logarithm of the failure times are taken, the data may be considered normally distributed with mean $\mu = \ln T_{50}$ and with standard deviation σ . After analysis, the failure times may be converted back from logarithmic time to normal time. The PDF for the lognormal model is given by

$$f(t) = \frac{1}{\sigma t \sqrt{2\pi}} \exp \left[- \left(\frac{1}{2\sigma^2} \right) (\ln t - \mu)^2 \right], \quad (3.25)$$

which is very similar to the PDF of the normal model in (3.19). The CDF of the lognormal lifetime distribution model, derived using (3.3), is given by

$$F(t) = \int_0^t \frac{1}{\sigma x \sqrt{2\pi}} \exp \left[- \left(\frac{1}{2\sigma^2} \right) (\ln x - \mu)^2 \right] dx = \Phi \left(\frac{\ln t - \mu}{\sigma} \right),$$

where Φ is again the standard normal CDF in (3.21). The reliability of a system may now be calculated according to the lognormal lifetime distribution model, using (3.4), as

$$R(t) = 1 - \Phi \left(\frac{\ln t - \mu}{\sigma} \right). \quad (3.26)$$

With both $f(t)$ and $R(t)$ known, (3.5) may be used to calculate the hazard rate of the system as

$$h(t) = \frac{\frac{1}{\sigma t \sqrt{2\pi}} \exp \left[- \left(\frac{1}{2\sigma^2} \right) (\ln t - \mu)^2 \right]}{1 - \Phi \left(\frac{\ln t - \mu}{\sigma} \right)}. \quad (3.27)$$

The MTTF for the lognormal lifetime distribution model is

$$M = T_{50} e^{\frac{1}{2}\sigma^2}.$$

The number of failures expected to have occurred by a certain time t may be calculated by taking the integral of (3.27), that is

$$H(t) = \int_0^t h(x) dx = \int_0^t \frac{\frac{1}{\sigma x \sqrt{2\pi}} \exp \left[- \left(\frac{1}{2\sigma^2} \right) (\ln x - \mu)^2 \right]}{1 - \Phi \left(\frac{\ln x - \mu}{\sigma} \right)} dx.$$

In order to calculate a confidence band for the expected failure time T_n , confidence intervals must again be calculated. This is achieved in exactly the same manner as for the exponential model, as explained in §3.3.1 — see (3.14) and (3.15).

As with the Weibull model, the lognormal model also has a three-parameter variation which is used in some cases, where the third parameter is again a *waiting parameter* θ . The set of equations for the three-parameter lognormal model is the same as those of the two-parameter

lognormal model, except that t is replaced by $t - \theta$ in every equation. The waiting parameter again provides for the possibility of no failure occurring in the system for a certain duration θ .

Taking the natural logarithm of the failure time data makes for a very convenient model, as the failure time data are now in normal form, which makes it mathematically easier to use. The lognormal model has been employed in the literature to model physical degradation in electronics for failures as a result of as corrosion, diffusion, crack growth, *etc.* [171, 173, 181].

3.3.5 The gamma model

The *Gamma distribution model*, referred to as the *Erlang distribution in queuing theory*, is also a flexible lifetime distribution model, but is not often used to model the lifetime for a failure model [167, 181]. The PDF for the Gamma lifetime distribution is

$$f(t) = \frac{t^{k-1} e^{-t/\theta}}{\theta^k \Gamma(k)} \quad \text{for } t, k, \theta > 0, \quad (3.28)$$

where k is a *shape parameter*, θ is a *scale parameter* and $\Gamma(k)$ is the gamma function in (3.18) evaluated at k . The CDF of the Gamma lifetime distribution model, derived using (3.3), is

$$F(t) = \int_0^t \frac{x^{k-1} e^{-x/\theta}}{\theta^k \Gamma(k)} dx = \frac{\gamma(k, \frac{t}{\theta})}{\Gamma(k)},$$

where $\gamma(k, \frac{t}{\theta})$ is the lower incomplete gamma function

$$\gamma(k, x) = \int_0^x y^{k-1} e^{-y} dy \quad (3.29)$$

evaluated at $(k, \frac{t}{\theta})$. The reliability of a system may now be calculated according to the gamma lifetime distribution model, using (3.4), as

$$R(t) = 1 - \frac{\gamma(k, \frac{t}{\theta})}{\Gamma(k)}. \quad (3.30)$$

With both $f(t)$ and $R(t)$ known, (3.5) can be used to calculate the hazard rate of the system as

$$h(t) = \frac{\frac{t^{k-1} e^{-t/\theta}}{\theta^k \Gamma(k)}}{1 - \frac{\gamma(k, \frac{t}{\theta})}{\Gamma(k)}} = \frac{t^{k-1} e^{-t/\theta}}{\theta^k \Gamma(k) \left(1 - \frac{\gamma(k, \frac{t}{\theta})}{\Gamma(k)}\right)} = \frac{t^{k-1} e^{-t/\theta}}{\theta^k \Gamma(k) - \theta^k \gamma(k, \frac{t}{\theta})}. \quad (3.31)$$

The MTTF for the Gamma lifetime distribution model is $M = k\theta$, whereas the expected number of failures to have occurred by a certain time t may be calculated from (3.13) as

$$H(t) = \int_0^t h(x) dx = k\theta t.$$

In order to calculate a confidence band for the expected failure time T_n , confidence intervals must be calculated. This is again achieved in the exact same manner as for the exponential model, as explained in §3.3.1 — see (3.14) and (3.15).

3.4 Repairable systems

The two types of maintenance performed on repairable systems are *condition maintenance* and *preventative maintenance*. Condition maintenance involves actions performed after a failure has occurred in the system and maintenance has to be performed in order to restore the system to functioning condition. Preventative maintenance, on the other hand, involves any action associated with maintenance performed on a system with the aim to prevent or delay a system failure [40]. The *rate of occurrence of failure* (ROCOF) or *repair rate* is defined as the rate at which failures occur during the lifetime of the system. Failure rates, on the other hand, are only applicable to the first failure in a non-repairable system; it is therefore incorrect to use this term in the context of repairable systems [171]. Time in a repairable system is measured as the number of time units during which the system has been in operation, from initial turn-on to the end of system life, also known as the global time. The system experiences failures as it ages and is repaired to operational condition after each failure [40, 171].

Repair models are either classified as *renewal models* or as *minimal repair models* [40]. In a renewal model, it is assumed that after each failure the system is repaired to an “as good as new” condition, whereas in a minimal repair model it is assumed that the system is only repaired to the state it was in just before the occurrence of the last failure. The latter type of model is more commonly used in very constrained systems [40]. The three main repair models used to represent repair rates in repairable systems are reviewed in this section. These models are the *Homogeneous poisson process* (HPP), the *Non-homogeneous poisson process* (NHPP) following an exponential law and the *NHPP following a power law*.

3.4.1 The Homogeneous poisson process

The HPP is recognised as one of the simplest models for representing and predicting failures in a repairable system [40, 167]. This model is widely used despite its simplicity and is justified by the shape of the Bathtub curve in Figure 3.1. Complex systems generally operate in the “Useful Life” portion of the Bathtub curve for most of the system’s lifetime during which a constant repair rate prevails. The HPP is one of the only models for repairable systems that applies to this portion of the Bathtub curve by exhibiting a constant repair rate [171, 173]. Therefore, the ROCOF in the HPP model is expressed by

$$h(t) = \lambda, \quad (3.32)$$

which does not depend on the lifetime of the system. The repair rate may be estimated by calculating the *mean time between failures*

$$\text{MTBF} = \frac{\text{Total system operating time}}{\text{Total number of observed failures}} = \frac{1}{\lambda} \quad (3.33)$$

for the system. The PDF of the basic HPP is the same as the PDF for the exponential model for non-repairable systems as presented in (3.11). The CDF in the basic HPP model is

$$F(t) = 1 - e^{-\lambda t}.$$

Therefore, the reliability of a system that is represented by an HPP model may be calculated as

$$R(t) = 1 - F(t) = e^{-\lambda t}. \quad (3.34)$$

From (3.32), which is similar to the exponential model for non-repairable systems described in §3.3.1, the expected number of failures to have occurred by a certain time t may be expressed as

$$H(t) = \int_0^t h(x) dx = \lambda t,$$

where t is the continuous global time of the system in question. In order to calculate confidence band for the expected failure time T_n , confidence intervals must be calculated. This is once again achieved in the exact same manner as for the exponential model, as explained in §3.3.1 — see (3.14) and (3.15).

3.4.2 The Non-homogeneous poisson process following an exponential law

The *NHPP model following an exponential law*, also known as the *log-linear model* or the *Cox-Lewis model*, has been employed successfully in a variety of applications [48, 167, 171, 173]. This model follows a ROCOF of the form

$$h(t) = e^{\alpha^0 + \alpha^1 t}, \quad (3.35)$$

where α^1 is positive for a repairable system and t represents continuous global time [136]. The number of failures expected to have occurred by a certain time t may be expressed according to the model by integrating (3.35), that is

$$H(t) = \int_0^t h(x) dx = \frac{e^{\hat{\alpha}^0} (e^{\hat{\alpha}^1 t} - 1)}{\hat{\alpha}^1}, \quad (3.36)$$

where $\hat{\alpha}^0$ and $\hat{\alpha}^1$ are the estimated values for α^0 and α^1 . From (3.36), it is possible to calculate the timing of the n -th failure by solving for t in the equation

$$H(t) = \frac{e^{\hat{\alpha}^0} (e^{\hat{\alpha}^1 t} - 1)}{\hat{\alpha}^1} = n, \quad (3.37)$$

where $n - 1$ failures have already been observed in the system. By using (3.10), it is possible to calculate the reliability of the system between times t_1 and t_2 as

$$R(t_1 < x < t_2) = \exp \left[-\frac{\exp(\hat{\alpha}^0 + \hat{\alpha}^1 t_2) - \exp(\hat{\alpha}^0 + \hat{\alpha}^1 t_1)}{\hat{\alpha}^1} \right].$$

A confidence band may be calculated around the expected failure time in terms of the variance of $h(t)$. This is accomplished by using the so-called *Fisher information matrix*¹ for the two estimates $\hat{\alpha}^0$ and $\hat{\alpha}^1$, expressed by

$$\begin{bmatrix} \text{Var}(\hat{\alpha}^0) & \text{Cov}(\hat{\alpha}^0, \hat{\alpha}^1) \\ \text{Cov}(\hat{\alpha}^0, \hat{\alpha}^1) & \text{Var}(\hat{\alpha}^1) \end{bmatrix} = \begin{bmatrix} -\frac{\partial^2 \ell_h}{\partial (\hat{\alpha}^0)^2} & -\frac{\partial^2 \ell_h}{\partial \hat{\alpha}^0 \partial \hat{\alpha}^1} \\ -\frac{\partial^2 \ell_h}{\partial \hat{\alpha}^0 \partial \hat{\alpha}^1} & -\frac{\partial^2 \ell_h}{\partial (\hat{\alpha}^1)^2} \end{bmatrix}^{-1},$$

where ℓ_h is the log-likelihood function. Using the Fisher information matrix, the variance of $h(t)$ can be calculated as

$$\begin{aligned} \text{Var}(\hat{h}(t)) &= \left(e^{\alpha^0 + \alpha^1 t} \right)^2 \text{Var}(\hat{\alpha}^0) + \left(t e^{\alpha^0 + \alpha^1 t} \right)^2 \text{Var}(\hat{\alpha}^1) \\ &\quad + 2 \left(e^{\alpha^0 + \alpha^1 t} \right) \left(t e^{\alpha^0 + \alpha^1 t} \right) \text{Cov}(\hat{\alpha}^0, \hat{\alpha}^1). \end{aligned} \quad (3.38)$$

¹The Fisher information matrix is used in statistics as a measure of the amount of information a random variable contains about a certain parameter of a distribution which may be used to model the random variable.

From this expression it is now possible to calculate an upper confidence limit $t_{u,n}$ and a lower confidence limit $t_{\ell,n}$ around the expected next failure estimation calculated in (3.37). This is achieved by solving for $t_{u,n}$ and $t_{\ell,n}$, respectively, in

$$\int_{t_r}^{t_{u,n}} \left[h(x) - Z_\beta \sqrt{\text{Var}(h(x))} \right] dx = 1 \quad (3.39)$$

and

$$\int_{t_r}^{t_{\ell,n}} \left[h(x) + Z_\beta \sqrt{\text{Var}(h(x))} \right] dx = 1, \quad (3.40)$$

where $\text{Var}(h(x))$ is calculated using (3.38) and Z_β is the corresponding Z value for the required significance level β , since in general, the maximum likelihood estimates of the parameters are asymptotically normal for a large enough data set [178]. It is important to note that the upper limits of the integrals are only defined for values larger than zero. In cases where the distribution does not fit the data well enough, the integral will not converge to 1, in which case no upper limit can be quantified.

3.4.3 The Non-homogeneous poisson process following a power law

The *NHPP model following a power law*, also known as the *Duane model* or the *United States Army Materials System Analysis Activity (AMSAA) model* [171], is another model that has proven successful for a variety of applications. This model follows a ROCOF of the form

$$h(t) = \lambda \delta t^{\delta-1}, \quad (3.41)$$

where δ is positive for a repairable system and t represents continuous global time [182]. As with the NHPP following an exponential law, the number of failures expected to have occurred over a certain time period may be calculated by integrating (3.41) over the time period in question, that is

$$H(t) = \int_0^t h(t) dt = \hat{\lambda} t^{\hat{\delta}}, \quad (3.42)$$

where t here denotes the end of the time period and where $\hat{\lambda}$ and $\hat{\delta}$ are estimates for the values of λ and δ . From (3.42), it is possible to calculate the timing of the n -th failure by solving for t in

$$H(t) = \hat{\lambda} t^{\hat{\delta}} = n, \quad (3.43)$$

where $n - 1$ is the total number of failures already observed in the system. Using (3.10), it is possible to calculate the reliability of the system between times t_1 and t_2 as

$$R(t_1 < x < t_2) = e^{-\lambda(t_2^{\delta} - t_1^{\delta})}.$$

A confidence band may be calculated around the expected failure time of the NHPP following a power law in terms of the variance of $h(t)$ as for the NHPP following an exponential law model. This may be accomplished by again using the *Fisher information matrix* for the two estimates $\hat{\lambda}$ and $\hat{\delta}$, expressed by

$$\begin{bmatrix} \text{Var}(\hat{\lambda}) & \text{Cov}(\hat{\lambda}, \hat{\delta}) \\ \text{Cov}(\hat{\lambda}, \hat{\delta}) & \text{Var}(\hat{\delta}) \end{bmatrix} = \begin{bmatrix} -\frac{\partial^2 \ell_h}{\partial \lambda^2} & -\frac{\partial^2 \ell_h}{\partial \lambda \partial \delta} \\ -\frac{\partial^2 \ell_h}{\partial \lambda \partial \delta} & -\frac{\partial^2 \ell_h}{\partial \delta^2} \end{bmatrix}^{-1} = \begin{bmatrix} \frac{n-1}{\hat{\lambda}^2} & t_{n-1}^{\hat{\delta}} \ln t_{n-1} \\ t_{n-1}^{\hat{\delta}} \ln t_{n-1} & \frac{n-1}{\hat{\delta}^2} + \hat{\lambda} t_{n-1}^{\hat{\delta}} \ln^2 t_{n-1} \end{bmatrix}^{-1},$$

where ℓ_h again denotes the log-likelihood function [178]. Using the Fisher information matrix for the NHPP following a power law, the variance of $h(t)$ is given by

$$\begin{aligned} \text{Var}(\hat{h}(t)) &= \left(\delta t^{\delta-1}\right)^2 \text{Var}(\hat{\lambda}) + \left(\lambda t^{\delta-1} + \lambda(\delta-1)t^{\delta-1}\right)^2 \text{Var}(\hat{\delta}) \\ &\quad + 2\left(\delta t^{\delta-1}\right)\left(\lambda t^{\delta-1} + \lambda(\delta-1)t^{\delta-1}\right) \text{Cov}(\hat{\lambda}, \hat{\delta}). \end{aligned} \quad (3.44)$$

From this expression it is again possible to calculate an upper confidence limit $t_{u,n}$ and a lower confidence limit $t_{\ell,n}$ around the expected next failure estimation calculated in (3.43). This is achieved by solving for $t_{u,n}$ and $t_{\ell,n}$ in (3.39) and (3.40) using the NHPP power law model's variance in (3.44). As with the NHPP following an exponential law, it is important to note that the integral upper limits are only defined for positive values. In cases where the distribution does not fit the data well enough, the integral will again not converge to 1, in which case no upper limit can be quantified.

3.5 Life data classification

In order to construct a lifetime distribution model which is capable of making a good prediction of the lifetime of a system, failure data or times-to-failure data are required that are accurate and complete. In practice, however, it is not always possible to obtain complete data or the data may contain some uncertainty, but such data may nevertheless still be useful for the model in some cases. Data such as those mentioned above may be classified into two categories, namely *complete data* and *censored data* [180].

3.5.1 Complete data

A complete set of data contains times to failure for all the systems in a sample. The exact failure time of each of the systems in the sample is therefore known in this case. A graphical representation of the case of complete data is shown in Figure 3.2 for a sample of six systems. Note that each system was observed until a failure occurred.

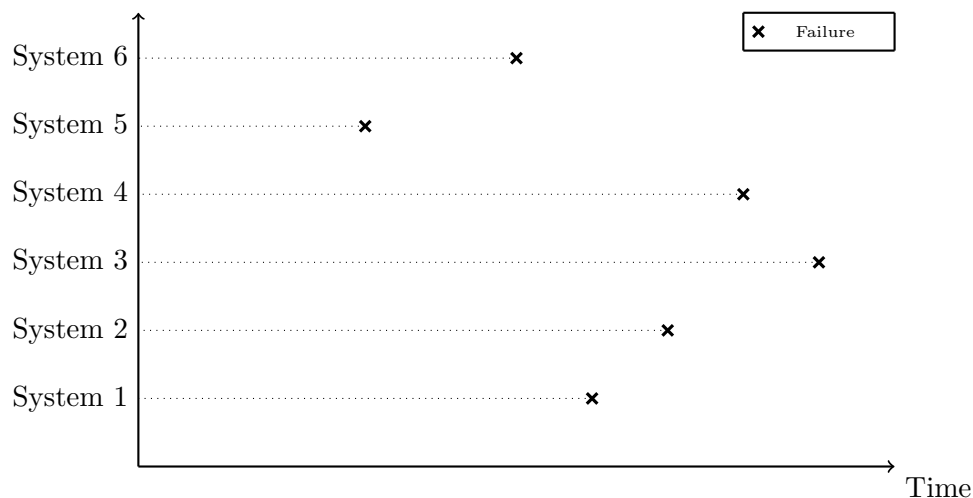


FIGURE 3.2: A graphical representation of the case of complete failure data.

3.5.2 Right-censored data

Suspended data, or *right-censored data*, is the most common case of censoring. In this case, a failure is not observed for every system in the sample. A graphical representation of right-censored data is shown in Figure 3.3 for a sample of six systems. In the figure, three of the six systems did not fail and therefore the failure data of these three systems are referred to as right-censored data [180]. The term *right-censored* is used, because failures are expected to occur to the right of the current lifetimes of these three systems.

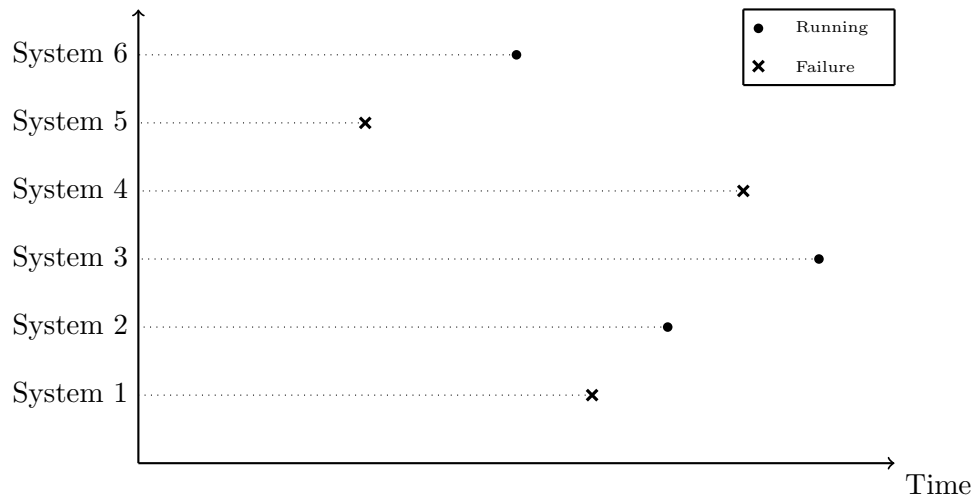


FIGURE 3.3: A graphical representation of the case of right-censored failure data.

3.5.3 Interval-censored data

Another type of censored data is *interval-censored data*. This type of censored data refers to a certain interval during which a system is known to have failed. The exact failure time is uncertain, but it is known in this case that the system failed within a certain time interval during the system's lifetime. This type of data typically arises when the state of a system is not continuously monitored but rather monitored at fixed points in time during the lifetime of the system. In this case, a system will be functioning when performing an inspection of the system, but when performing the next inspection, it may be noticed that the system has already failed [146]. A graphical representation of interval-censored data is shown in Figure 3.4 for a sample of six systems. When it is not possible to continuously monitor a system in order to observe its failures, the interval inspection approach has to be adopted. Interval inspection, however, does not capture as much information as complete data or right-censored data, and is therefore avoided if possible.

3.5.4 Left-censored data

Left-censored data are similar to interval-censored data in the sense that the exact failure time of the system is not known. A failure in the case of left-censored data is, however, only known to have occurred before a certain time during the system's lifetime. This type of data is typically gathered when the inspection interval is too large, which causes the system to fail before it is inspected for the first time [206]. A graphical representation of the case of left-censored data is shown in Figure 3.5 for a sample of six systems.

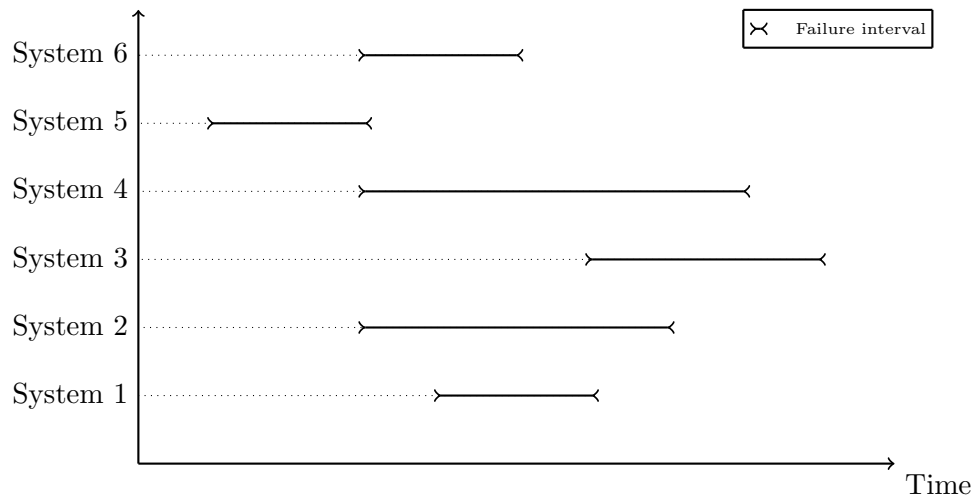


FIGURE 3.4: A graphical representation of the case of interval-censored failure data.

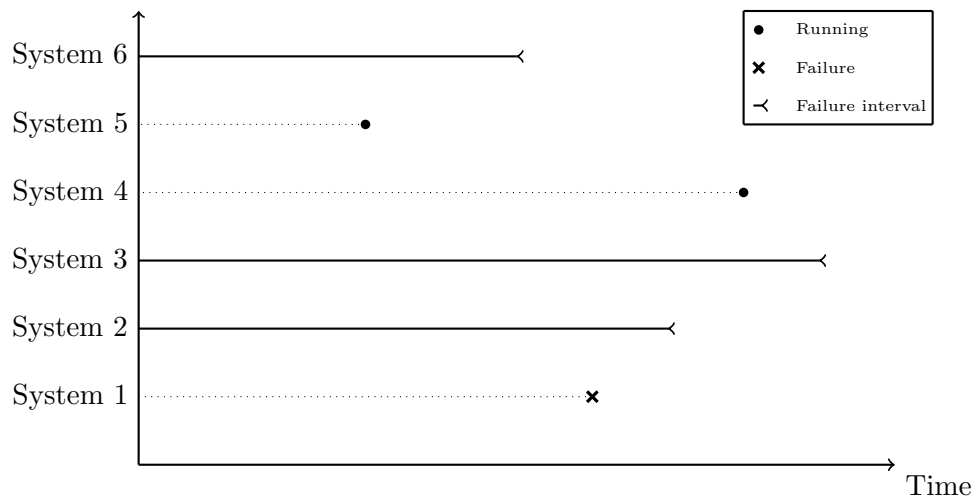


FIGURE 3.5: A graphical representation of the case of left-censored failure data.

3.6 Trend tests

Three major failure data trend tests are reviewed in this section. These tests may be used to determine whether or not a trend exists in the failure times of a data set and include the *reverse arrangement test*, the *military handbook test* and the *Laplace trend test*. There are several alternative trend tests, such as the *Lewis-Robinson test* [10] and the *generalised Anderson-Darling test* [143], but these alternatives are not as popular as the three trend test reviewed here [171].

3.6.1 The reverse arrangement test

The *reverse arrangement test* (RAT) is a simple method for testing for the presence of a trend in a data set without making any assumptions about what kind of model the trend will follow [167, 171]. The RAT is related to Kendall's so-called *tau test* which was first introduced in 1938 [124] and may be used to distinguish between independent and identically distributed interarrival

times on the one hand, and a monotonic trend on the other [204].

Consider a data set with $n - 1$ interarrival times, T_1, \dots, T_{n-1} . A *reversal* is defined as a later observed instance that is strictly greater than an earlier observed instance [167, 171]. A reversal therefore occurs each time $T_i < T_j$ for $i = 1, 2, \dots, n - 2$, and $j = i + 1, \dots, n - 1$. Kendall demonstrated that the distribution of the total number of reversals of a system approaches a normal distribution rapidly as n increases [124]. More specifically, the random variable

$$Z = \frac{R - \frac{(n-1)(n-2)}{4} + 0.5}{\sqrt{\frac{(n-1)(n-2)(2(n-1)+5)}{72}}}$$

is approximately normally distributed with a mean of 0 and a standard deviation of 1 for large values of n , where R is the total number of reversals and $n - 1$ denotes the total number of observed failures. This normal distribution approximation is accurate for values of n greater than 12 [204]. In fact, at a significance level β , the upper critical value for the distribution is

$$R_{n-1,u}^\beta = z_{\text{crit}} \sqrt{\frac{(n-1)(n-2)(2(n-1)+5)}{72}} + \frac{(n-1)(n-2)}{4} - 0.5$$

for n -values larger than 12, where z_{crit} is the corresponding z -value of the required significance level β . The lower corresponding critical value $R_{n-1,\ell}^\beta$ is

$$R_{n-1,\ell}^\beta = R_{n-1,\text{max}} - R_{n-1,u}^\beta,$$

where $R_{n-1,\text{max}} = (n - 1)(n - 2)/2$. For n -values at most 12, the corresponding upper and lower critical values are given in Table 3.1.

TABLE 3.1: Critical values of the random variable R for n -values less than or equal to 12. These are the upper and lower significant levels for the reverse arrangement test.

Sample size (n)	Single-sided lower significance level			Single-sided upper significance level		
	1%	5%	10%	10%	5%	1%
4		0	0	6	6	
5	0	1	1	9	9	10
6	1	2	3	12	13	14
7	2	4	5	16	17	19
8	4	6	8	20	22	24
9	6	9	11	25	27	30
10	9	12	14	31	33	36
11	12	16	18	37	39	43
12	16	20	23	43	46	50

3.6.2 The military handbook test

The *military handbook test* (MHT) was first introduced in 1981 by the United States Department of Defence [209] and is generally the best method to employ when deciding between a *Non-homogeneous poisson process* (NHPP) power law model, also known as the *Duane model*, and

no trend in failure data emanating from a system [171, 173]. The MHT method tests the hypothesis that no trend exists in the data set, based on the chi-square distribution. The relevant chi-square statistic is

$$\chi_{2(n-1)}^2 = 2 \sum_{i=1}^{n-1} \ln \frac{Q_{\text{end}}}{T_i},$$

where Q_{end} denotes the discrete observation time (measured in global units of time) at the end of the observation period which may be an observed failure time or merely the system lifetime to date, as illustrated in Figure 3.6. Furthermore, T_i denotes the discrete failure time (also measured in global units of time) of the i -th failure as before, and $n - 1$ again denotes the total number of observed failure events [167, 171].

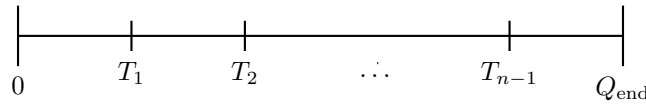


FIGURE 3.6: A graphical representation of the parameter Q_{end} .

This statistic is compared to the percentiles of the chi-square distribution with $2(n - 1)$ degrees of freedom. In the case where $\chi_{2(n-1)}^2 > \chi_{\text{crit},\beta}^2$, at a significance level of β , evidence exists against the null hypothesis of no trend present in the data, in which case the data set may be classified as emanating from a repairable system and exhibiting an improving trend. In the case where $\chi_{2(n-1)}^2 < \chi_{\text{crit},1-\beta}^2$, however, evidence also exists that a trend is present in the data, in which case the data set is classified as emanating from a repairable system and exhibiting a decreasing trend [209]. Finally, in the case where $\chi_{\text{crit},\beta}^2 < \chi_{2(n-1)}^2 < \chi_{\text{crit},1-\beta}^2$, there is not enough evidence to reject the null hypothesis of no trend and the data is classified as emanating from a non-repairable system [171]. A graphical representation of the outcomes of the MHT is shown in Figure 3.7 at a significance level of β .

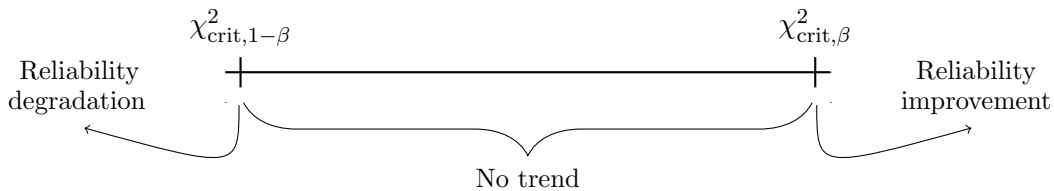


FIGURE 3.7: Possible outcomes of the military handbook test at significance level β .

3.6.3 The Laplace test

The *Laplace trend test*, also known as the *centroid test*, is most often used for the purpose of identifying a trend in a data set of failure times [47] and is generally the most accurate trend test when deciding between an NHPP following an exponential law and no trend. It was developed by De Laplace in 1773 to test whether or not comets originate in our solar system [47]. The Laplace trend test is carried out to test the hypothesis that a trend does not exist among a set of interarrival failure data and can determine whether the system from which the data emanate exhibits improvement, deterioration or whether there is no trend in the data set. The test statistic for the hypothesis test is

$$U = \frac{\sum_{i=1}^{n-2} (T_i - \frac{T_{n-1}}{2}) \times \sqrt{12(n-2)}}{(n-2) \times T_{n-1}},$$

where T_i denotes the discrete failure time (measured in global units of time) of the i -th failure and $n - 1$ denotes the total number of observed failure events. In the case where the observation period is not ended at the instant when a failure is observed, the Laplace trend test statistic is

$$U = \frac{\sum_{i=1}^{n-1} (T_i - \frac{Q_{\text{end}}}{2}) \times \sqrt{12(n-1)}}{(n-1) \times Q_{\text{end}}},$$

where Q_{end} again denotes the time (again measured in global units of time) at the end of the observation period, as shown in Figure 3.6. This test statistic approximates a standard normal distribution, and so the critical value Z for the hypothesis test is obtained from the standard normal table at a given significance level β . If $U \geq Z_{1-\beta/2}$, then evidence exists of strong reliability degradation of the system from which the failure data emanate, whereas if $U \leq Z_{\beta/2}$, evidence exists of strong reliability improvement. In both these cases there is enough evidence to reject the hypothesis that there does not exist a trend in the data set, in which case the set is classified as data emanating from a repairable system. Finally, in the case where $Z_{\beta/2} < U < Z_{1-\beta/2}$, there is not enough evidence to reject the hypothesis that there exists no trend in the data, in which case the data set is referred to as *non-committal* and classified as emanating from a non-repairable system [179]. A graphical representation of the outcomes of the Laplace trend test is shown in Figure 3.8 at a significance level of β .

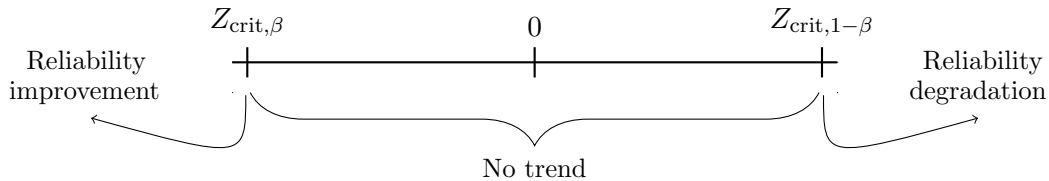


FIGURE 3.8: Possible outcomes of the Laplace trend test at significance level β .

3.7 Model parameter estimation methods

Lifetime distributions involve certain parameters which have to be estimated in order to fit the models to sample data. These parameters may be estimated by evaluating the sample data. In this section, three such parameter estimation methods are reviewed, namely the *maximum likelihood method*, the *least square method* and the *Bayesian parameter estimation method*.

3.7.1 The maximum likelihood method

One of the most robust parameter estimation methods is the *maximum likelihood estimation* (MLE) method [171]. This method determines the most likely parameter values for a selected distribution. The MLE method essentially takes a sample data set as fixed values and then selects model parameters which maximise a likelihood function for the given data set. The MLE method may be formulated mathematically by taking T_n as a continuous random variable with $n - 1$ independent observations T_1, \dots, T_{n-1} which has a pdf of

$$f(T_1, \dots, T_{n-1}; \theta_1, \dots, \theta_k), \quad (3.45)$$

where $\theta_1, \dots, \theta_k$ are the unknown parameters of the lifetime distribution model that have to be estimated [165]. The likelihood function for (3.45) is given by

$$L(\theta_1, \dots, \theta_k | T_1, \dots, T_{n-1}) = \prod_{n=1}^{n-1} f(T_1, \dots, T_{n-1}; \theta_1, \dots, \theta_k). \quad (3.46)$$

In many cases, the so-called *logarithmic likelihood function* (*log-likelihood function*) is more convenient to use, which is simply the logarithm of (3.46), *i.e.*

$$\Lambda = \ln L(\theta_1, \dots, \theta_k | T_1, \dots, T_{n-1}) = \sum_{n=1}^{n-1} \ln f(T_1, \dots, T_{n-1}; \theta_1, \dots, \theta_k). \quad (3.47)$$

To obtain the best estimates for the parameter values $\theta_1, \dots, \theta_k$, either (3.46) or (3.47) may be maximised in view of the monotonic growth property of the logarithmic function. The estimated parameter values are the simultaneous solutions to the k equations

$$\frac{\delta L}{\delta \theta_j} = \frac{\delta \Lambda}{\delta \theta_j} = 0, \quad j = 1, \dots, k. \quad (3.48)$$

The MLE method is a popular estimation method because of various advantages. As the sample size increases, for example, the estimation of the parameter values converge to the true values of the parameters. Another advantage is that the distribution of the parameter estimates is normally distributed, which enables the use of the Fisher information matrix. Another advantage is that the method can also accommodate right-censored and interval-censored data. To obtain a desirably accurate outcome from the MLE method, however, a large sample is often required, because if the sample size is too small, the method is known to perform poorly [180].

3.7.2 The least squares method

The *least squares* (LS) estimation method aims to minimise the sum of squared residuals between the actually observed values and the values provided by the fitted model [165]. Let T_n be a continuous random variable with $n - 1$ independent observations T_1, \dots, T_{n-1} . Then the residuals may be calculated as

$$r_i = y_i - f(T_1, \dots, T_{n-1} | \theta_1, \dots, \theta_k), \quad (3.49)$$

where y_i is the actually observed value and $f(T_1, \dots, T_{n-1} | \theta_1, \dots, \theta_k)$ is the value of the fitted model, given the model parameter values $\theta_1, \dots, \theta_k$. The sum of the squared residuals

$$S = \sum_{i=1}^{n-1} r_i^2 \quad (3.50)$$

is calculated and minimised in this case. This may be achieved by taking the partial derivative of S with respect to $\theta_1, \dots, \theta_k$ and setting each partial derivative equal to zero, as was done in (3.48). The resulting set of equations are then solved to obtain the unknown values. This process can easily be carried out for linear cases, but in cases where the model is nonlinear, an iterative numerical approximation algorithm is required to obtain the optimal values for the parameters of the model. When employing the LS method, it is therefore useful to linearise the function (if this is possible) in order to ease the process of obtaining good parameter values. If this is achievable, the LS method is straightforward to apply and a solution can be found relatively easily [180]. The LS method is, however, reported to perform poorly in some cases in respect of censored data.

3.7.3 The Bayesian parameter estimation method

Bayes' theorem [212] describes the conditional probability of an event based on certain conditions which may be related to the event. It relates the probability that an event will occur, to

the probability of associated events occurring or not occurring. The theorem combines prior information with sample data [180]. This characteristic of Bayes' theorem may be exploited to make inferences about a model fitted to sample data where the prior knowledge is the assumed model parameter values. Let $\varphi(\theta_1, \dots, \theta_k)$ be the prior distribution, where $\theta_1, \dots, \theta_k$ are the assumed model parameter values. The posterior distribution, given the sample data T_1, \dots, T_{n-1} , is expressed by

$$f(\theta_1, \dots, \theta_k | T_1, \dots, T_{n-1}) = \frac{L(T_1, \dots, T_{n-1} | \theta_1, \dots, \theta_k) \varphi(\theta_1, \dots, \theta_k)}{\int_{\zeta} L(T_1, \dots, T_{n-1} | \theta_1, \dots, \theta_k) \varphi(\theta_1, \dots, \theta_k) d\theta} \quad (3.51)$$

where $L(T_1, \dots, T_{n-1} | \theta_1, \dots, \theta_k)$ is the likelihood function described in (3.46) and ζ is the range of the parameter values $\theta_1, \dots, \theta_k$. The denominator in (3.51) may be interpreted as the probability of obtaining the sample data given the selected model parameter values. The integral generally does not admit a closed-form evaluation and numerical methods are therefore employed in many cases to obtain a solution.

To estimate the model parameters, three approaches may be adopted, namely obtaining the expected values of $\theta_1, \dots, \theta_k$, obtaining the median values of $\theta_1, \dots, \theta_k$, or obtaining any other percentile of the parameter values $\theta_1, \dots, \theta_k$. The most popular approach involves estimating the expected values (mean values) of the parameter values. This is achieved by calculating the mean

$$E(\theta_i) = \mu_{\theta_i} = \int_{\zeta_i} \theta_i f(\theta_1, \dots, \theta_k | T_1, \dots, T_{n-1}) d\theta_i \quad (3.52)$$

of each parameter $\theta_i \in \{\theta_1, \dots, \theta_k\}$ individually, where ζ_i denotes the i -th orthotope of ζ . A similar approach is taken to estimate the medians of the parameters. The i -th median is obtained by solving for $\theta_i^{0.5}$ in

$$\int_0^{\theta_i^{0.5}} f(\theta_1, \dots, \theta_k | T_1, \dots, T_{n-1}) d\theta_i = \frac{1}{2}. \quad (3.53)$$

This approach may be adapted to estimate any other percentile of the parameter values, by solving for θ_i^z in

$$\int_0^{\theta_i^z} f(\theta_1, \dots, \theta_k | T_1, \dots, T_{n-1}) d\theta_i = z, \quad (3.54)$$

where z is the desired percentile. An advantage of employing the Bayesian parameter estimation method is that it provides a theoretical framework for combining prior information with sample data. The inferences are exact and are conditional on the data — therefore no asymptotic assumptions have to be made. This approach does not, however, specify which prior distribution should be selected and so an external justification for assuming a particular distribution for the sample data is required [26].

3.8 Acceleration models

When a system operates at stress, failures are expected to occur earlier during the lifetime of the system than when the system functions under normal conditions. This phenomenon is referred to as *acceleration*. The stress under which the system operates may result from high temperature, voltage, humidity or pressure [73]. Such accelerated failures in the system may occur due to mechanical fatigue, diffusion or corrosion, which also cause failures in a system under normal conditions, but according to a decelerated time scale.

It is commonly assumed that the acceleration of the failure rate time scale as a result of system stress is linear, which means that the time to failure under high stress is simply multiplied by

a certain factor, called the *acceleration factor* (AF) [171]. The notion of acceleration may be used to analyse the failure rate of a system under a certain stress level, and may then be used to predict the behaviour of the system at a different stress level. The AF may be expressed as

$$\text{AF} = \frac{G(V_{use})}{G(V_{acc})}, \quad V_{acc} \geq V_{use}, \quad (3.55)$$

where $G(V_{use})$ is a life measure of the system, such as the mean lifetime, or a stress model function at the use stress level V_{use} , while $G(V_{acc})$ is the same life measure of the system at the accelerated stress level V_{acc} .

An acceleration model in reliability theory is a model that may be used to predict the time to failure of a system as a function of stress. Such models are typically based on physical or chemical induced fatigue. In this section, two acceleration models are reviewed, namely the *Arrhenius model* and the *Eyring model*.

3.8.1 The Arrhenius model

The most common life-stress acceleration model is the so-called *Arrhenius model* [171]. This model predicts how temperature in a system influences the time-to-failure of the system. The model is derived from the so-called Arrhenius reaction rate equation dating back to 1887 [177],

$$R(T) = Ae^{-\frac{E_A}{KT}}, \quad (3.56)$$

where R denotes the reaction speed, T is the absolute temperature of a reaction in Kelvin, A is an unknown value called the *nonthermal* constant, E_A denotes the activation energy required for the chemical process, and K denotes Boltzman's constant (8.617385×10^{-5} eV k^{-1}).

The Arrhenius life-stress model is derived from (3.56) under the assumption that the inverse reaction rate is proportional to the lifetime of the process. The stress relationship is formulated as

$$G(V) = Ce^{\frac{B}{V}}, \quad (3.57)$$

where G denotes an appropriate life measure of the system, V denotes the stress level at which the system is operating (measured in absolute temperature), and B and C are model parameters.

The AF for the Arrhenius model may be derived, using (3.55), as

$$\text{AF} = \frac{G_{use}}{G_{acc}} = \frac{Ce^{\frac{B}{V_{use}}}}{Ce^{\frac{B}{V_{acc}}}} = e^{\left(\frac{B}{V_{use}} - \frac{B}{V_{acc}}\right)}. \quad (3.58)$$

The Arrhenius model has successfully been employed in the contexts of electronic equipment failures due to chemical reactions, migration processes and diffusion processes [171].

3.8.2 The Eyring model

The *Eyring model* arose from theoretical studies in chemistry and quantum mechanics. As with the Arrhenius model, the Eyring model is most often applied when the stress causing system acceleration is temperature-dependent. This acceleration model, however, can take into account stresses other than temperature, such as voltage or humidity [171]. The Eyring model is very similar to the Arrhenius model, but differs from it in that more than one type of stress can be included in the model. The stress relationship of the model is given by

$$G(V) = \frac{1}{V}e^{-(A-\frac{B}{V})}, \quad (3.59)$$

where G again denotes a suitable life measure of the system, V denotes the stress level at which the system is operating, and A and B are model parameters [177].

The AF for the Eyring model is

$$\text{AF} = \frac{G_{use}}{G_{acc}} = \frac{\frac{1}{V_{use}} e^{-\left(A - \frac{B}{V_{use}}\right)}}{\frac{1}{V_{acc}} e^{-\left(A - \frac{B}{V_{acc}}\right)}} = \frac{V_{acc}}{V_{use}} e^{\left(\frac{B}{V_{use}} - \frac{B}{V_{acc}}\right)}. \quad (3.60)$$

3.9 Chapter summary

A literature review of certain basic notations and methods from the realm of reliability theory was provided in this chapter. In §3.1, various general considerations in reliability theory were covered in some detail. The mathematical formulations of reliability expressions were provided in §3.2, and this was followed by a review of the two main reliability system types (described in §3.3 and §3.4), namely *non-repairable systems* and *repairable systems*, respectively. The standard life classification of failure data was described in §3.5, including complete data, right-censored data, interval-data and left-censored data. This was followed in §3.6 by a review of a number of popular methods of detecting trends in a data set. In §3.7, three methods were described for estimating model parameters, namely the maximum likelihood method, the method of least squares and the Bayesian estimation method. Finally, two acceleration models were described in §3.8, namely the *Arrhenius model* and the *Eyring model*.

Part II

Mathematical modelling framework

CHAPTER 4

Mathematical model formulations

Contents

4.1	The GMS objective functions selected	65
4.2	Model assumptions	67
4.3	GMS models	69
4.4	Chapter summary	77

In this chapter, a motivation is provided for the objective functions employed in this dissertation as scheduling criteria in newly proposed models for the GMS problem. This is followed by a discussion on and motivation of a number of assumptions made in order to facilitate derivation of the mathematical model formulations for the GMS problem adopted in this dissertation. These model formulations are presented in some detail, and this presentation includes an overview of the model variables, the two objective functions employed, and the model constraints.

4.1 The GMS objective functions selected

In §2.1.3, an overview was provided of the three main types of GMS criteria adopted in the literature. In many cases, and especially in privately owned power utilities, economic scheduling criteria are considered the most important type of criteria as minimising the overall cost of producing energy will increase the profit of the power utility. National power utilities, which are typically owned by the governments of nations, are, however, typically not as cost-orientated. A national power utility is usually responsible for supplying an entire nation with power and in such a case the reliability of the energy supply is more important than the cost of generating the power. As the type of decision support pursued in this dissertation is aimed at national power utilities, the focus is on the class of reliability GMS criteria.

4.1.1 Minimising probability of unit failure

Reliability scheduling criteria have widely been adopted in GMS model formulations. In such models, examples of scheduling objectives include minimising the sum of the squared energy reserves, minimising the difference between the average reserve load and the actual reserve load, and minimising the loss of load probability. The author could, however, not find any reference in the literature to an objective function that explicitly takes into account the probability of PGU failure. A GMS objective function which minimises the probability that any PGU in the

power generating system will fail during a fixed scheduling window is expected to minimise the chance that unexpected outages occur during power generation. Such a function may be very useful in a power generating system that is old and has to supply a high energy demand, which typically increases the probability of PGU failures.

An objective function is therefore adopted in one of the two GMS models put forward in this dissertation which minimises the probability that any PGU in the system will fail during the scheduling window. These failure probabilities are weighted according to the rated capacities of the various PGUs in the system so as to ensure that PGUs contributing a larger portion to the overall capacity of the power generating system do not operate at high probabilities of failure as this may cause the system not to be able to supply the required demand reliably.

The main difference between the LOLP scheduling criterion, reviewed in §2.2.1, and minimising the newly proposed unit failure GMS criterion is that the LOLP does not take into account the severity of a failure in the system. For the same value of LOLP, the degree of the loss may be less than 1 MW or greater than 500 MW, for example. The LOLP therefore does not recognise the degree of either spare capacity or energy shortages, which is accounted for in the newly proposed scheduling criterion by weighting the objective function by means of the rated capacity. This means that the LOLP will give the same scheduling preference for a small PGU whose failure is imminent as for a large PGU whose failure is equally imminent, which may have a more detrimental effect on the service level of the overall power system. The newly proposed scheduling criterion, however, places more emphasis on minimising the probability that any PGU in the system will fail, whereas the LOLP places focus on the probability that the load will exceed the available capacity.

4.1.2 Maximising expected energy production

An alternative PGU maintenance scheduling objective to the one discussed in §4.1.1 involves the maximisation of the expected system-wide energy production within a pre-determined scheduling window, taking into account the possibility of unexpected failures of the PGUs in the system. As was the case for the scheduling objective of minimising a weighted measure of the probability of PGU failure, the author could not find any reference in the literature in which the expected energy production is maximised explicitly whilst taking into account unexpected failures of PGUs in the system. Such a schedule will also minimise the negative effects on power generation capabilities of the occurrence of unexpected failures of PGUs in the generating system. A scheduling objective which maximises the system-wide expected energy production may be very useful for power utilities as this is a measure of the total amount of available energy anticipated over a given scheduling window. This measure may aid in decision making when forecasting energy production levels for the given scheduling window.

A scheduling criterion is therefore also put forward in this dissertation which seeks to maximise the expected energy production during the PGU maintenance scheduling window. This function takes into account three possible cases of failure occurrences of PGUs in the system under the assumption that each PGU will fail at most once during the maintenance scheduling window¹. The first case is where a PGU failure is observed before planned maintenance is scheduled to be performed on that PGU. The second case is where a PGU failure is observed after planned maintenance has already been performed on that PGU, but still within the scheduling window. The third case is where a failure is only observed after the current scheduling window has ended. The timing of PGU failures is modelled by the incorporation of random variables into the GMS

¹This assumption is justified as long as the GMS window is not too long.

model facilitating calculation of the expected failure time for each PGU, using methods from probability theory, reliability theory and the estimated failure rates for the PGUs.

Comparing this proposed function to the EENS scheduling criterion, which may at first seem very similar to the maximising expected energy production criterion, reveals that the proposed function for calculating the expected energy is more realistic in the way that it takes into account the duration that a PGU is offline during maintenance and whether or not a failure occurs before or after maintenance has been performed. This provides the decision maker with a better estimate of the expected energy produced during the scheduling window. The proposed criterion therefore takes into account the reliability of the PGUs and schedules these PGUs for maintenance by taking into account the expected failures and durations of maintenance in such a way as to maximise energy production.

The EENS criterion (reviewed in §2.2.1), on the other hand, minimises the energy not served during occasions when the load exceeds the available capacity. The newly proposed GMS criterion therefore aims to schedule maintenance such that the PGUs are optimised for maximum energy generation on expectation, whereas the EENS criterion schedules PGUs for maintenance during time periods of low demand in order to ensure that the demand does not exceed the available capacity. In other words, the latter function considers the demand during each time period of the scheduling window and schedules accordingly, whereas the newly proposed function only considers the reliability of the PGUs in the system and aims to maximise the energy that each PGU is expected to generate during the scheduling window based on the constraints of the system, which includes the demand constraint.

Two scheduling solutions (one obtained via the EENS criterion and the other according to the newly proposed function) may, for example, exhibit the exact same available capacity over the scheduling window and both may adhere to the demand constraint, but for the newly proposed function, the reliability of the PGUs (as well as the overall energy produced by the system) will be higher as this is what the function explicitly maximises.

4.2 Model assumptions

In this section, a number of assumptions are presented and motivated in order to arrive at manageable model formulations for the GMS problem in which the objective is either to minimise the probability of PGU failure or to maximise the system-wide expected energy production. Some of these assumptions are aimed at decreasing the complexity of the problem, thereby making it possible to solve the model efficiently. The model complexity is, however, decreased in such a manner so as not to generate maintenance schedules that are unrealistic or unfit for use in practice.

1. *Components of the PGUs.* A number of components are required to generate electric power, including boilers, steam turbines and generators. Failure of any one of these components typically causes the power generation process to be interrupted until the component has been repaired or replaced. The combination of all of these components are together referred to as a PGU. Therefore, a failure in one of the components of the PGU typically leads to failure of the entire PGU. For modelling purposes, all the components of a PGU are considered as a whole in the sense that PGU failures are not attributed to any specific components.
2. *Frequency of maintenance.* A number of different types of planned maintenance procedures may be performed on PGUs, including complete overhauls of PGUs as opposed to mere

routine check-ups. Moreover, PGUs may require planned maintenance more than once over a GMS planning period, especially during long scheduling windows. For the purpose of the models considered in this dissertation, however, it is assumed that only one type of maintenance procedure is performed on the PGUs in a system. It is therefore implicitly assumed that the length of the scheduling window is short enough to justify the assumption that exactly one planned maintenance is required for each PGU during the scheduling window. The duration of planned maintenance may, however, vary from one PGU to another as dictated by the power system scenario.

3. *Frequency of failure.* A number of different fuel types (*i.e.* coal, gas or water) are required for power generation in different types of PGUs. The fuel type may influence the failure rates of PGUs. The failure rates may also differ over instances of the same type of PGU. These difference in failure rates of the PGUs may lead to some PGUs failing more often than others, especially during long scheduling windows. For the purpose of the models considered in this dissertation, it is, however, assumed that no PGU will fail more than once during a scheduling window. This assumption is again justified if the scheduling window is not too long.
4. *Contiguity of maintenance procedures.* In any specific GMS problem instance, the duration of maintenance performed in respect of any given PGU is assumed to be constant. The duration of the maintenance furthermore has to be performed without interruption. That is, when a PGU is scheduled for maintenance, the entire period of maintenance has to be completed during consecutive time periods.
5. *Initial conditions.* Planned maintenance of PGUs is only scheduled over one scheduling window at a time. It is furthermore assumed that no PGU failure has been observed since completion of the previous maintenance of a PGU up to the end of the previous scheduling window. It is therefore implicitly assumed that at the beginning of the scheduling window, all of the PGUs in the power system are in a working condition.
6. *Reliability after maintenance.* When maintenance is performed on a PGU, the aim is to increase the reliability of the PGU. This reliability can either be assumed to be increased to “as good as new” or to the level of reliability at which it was operating before performing the last maintenance procedure, as explained in §3.4. In this dissertation, it is assumed that after having performed maintenance on a PGU and placing it back into operation, the PGU starts to operate at 100% reliability.
7. *Transmission line maintenance and constraints.* The problem of transmission line maintenance was described in some detail in §2.1.5. The maintenance of transmission lines in a power system typically depends on GMS as maintenance on transmission lines relaying power from a PGU is only possible during periods when maintenance is performed on the PGU in question. The reason for this is that maintenance cannot be performed on transmission lines while these lines are actively used for the transmission of electricity. Therefore, when the PGU providing electricity via a certain transmission line is scheduled for planned maintenance, that transmission line is typically also scheduled for maintenance. In the GMS model formulations in this dissertation, however, the focus is on GMS — the scheduling of planned maintenance on transmission lines in the power system is not taken into account. The transmission constraints concerned with transmission capabilities and the transmission network, as described in §2.2.2, are therefore excluded from the mathematical model formulation in this dissertation.
8. *Resources required for maintenance.* In a realistic power system, many resources are required to perform planned maintenance on PGUs successfully. These resources include

maintenance personnel and spare parts. A model accommodating constraints on all the required resources for PGU planned maintenance is expected to be very complex. For the purposes of the mathematical model formulations in this dissertation, it is therefore assumed that the only resource required for PGU planned maintenance is the maintenance crew responsible for performing the planned maintenance. This is not an unrealistic assumption, since the type of maintenance being scheduled is planned maintenance, which means that it is known beforehand that maintenance of any particular PGU will occur during a certain period within the scheduling window. Provision can therefore be made well in advance of each maintenance event to ensure that the spare parts and maintenance equipment required to perform the maintenance successfully, are indeed available.

9. *Varying maintenance crew requirements.* The complexity of the model is slightly increased by assuming that the maintenance crew required during each time period of planned maintenance is not necessarily the same. It is assumed that during each time period of planned PGU maintenance, a possibly different number of maintenance crew members may be required in order to successfully complete the planned maintenance. In other words, the number of maintenance crew members required to perform planned maintenance successfully on any particular PGU may vary over the duration of such maintenance.
10. *Nature of the generating system.* Within the realm of reliability theory, two main types of systems prevail, namely non-repairable systems and repairable systems, as described in §3.3 and §3.4, respectively. The type of system is typically determined by the trend existing in the failure data. Wang and McDonald [214], however, claim that the components of a PGU “are all repairable.” According to Assumption 1, the PGU is furthermore seen as a whole (that is, separate analyses of individual PGU component failures are not carried out). For this reason, it is assumed that each PGU as a whole is also a repairable system.
11. *The failure rates of the PGUs.* Wang and McDonald [214] also state that although it is thought that the failure rate functions of the components of a PGU follow a bath-tub curve, as explained in §3.1, these components spend most of their time in the “useful life” phase of this curve, thus exhibiting an approximately constant failure rate which follows an exponential distribution. In this dissertation, PGU reliability incorporated into the GMS objective functions is therefore formulated as an exponential function for a repairable system which is in its “useful life” stage. Under this assumption, it is also safe to assume that only one planned maintenance is to be performed for each PGU during the scheduling window, if this scheduling window is not too long, as explained in Assumption 2.
12. *Independence of PGU failures.* Failures that occur in a system of PGUs are assumed to be independent of one another. A PGU that is taken out of operation due to a failure is therefore assumed to have little or no effect on the timing of failures of the other PGUs in the power generating system.

4.3 GMS models

In this section, two mathematical models are derived for the GMS problem which are used throughout the remainder of the dissertation. The aim is either to minimise the overall probability of unit failure, weighted by the rated capacities of PGUs, or to maximise the expected system-wide energy production, subject to a set of constraints over a fixed scheduling window, as outlined in §4.2.

4.3.1 The model variables

Suppose the power system contains n PGUs indexed by the set $\mathcal{U} = \{1, \dots, n\}$ and that the scheduling window for the system is discretised into m time periods of equal length, indexed by the set $\mathcal{P} = \{1, \dots, m\}$. Furthermore, let $x_{u,p}$ be a binary decision variable taking the value 1 if planned maintenance of PGU $u \in \mathcal{U}$ is scheduled to start during time period $p \in \mathcal{P}$, or zero otherwise. Let $y_{u,p}$ be a binary auxiliary variable taking the value 1 if planned maintenance of PGU $u \in \mathcal{U}$ is scheduled to take place during time period $p \in \mathcal{P}$, or zero otherwise.

A schematic representation of the variables defined above is provided in Figure 4.1, where x'_u is a predetermined (negative) parameter denoting the time period during which PGU $u \in \mathcal{U}$ entered into operation again after the previous maintenance operation which was performed on the PGU (during the previous scheduling window) and where d_u is the duration of planned maintenance for PGU $u \in \mathcal{U}$.

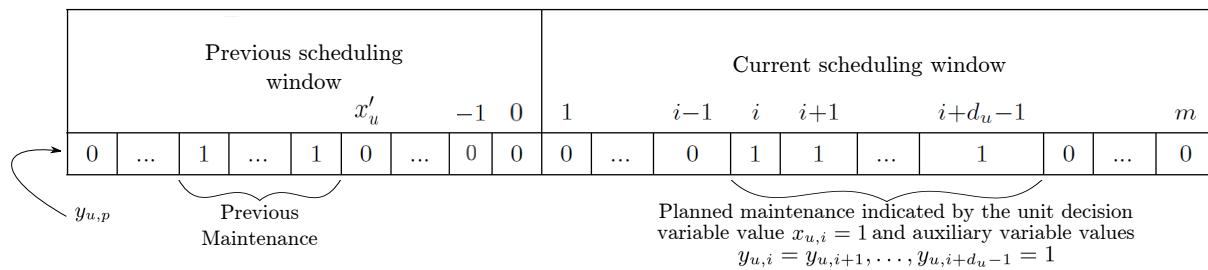


FIGURE 4.1: Representation of the variables and parameters related to PGU $u \in \mathcal{U}$ in the mathematical model formulations.

4.3.2 The objective functions

In this section, two new GMS objectives are introduced. The first scheduling criterion aims to minimise the probability of PGU failure, whereas the second scheduling criterion aims to maximise the expected system-wide energy production over a pre-determined scheduling window.

Minimising probability of unit failure

An alternative to the GMS objectives described in §2.2.1 may be to maximise the probability that no PGU will fail during the time interval between when a unit was placed back into service after its previous maintenance and when its next maintenance is scheduled. When a PGU is placed back into operation after its previous maintenance, it is assumed that the unit is as good as new and is 100% reliable at that point in time, as per Assumption 6 of §4.2. Let z_u denote the time that has elapsed since the previous maintenance of PGU $u \in \mathcal{U}$ ended (during the previous scheduling window). The probability that a failure of PGU $u \in \mathcal{U}$ will have been observed by time z_u is

$$F(z_u) = \int_0^{z_u} f(x) dx = \int_0^{z_u} \lambda_u e^{-\lambda_u x} dx = 1 - e^{-\lambda_u z_u}, \quad (4.1)$$

according to Assumption 11, where λ_u is the failure rate of PGU $u \in \mathcal{U}$. In the expression (4.1), the variable z_u is continuous, but this may be viewed in the discretised context of Figure 4.1 in order to estimate the probability $F(z_u)$ in terms of the number of scheduling time periods that have elapsed since the previous maintenance of PGU u ended (*i.e.* period x'_u) to when maintenance is scheduled to commence during the current scheduling window (*i.e.* $\sum_{p \in \mathcal{P}} p x_{u,p}$).

Note that $z_u \approx \sum_{p \in \mathcal{P}} px_{u,p} - x'_u$ for PGU $u \in \mathcal{U}$ by Assumption 5. The probability $F(z_u)$ in (4.1) that a failure will be observed for PGU $u \in \mathcal{U}$ by time z_u may therefore be approximated by

$$F(x_{u,p}) = 1 - e^{-\lambda_u \left(\sum_{p \in \mathcal{P}} px_{u,p} - x'_u \right)}, \quad (4.2)$$

where $x'_u < 0$. This approximation of the probability of failure improves as the scheduling window discretisation coarseness decreases (*i.e.* as $|\mathcal{P}|$ increases).

From (4.2), the probability that a unit will survive between periods x'_u and $\sum_{p \in \mathcal{P}} px_{u,p}$, *i.e.* its reliability, is approximated by

$$R(x_{u,p}) = 1 - F(x_{u,p}) = 1 - \left(1 - \exp \left(-\lambda_u \left(\sum_{p \in \mathcal{P}} px_{u,p} - x'_u \right) \right) \right) = \exp \left(-\lambda_u \left(\sum_{p \in \mathcal{P}} px_{u,p} - x'_u \right) \right).$$

A schematic representation of this reliability is shown in Figure 4.2.

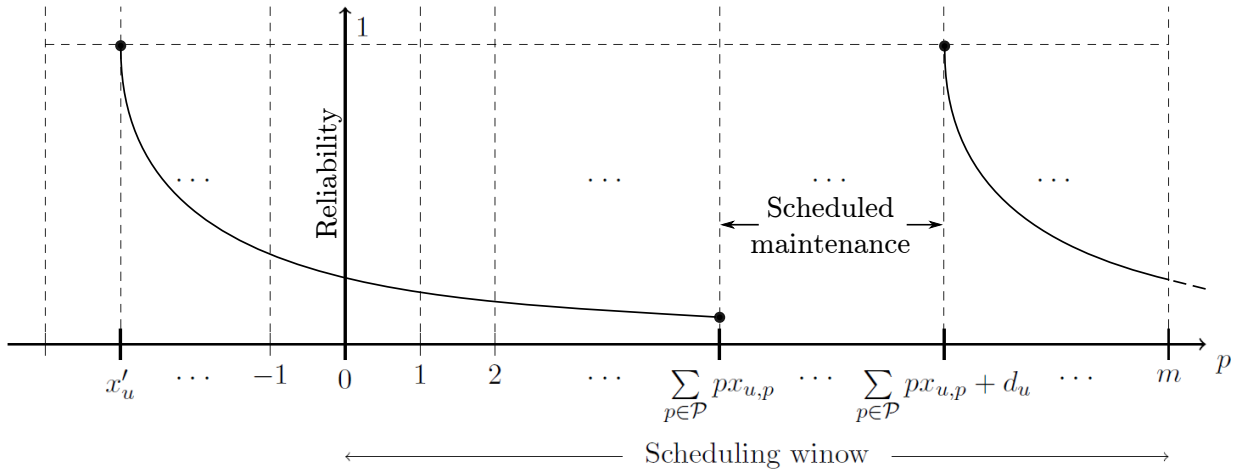


FIGURE 4.2: Reliability of PGU u with the previous maintenance and next maintenance periods indicated.

Assuming independence between the events of different PGUs failing, as per Assumption 12, the probability that no PGU will fail by the end of the scheduling window may therefore be approximated by

$$\prod_{u \in \mathcal{U}} \exp \left(-\lambda_u \left(\sum_{p \in \mathcal{P}} px_{u,p} - x'_u \right) \right). \quad (4.3)$$

Maximising the probability of no unit failing during the scheduling window, expressed by the function in (4.3), is equivalent to minimising

$$\prod_{u \in \mathcal{U}} \frac{1}{\exp \left(-\lambda_u \left(\sum_{p \in \mathcal{P}} px_{u,p} - x'_u \right) \right)}. \quad (4.4)$$

A problem with (4.4) as potential objective function for GMS, is that PGUs which contribute a larger portion to the overall capacity of the power generating system do not receive any preference in terms of earlier maintenance urgency. In its current form, the function in (4.4) only promotes PGUs with the large failure rates in terms of earlier maintenance preference. In order to adapt the function in (4.4) so as to take into account the rated capacity of each PGU in

the system, the various factors in the product may be weighted by normalised weights relative to the PGU with the largest capacity so as to arrive at the function

$$\begin{aligned} Z(\mathbf{X}) &= \prod_{u \in \mathcal{U}} \left[\frac{1}{\exp \left(-\lambda_u \left(\sum_{p \in \mathcal{P}} p x_{u,p} - x'_u \right) \right)} \right]^{\frac{C_u}{C_{\max}}}, \\ &= \prod_{u \in \mathcal{U}} \left[\exp \left(-\lambda_u \left(\sum_{p \in \mathcal{P}} p x_{u,p} - x'_u \right) \right) \right]^{-\frac{C_u}{C_{\max}}}, \end{aligned} \quad (4.5)$$

where $\mathbf{X} = [x_{u,p}]_{u \in \mathcal{U}, p \in \mathcal{P}}$ is the matrix of decision variables, C_u is the rated capacity of PGU $u \in \mathcal{U}$ and C_{\max} is the rated capacity of the PGU with the largest capacity in the system, and where the objective is to minimise (4.5). The function in (4.3) represents the probability that no unit will fail during the scheduling window, which is a value between zero and one. By taking the inverse of this function, as shown in (4.4), a value is obtained that is larger than unity, which can be exponentiated as in (4.5), in order to obtain the desired outcome. This process of exponentiation brings with it the added benefit that the function $Z(\mathbf{X})$ in (4.5) may easily be linearised, thus facilitating the efficient computation of optimal GMS solutions when adopting PGU reliability as objective. This may be achieved by adopting

$$\begin{aligned} Y(\mathbf{X}) &= \ln Z(\mathbf{X}), \\ &= \sum_{u \in \mathcal{U}} -\frac{C_u}{C_{\max}} \ln \left[\exp \left(-\lambda_u \left(\sum_{p \in \mathcal{P}} p x_{u,p} - x'_u \right) \right) \right], \\ &= \sum_{u \in \mathcal{U}} \frac{C_u}{C_{\max}} \left[\lambda_u \left(\sum_{p \in \mathcal{P}} p x_{u,p} - x'_u \right) \right], \end{aligned} \quad (4.6)$$

as the GMS objective, to be minimised, instead of (4.5) in view of the monotonic growth property of the natural logarithm.

It is acknowledged that the method of weighting the reliability function by means of exponentiation by the ratio C_u/C_{\max} is, in fact, merely an example of a form of utility function for a decision maker. The author concedes that in the current form, certain preferences regarding the relative importance of the reliability and rated capacity are imposed on the decision maker. If the newly proposed GMS criterion were to be implemented within a tool for scheduling maintenance at a specific power utility, the method of weighting the reliability by a function f of the aforementioned ratio may have been achieved by means of utility function elicitation. This function would typically then be a general utility function $f(C_u/C_{\max})$. In the current model, access to potential users was, however, not possible and for this reason, research was not conducted in respect of identifying a specific utility function f based on the decision makers' preferences.

From an interaction with a specific decision maker, valuable information and user preferences may be gathered in order to identify how much more important the rated capacity of a PGU is to a typical power utility in a developing context (say) than the reliability of the PGUs. This would provide insight into the shape of the utility function of the decision maker. This function can either be concave or convex so as to award a higher importance to either rated capacity or PGU reliability. As it was not possible to have this interaction in this study, the identity function was used, *i.e.* $f(C_u/C_{\max}) = C_u/C_{\max}$. In the case where a decision maker's actual utility function (other than the identity function) were to be incorporated into the model, a similar approach may clearly have been adopted to linearise the objective function. Although the resulting schedules would be different for different utility functions, the scheduling methodology is independent of the utility function (*i.e.* the difficulty of solving the model does not increase).

Maximising expected energy production

An alternative objective to that of minimising the weighted measure (4.6) of the probability that any PGU will fail, is to maximise the expected energy produced over the scheduling window, taking into account the occurrence of failures of the PGUs in the system. In order to achieve this, two assumptions are made, namely that no PGU will fail more than once during the scheduling window, as per Assumption 3 of §4.2, and that each PGU will be subjected to maintenance exactly once during the scheduling window, as per Assumption 2 of §4.2.

Let X_u be a random variable denoting the timing of the next failure of PGU $u \in \mathcal{U}$ after commencement of the current scheduling window. Adopting the convention that the scheduling window is the interval $\tau = [0, T]$ on the real line, it follows that $X_u \in [0, \infty)$ for each PGU $u \in \mathcal{U}$. It follows from Assumption 12 of §4.2 that the PDF for the random variable X_u is

$$f(X_u) = \lambda_u e^{-\lambda_u X_u}, \quad (4.7)$$

where λ_u is again the failure rate of PGU $u \in \mathcal{U}$. The expected failure time $E(X_u)$ and the standard deviation $\sigma(X_u)$ of the inter-failure times for PGU $u \in \mathcal{U}$ are therefore both

$$E(X_u) = \sigma(X_u) = \frac{1}{\lambda_u}. \quad (4.8)$$

Let $x_{u,p}$ again be a binary decision variable taking the value 1 if planned maintenance of PGU $u \in \mathcal{U}$ is scheduled to start during time period $p \in \mathcal{P}$, or zero otherwise. Let $k_u = \sum_{p \in \mathcal{P}} p x_{u,p} \in \mathcal{P}$ denote the time period during which maintenance is scheduled to commence within the scheduling window τ for PGU $u \in \mathcal{U}$.

There are three cases to consider:

- I A failure occurs before planned maintenance is performed on the PGU and maintenance is either performed when the failure occurs or sometime thereafter, *i.e.* $0 \leq X_u \leq k_u$,
- II maintenance is performed before the next failure of the PGU, thus in effect postponing the failure to some time which is still within the scheduling window, *i.e.* $k_u < X_u \leq T$, or
- III maintenance is performed before the next failure of the PGU, but the failure occurs after the scheduling window has ended, *i.e.* $T < X_u$.

The continuous variable X_u may again be viewed within the discretised context of Figure 4.1. Let $n(k_u, X_u)$ denote the amount of energy generated by PGU $u \in \mathcal{U}$ during the scheduling window τ if PGU u is subjected to maintenance from period $k_u \in \mathcal{P}$ to period $k_u + d_u - 1$, given that a failure of PGU u occurs during period X_u . Then the expected energy produced by PGU $u \in \mathcal{U}$ during the scheduling window τ may be determined as

$$E(n(k_u, X_u)) = \int_0^{\infty} n(k_u, t) f(t) dt, \quad (4.9)$$

where $f(t)$ is the PDF of the failure model, specifying the probability $P(X_u = t)$ of failure of PGU $u \in \mathcal{U}$ at time $t \in \tau$. The energy produced in the above three cases is

$$n(k_u, X_u) = \begin{cases} C_u X_u + C_u(T - (k_u + d_u - 1)), & \text{if } 0 \leq X_u \leq k_u \text{ (Case I),} \\ C_u k_u + C_u(X_u - (k_u + d_u - 1)), & \text{if } k_u < X_u \leq T \text{ (Case II),} \\ C_u k_u + C_u(T - (k_u + d_u - 1)), & \text{if } k_u < T < X_u \text{ (Case III),} \end{cases} \quad (4.10)$$

where d_u is the duration of maintenance performed on PGU $u \in \mathcal{U}$ and T is the length of the scheduling window, as illustrated in Figure 4.3. The expected energy produced by PGU $u \in \mathcal{U}$

during the scheduling window is therefore

$$\begin{aligned}
 E(n(k, X_u)) &= \int_0^{k_u} [C_u t + C_u(T - (k_u + (d_u - 1)))] \lambda_u e^{-\lambda_u(t-x'_u)} dt \\
 &+ \int_{k_u+(d_u-1)}^T [C_u k_u + C_u(t - (k_u + (d_u - 1)))] \lambda_u e^{-\lambda_u[t-(k_u+(d_u-1))]} dt \\
 &+ [C_u k_u + C_u(T - (k_u + (d_u - 1)))] \int_T^\infty \lambda_u e^{-\lambda_u[t-(k_u+(d_u-1))]} dt, \\
 &= \frac{C_u e^{-\lambda_u(k_u-x'_u)} [(d_u - 1)\lambda_u - T\lambda_u - 1 + e^{k_u\lambda_u}(1 - \lambda_u((d_u - 1) + k_u - T))]}{\lambda_u} \\
 &+ \frac{C_u [1 + k_u\lambda_u + e^{-\lambda_u[T-(k_u+(d_u-1))]}] ((d_u - 1)\lambda_u - T\lambda_u - 1)}{\lambda_u} \\
 &+ C_u e^{-\lambda_u[T-(k_u+(d_u-1))]} [T - (d_u - 1)], \tag{4.11}
 \end{aligned}$$

where $x'_u < 0$ is again a pre-determined parameter denoting the time period during which PGU $u \in \mathcal{U}$ was placed back into operation after having carried out its previous maintenance operation. In Case I (the first term in (4.11)), energy is only available from the beginning of the scheduling window, when PGU u is assumed to be in operating condition, to the time when the failure occurs and then again from when scheduled maintenance has been completed until the end of the scheduling window, as shown in Figure 4.3(a).

In Case II (the second term in (4.11)), energy is only available from the beginning of the scheduling window, when the PGU is assumed to be in operating condition, to when the maintenance is scheduled to start on PGU u and then again from the completion of the scheduled maintenance until when the failure occurs, as shown in Figure 4.3(b). As it is assumed that maintenance is only performed once during the current scheduling window, no energy will be generated by the PGU after the time at which the failure occurs (up to the end of the scheduling window) in Case II.

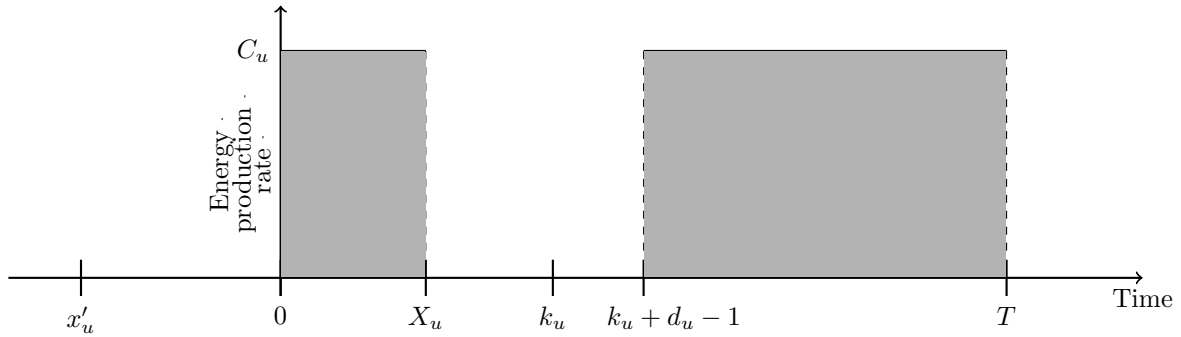
In Case III (the third term in (4.11)), energy is available from the beginning of the scheduling window, when the PGU is assumed to be in operating condition, to when the maintenance is scheduled to start on PGU u and then from the completion of the scheduled maintenance until the end of the scheduling window, as shown in Figure 4.3(c).

The expected energy produced by the entire system of PGUs during the scheduling window is therefore

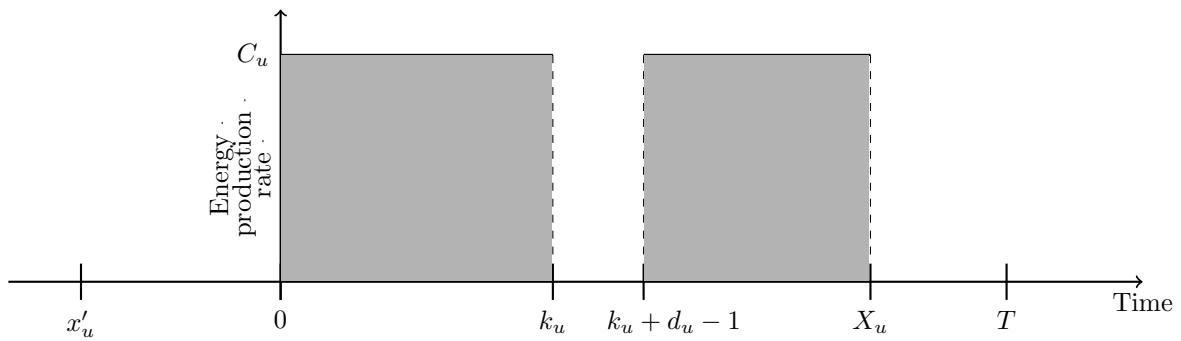
$$\begin{aligned}
 \sum_{u \in \mathcal{U}} &\left[\frac{C_u e^{-\lambda_u(k_u-x'_u)} [(d_u - 1)\lambda_u - T\lambda_u - 1 + e^{k_u\lambda_u}(1 - \lambda_u((d_u - 1) + k_u - T))]}{\lambda_u} \right. \\
 &+ \frac{C_u [1 + k_u\lambda_u + e^{-\lambda_u[T-(k_u+(d_u-1))]}] ((d_u - 1)\lambda_u - T\lambda_u - 1)}{\lambda_u} \\
 &\left. + C_u e^{-\lambda_u[T-(k_u+(d_u-1))]} [T - (d_u - 1)] \right], \tag{4.12}
 \end{aligned}$$

which may be maximised as a maintenance scheduling criterion. This objective function is nonlinear, which makes it cumbersome to optimise exactly. Each term in the sum (4.12) may, however, be approximated by a piecewise linear function. This approximation process will result in the introduction of additional binary variables (and hence yield a larger model), but the benefit of the process is that it results in a linear optimisation model which is easier to solve exactly.

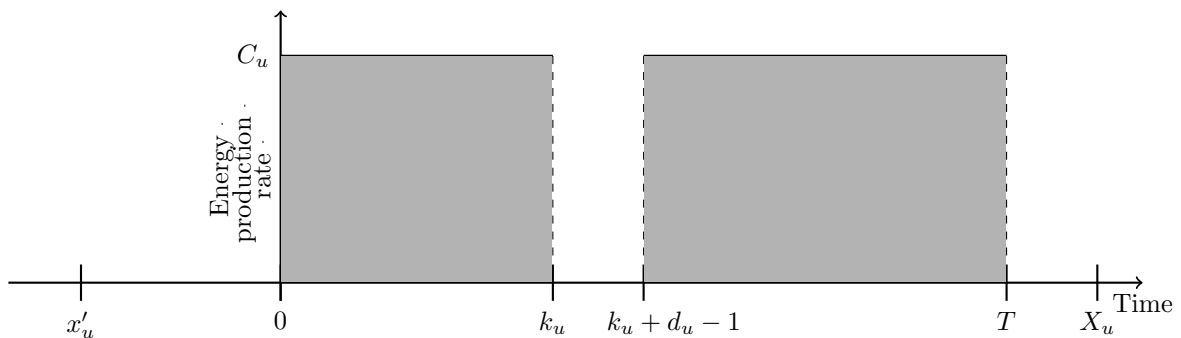
As a result of the increase in the complexity of the GMS model brought about by the introduction of the linearisation binary variables referred to above, an exact solution approach may not always



(a) The energy produced by PGU $u \in \mathcal{U}$ when a failure occurs during time period $X_u \in \mathcal{P}$ before maintenance of PGU u is performed from time period $k_u \in \mathcal{P}$ onwards in the case where maintenance is either performed when the failure occurs ($k_u = X_u$) or sometime afterwards ($k_u > X_u$), i.e. Case I. The total energy produced during the scheduling window is represented by the shaded area.



(b) The energy produced by PGU $u \in \mathcal{U}$ when a failure occurs during time period $X_u \in \mathcal{P}$ after maintenance of PGU u is performed from time period $k_u \in \mathcal{P}$ onwards in the case where maintenance is performed before a failure occurs, but the failure occurs within the scheduling window ($k_u < X_u < T$), i.e. Case II. The total energy produced during the scheduling window is represented by the shaded area.



(c) The energy produced by PGU $u \in \mathcal{U}$ when a failure occurs during time period $X_u \in \mathcal{P}$ after maintenance of PGU u is performed from time period $k_u \in \mathcal{P}$ onwards in the case where maintenance is performed before a failure occurs, but the failure occurs after the scheduling window has ended ($k_u < T < X_u$), i.e. Case III. The total energy produced during the scheduling time window is represented by the shaded area.

FIGURE 4.3: Graphical representation of the expected energy produced for the three different cases.

be carried out successfully within a reasonable amount of time. This necessitates the design of a metaheuristic solution approach, such as the method of SA, which does not require a linear objective function.

An advantage of adopting the scheduling criterion in (4.12) is that it may be derived mathematically from first principles to obtain an intuitively understandable measure of maintenance schedule desirability (measured in easily interpreted units if expected energy produced during the scheduling window), whereas the scheduling criterion (4.6) of minimising a weighted measure of probability that a PGU within the system will fail is expected to be a less intuitive measure for a maintenance planning manager.

4.3.3 The model constraints

In the models considered in this dissertation, four classes of constraints are included: energy demand satisfaction constraints for the system as a whole, maintenance window constraints for each of the PGUs, maintenance crew availability constraints and exclusion constraints ensuring that certain subsets of PGUs are not all scheduled for simultaneous planned maintenance.

Let e_u and ℓ_u denote respectively the earliest starting time period and latest starting time period for planned maintenance of PGU $u \in \mathcal{U}$. Recall that planned maintenance of a PGU may only be scheduled once during the scheduling window according to Assumption 2 of §4.2. The maintenance window constraints are therefore formulated as

$$\sum_{p=e_u}^{\ell_u} x_{u,p} = 1, \quad u \in \mathcal{U}. \quad (4.13)$$

The GMS models of this dissertation are formulated in such a manner that the maintenance window constraints are specified as a hard constraint set, which means that a unit may not be scheduled for planned maintenance outside of its allowable maintenance window. Therefore,

$$x_{u,p} = 0, \quad \text{for all } p < e_u \text{ and all } p > \ell_u, \quad u \in \mathcal{U}. \quad (4.14)$$

Recall that d_u is the duration of planned maintenance on PGU $u \in \mathcal{U}$. Clearly, therefore,

$$y_{u,p} = 0, \quad \text{for all } p < e_u \text{ and all } p > \ell_u + d_u - 1, \quad u \in \mathcal{U}. \quad (4.15)$$

The required duration of planned maintenance on each PGU is incorporated into the model by requiring that

$$\sum_{p=e_u}^{\ell_u+d_u-1} y_{u,p} = d_u, \quad u \in \mathcal{U}. \quad (4.16)$$

In order to enforce contiguity of the planned maintenance periods scheduled for PGU $u \in \mathcal{U}$, the linking constraints

$$\begin{aligned} x_{u,p} &\geq y_{u,p} - y_{u,p-1}, \quad u \in \mathcal{U}, \quad p \in \mathcal{P} \setminus \{1\}, \\ x_{u,1} &\geq y_{u,1}, \quad u \in \mathcal{U} \end{aligned} \quad (4.17)$$

between the auxilliary and decision variables are also included in the model formulations according to Assumption 4.

Let D_p denote the demand for power available from the entire system of PGUs during time period $p \in \mathcal{P}$, and let S denote the safety margin specified for the system (a proportion of

the total load requirement of the system over and above the demand). Then the load demand constraint may be formulated as

$$D_p(1 + S) \leq \sum_{u=1}^n C_u(1 - y_{u,p}), \quad p \in \mathcal{P}. \quad (4.18)$$

The resources required during each time period of planned maintenance is not necessarily the same, as per Assumption 9. It is assumed that during the i -th time period of planned maintenance of a PGU $u \in \mathcal{U}$, the unit requires Ψ_u^i resources. Furthermore, let $\psi_{u,p,v}$ denote the resources required for planned maintenance on PGU $u \in \mathcal{U}$ during time period p if planned maintenance were scheduled to start during time period $v \in \mathcal{P}$. The resource requirement parameters $\psi_{u,p,v}$ are calculated as

$$\psi_{u,p,v} = \begin{cases} \Psi_u^{p-v+1}, & \text{if } 0 \leq p - v < d_u, \\ 0, & \text{otherwise.} \end{cases} \quad (4.19)$$

The resource availability constraint may then be formulated as

$$\sum_{u=1}^n \sum_{v=1}^p \psi_{u,p,v} x_{u,v} \leq M_p, \quad p \in \mathcal{P}, \quad (4.20)$$

where M_p denotes the pre-specified number of available resources during time period $p \in \mathcal{P}$.

The models also allow for the incorporation of exclusion constraints. That is, the specification of sets $\mathcal{J}_1, \dots, \mathcal{J}_w$ of generating units which may not all be scheduled simultaneously for planned maintenance. Let I_i denote the maximum number of PGUs that are allowed to be in simultaneous maintenance during any time period in exclusion set \mathcal{J}_i . Then the exclusion constraints may be formulated as

$$\sum_{u \in \mathcal{J}_i} y_{u,p} \leq I_i, \quad p \in \mathcal{P}, \quad i \in \{1, \dots, w\}. \quad (4.21)$$

The final constraints required in the models are constraints specifying the binary nature of the variables, that is

$$x_{u,p}, y_{u,p} \in \{0, 1\}, \quad u \in \mathcal{U}, \quad p \in \mathcal{P}. \quad (4.22)$$

4.4 Chapter summary

This chapter contains two novel mathematical model formulations for the GMS problem. A motivation was presented in §4.1 for the selection of the objective functions in (4.6) and (4.12) as scheduling criteria. In §4.2, a number of assumptions were made in order to facilitate the model derivations. More specifically, assumptions were made about the PGU components, the frequency of maintenance performed on PGUs, the contiguity of such maintenance performance, the initial conditions of PGUs in the system in terms of their reliability levels, their reliability immediately after a maintenance procedure has been performed, the exclusion of transmission line maintenance and accompanying constraints, the resources required to perform maintenance, the nature of the power generating system, the failure rates of the PGUs and the independence of failures of the PGUs. This was followed in §4.3 by an introduction of the decision and auxiliary variables employed in the mathematical model as well as the mathematical representation of the GMS objective functions and the constraints of the models. The model constraints included energy demand satisfaction constraints, maintenance window constraints, maintenance resource constraints and maintenance exclusion constraints.

In summary, the two GMS models considered in this dissertation involve either

$$\text{minimising } \sum_{u \in \mathcal{U}} \frac{C_u}{C_{\max}} \left[\lambda_u \left(\sum_{p \in \mathcal{P}} p x_{u,p} - x'_u \right) \right], \quad (4.23)$$

or

$$\begin{aligned} \text{maximising } \sum_{u \in \mathcal{U}} & \left[\frac{C_u e^{-\lambda_u(k_u - x'_u)} [(d_u - 1)\lambda_u - T\lambda_u - 1 + e^{k_u \lambda_u} (1 - \lambda_u((d_u - 1) + k_u - T))]}{\lambda_u} \right. \\ & + \frac{C_u [1 + k_u \lambda_u + e^{-\lambda_u [T - (k_u + (d_u - 1))]} ((d_u - 1)\lambda_u - T\lambda_u - 1)]}{\lambda_u} \\ & \left. + C_u e^{-\lambda_u [T - (k_u + (d_u - 1))]} [T - (d_u - 1)] \right], \quad (4.24) \end{aligned}$$

both subject to the constraints

$$\sum_{p=e_u}^{\ell_u} x_{u,p} = 1, \quad u \in \mathcal{U}, \quad (4.25)$$

$$x_{u,p} = 0, \quad \text{for all } p < e_u \text{ and all } p > \ell_u, \quad u \in \mathcal{U}, \quad (4.26)$$

$$y_{u,p} = 0, \quad \text{for all } p < e_u \text{ and all } p > \ell_u + d_u - 1, \quad u \in \mathcal{U}, \quad (4.27)$$

$$\sum_{p=e_u}^{\ell_u + d_u - 1} y_{u,p} = d_u, \quad u \in \mathcal{U}, \quad (4.28)$$

$$\begin{aligned} x_{u,p} & \geq y_{u,p} - y_{u,p-1}, & u \in \mathcal{U}, \quad p \in \mathcal{P} \setminus \{1\}, \\ x_{u,1} & \geq y_{u,1}, & u \in \mathcal{U}, \end{aligned} \quad (4.29)$$

$$D_p(1 + S) \leq \sum_{u=1}^n C_u(1 - y_{u,p}), \quad p \in \mathcal{P}, \quad (4.30)$$

$$\psi_{u,p,v} = \begin{cases} \Psi_u^{p-v+1}, & \text{if } 0 \leq p - v < d_u, \\ 0, & \text{otherwise,} \end{cases} \quad (4.31)$$

$$\sum_{u=1}^n \sum_{v=1}^p \psi_{u,p,v} x_{u,v} \leq M_p, \quad p \in \mathcal{P}, \quad (4.32)$$

$$\sum_{u \in \mathcal{J}_i} y_{u,p} \leq I_i, \quad p \in \mathcal{P}, \quad i \in \{1, \dots, w\}, \quad (4.33)$$

$$x_{u,p}, y_{u,p} \in \{0, 1\}, \quad u \in \mathcal{U}, \quad p \in \mathcal{P}. \quad (4.34)$$

CHAPTER 5

Model solution approaches

Contents

5.1	Linearisation of the model of §4.3	79
5.2	Exact model solution approach	83
5.3	Approximate model solution approach	86
5.4	Chapter summary	97

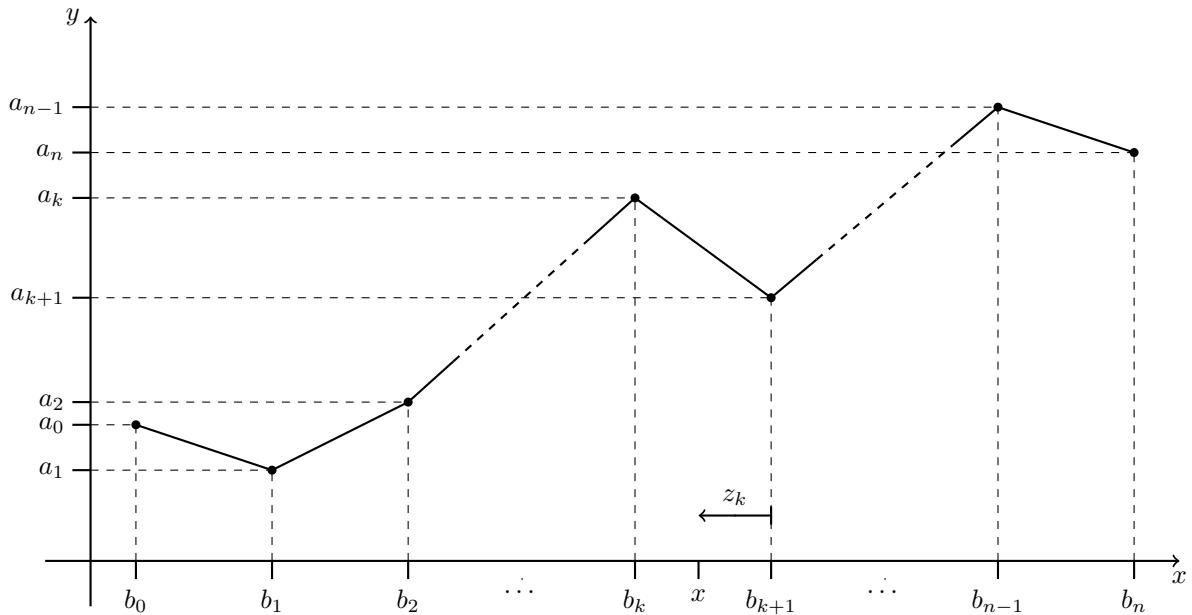
The model solution methodologies adopted in this dissertation are described in some detail in this chapter. An approach towards linearising the nonlinear model proposed in §4 is first presented in §5.1. Two methodologies are adopted to solve each of the models in §4.3 — an “exact” solution approach applied to a linearisation of the nonlinear model and a metaheuristic (approximate) solution approach. For the “exact” solution approach, the IBM ILOG CPLEX Optimizer Studio [112, 115] is employed as described in §5.2, whereas for the approximate solution approach, the method of SA is implemented in R [175, 184], as described in §5.3. The chapter closes with a brief summary of its contents in §5.4.

5.1 Linearisation of the model of §4.3

In this section, the method employed to linearise the nonlinear GMS model proposed in §4.3 is described in some detail. This linearisation involves approximating the nonlinear model objective by an appropriate piecewise linear function. A method for computing optimal positions for the breakpoints of the piecewise linearisation is also presented in this section.

5.1.1 Piecewise linear function approximation

In order to linearise the nonlinear model of §4.3, piecewise linear functions are employed to approximate the nonlinear objective function (4.12) by sampling certain points close to the curve and interpolating linearly between these points. The sampled points are called *breakpoints* and the approximating piecewise linear function consists of several straight line segments between these breakpoints [218]. Although a piecewise linear function itself is nonlinear in its entirety, binary variables may be used in the model formulation to represent the function in linear form locally. Consider an n -segment piecewise linear function $y = y(x)$ on $n + 1$ breakpoints b_0, b_1, \dots, b_n , along the x -axis and suppose the function value at breakpoint b_k is a_k , as shown in Figure 5.1.


 FIGURE 5.1: A piecewise linear function $y = y(x)$ on $n + 1$ breakpoints b_0, \dots, b_n .

Then it follows that the function $y = y(x)$ may be expressed as

$$y = \frac{a_{k+1}(x - b_k) - a_k(x - b_{k+1})}{b_{k+1} - b_k} \quad \text{if} \quad b_k \leq x \leq b_{k+1}, \quad (5.1)$$

for each $k = 0, 1, \dots, n - 1$. The if-condition in (5.1) becomes $1 \geq z_k \geq 0$ if the variable x is rewritten as

$$x = z_k b_k + (1 - z_{k+1}) b_{k+1}, \quad (5.2)$$

in terms of n new (real) variables z_1, \dots, z_n . Note that $x = b_k$ in (5.2) if $z_k = 1$ and that $x = b_{k+1}$ if $z_k = 0$. As a result of the linear combination in (5.2), the new variable z_k therefore represents the magnitude by which the breakpoint b_{k+1} exceeds x if the condition in (5.1) is satisfied, in which case

$$y = z_k a_k + (1 - z_k) a_{k+1}. \quad (5.3)$$

The if-condition in (5.1) may be accommodated algebraically by the introduction of a further $n - 1$ binary variables $\zeta_1, \dots, \zeta_{n-1}$ which satisfy the constraints

$$z_1 \leq \zeta_1, \quad (5.4)$$

$$z_2 \leq \zeta_2 + \zeta_3, \quad (5.5)$$

$$z_3 \leq \zeta_3 + \zeta_4, \quad (5.6)$$

$$\vdots$$

$$z_{n-1} \leq \zeta_{n-2} + \zeta_{n-1}, \quad (5.7)$$

$$z_n \leq \zeta_{n-1}, \quad (5.8)$$

$$\zeta_1 + \dots + \zeta_{n-1} = 1 \quad (5.9)$$

if

$$z_1 + \dots + z_n = 1 \quad (5.10)$$

and

$$x = z_1 b_1 + z_2 b_2 + \cdots + z_n b_n. \quad (5.11)$$

Constraint (5.9) ensures that exactly one of the binary variables $\zeta_1, \dots, \zeta_{n-1}$ assumes the value 1 which, in conjunction with (5.4)–(5.8), ensures that at most two of the variables z_1, \dots, z_n are positive. Furthermore, these two variables have consecutive indices, and the values of these variables sum up to 1 by (5.10). Therefore, constraint (5.11) reduces to (5.2).

In summary, the equation of the piecewise linear function $y = y(x)$ in Figure 5.1 is hence given by

$$y(x) = \sum_{i=1}^n a_i z_i, \quad (5.12)$$

subject to the constraints

$$\sum_{i=1}^n b_i z_i = x, \quad (5.13)$$

$$\sum_{i=1}^n z_i = 1, \quad (5.14)$$

$$\sum_{i=1}^{n-1} \zeta_i = 1, \quad (5.15)$$

$$z_1 \leq \zeta_1, \quad (5.16)$$

$$z_i \leq \zeta_{i-1} + \zeta_i, \quad i = 2, \dots, n-2 \quad (5.17)$$

$$z_n \leq \zeta_{n-1}, \quad (5.18)$$

$$\zeta_i \in \{0, 1\}, \quad i = 1, \dots, n-1, \quad (5.19)$$

$$z_i \geq 0, \quad i = 1, \dots, n. \quad (5.20)$$

5.1.2 Breakpoint selection during piecewise linear function approximation

The piecewise linear approximation method described in §5.1.1 may be applied to the (nonlinear) expected energy production function of each PGU in the objective function (4.12) of the nonlinear GMS model over the scheduling window $[1, T]$. This scheduling window contains $T-1$ distinct possible starting times $1, \dots, T-1$ at which the internal breakpoints b_1, \dots, b_{n-1} in (5.11) may be positioned. It may, of course, be assumed without loss of generality that $b_n = T$. Two questions arise naturally at this point: (1) What should the value of n be in order to achieve a close approximation of a PGU expected energy production function? (2) Given that a suitable value of n has been decided upon, what are the optimal positions for the internal breakpoints b_1, \dots, b_{n-1} ? The answer to the former question depends on the desired degree of closeness of the piecewise linear approximation of the expected energy production function. Once the first question has, however, been answered, the second question is an optimisation problem that can be solved by means of dynamic programming (see §2.3.1). This section is devoted to a description of how this optimisation process may be carried out.

The desirability of the locations of a given set of internal breakpoints b_1, \dots, b_{n-1} may be quantified by assessing the piecewise linear approximation of the expected energy production function of a PGU with respect to a regression model of the form (5.1). The positioning of these breakpoints may therefore be maximised by minimising the *sum of squared residuals* (SSR), described in §3.7.2, of the aforementioned linear regression model for each of the resulting n line

segments [15]. An algorithm for this purpose was developed by Bai and Perron [14, 15]. This regression problem was previously considered by Bellman and Roth [24] as well as by Guthery [102]. Their work was, however, extended in 1997 by Bai and Perron [15] to accommodate multiple regression models and partial structural changes to the original model.

The method is initialised by constructing a $T \times T$ upper-triangular matrix of SSR of all the possible line segments in a piecewise linear approximation of a PGU expected energy production function. The rows of the matrix represent the possible starting dates corresponding to the line segment, while its columns represent the possible ending dates corresponding to the line segment. The entry in row i and column j of this matrix, denoted by $SSR(i, j)$, represents the SSR of an approximate line segment starting at time period i and ending at time period j . The structure of such a triangular matrix is shown in Table 5.1.

TABLE 5.1: *The triangular matrix containing the SSR for line segments starting at date i and ending at date j .*

		Terminal date				
		1	2	3	...	T
Starting date	1	$SSR(1, 1)$	$SSR(1, 2)$	$SSR(1, 3)$...	$SSR(1, T)$
	2		$SSR(2, 2)$	$SSR(2, 3)$...	$SSR(2, T)$
	3			$SSR(3, 3)$...	$SSR(3, T)$
	⋮				⋱	⋮
	T					$SSR(T, T)$

The upper-triangular matrix in Table 5.1 is constructed by means of a standard updating formula which calculates the recursive residuals¹ on a segment-by-segment basis. Let $v(i, j)$ be the recursive residual at time j using a sample of the observations starting at time i . Then the recursive relationship

$$SSR(i, j) = SSR(i, j - 1) + v(i, j)^2$$

holds [33], which may be used to populate the matrix. Once the matrix has been constructed with the relevant SSR contribution calculated for each line segment, a dynamic programming algorithm is employed to evaluate which combination of line segments achieves a global minimum of SSR.

Suppose the minimum length of an approximating line segment is h and let $SSR(\{P_{r,k}\})$ be the SSR associated with an optimal piecewise linear approximation containing r internal breakpoints by sampling the first k observations² in the data set. Then the global SSR for any number of line segments is a linear combination of the entries in the upper-triangular matrix of Table 5.1 [14]. An optimal partition is therefore a solution to the recursive problem

$$SSR(\{P_{n,T}\}) = \min_{nh \leq j \leq T-h} [SSR(\{P_{n-1,j}\}) + SSR(j, T)]. \quad (5.21)$$

The procedure commences by evaluating all the sub-samples that allow one possible breakpoint ranging from observation h to $T - nh$ in order to obtain a piecewise linear approximation with one internal breakpoint. During this step, the SSR of $T - (n - 1)h + 1$ optimal breakpoint partitions are calculated and stored. Each of these values has a corresponding ending date ranging from $2h$ and $T - (n - 1)h$ (inclusive). The next step is to obtain an optimal partitioning with two breakpoints, each of which each has a corresponding ending date ranging between $3h$

¹A recursive residual is the difference between a true value and an estimated value, but these residuals are only calculated for a certain number of observations in sequence [156].

²Here an observation corresponds to the dates within the maintenance scheduling window.

and $T - (n - 2)h$ (inclusive). This is achieved by considering all the possible one-breakpoint partitions in order to insert another breakpoint that will achieve a minimum SSR. In this way an optimal piecewise linear approximation containing two internal breakpoints is obtained. This forward recursive procedure is repeated until a set of $T - (n + 1)h + 1$ optimal internal breakpoints has been obtained, each of which has corresponding ending dates ranging from $(n - 1)h$ and $T - 2h$ (inclusive).

Although the algorithm executes very quickly, the majority of the computational time is attributed to the construction of the upper-triangular matrix of Table 5.1 [14]. This procedure is employed later in this dissertation to linearise the model (4.12). The procedure is implemented by invoking an R package called `strucchange` which contains a function called `breakpoints`. This function computes the optimal positions of the internal breakpoints b_1, \dots, b_{n-1} , given a value of n , according to the method described in this section [223].

5.2 Exact model solution approach

As described in §4, the GMS problem may be formulated as a mathematical program in binary form. This poses the possibility of finding optimal solutions to problem instances of the GMS problem by means of mathematical programming solution techniques, as described in §2.3.1.

Because of the novelty of objective functions in the models of §4.3 within the GMS domain, approximate solutions obtained by solving even the linear model of §4.3 by means of fast heuristic or metaheuristic cannot be compared to results previously obtained in the literature. Hence mathematical programming solution techniques are employed in an attempt to obtain optimal solutions to instances of the linear model in §4.3. The same approach is followed for a linearised version of the nonlinear model of §4.3 (although an exact solution to the linearised model is, of course, not necessarily an optimal solution to the full nonlinear model). Even if an exact solution technique fails to uncover optimal solutions to these mathematical programming problems, it is usually possible to provide a guarantee on how far from optimal the best solutions uncovered are.

5.2.1 Motivation for the choice of optimisation platform

A number of off-the-shelf integer and MIP solvers exist for solving a wide range of mathematical programming models. `Lingo` is a modelling language that employs the `Lindo` [145] solver to solve mathematical programming problems and has been applied successfully in many practical problem instances [25, 58, 201].

Another modelling language, called the *open programming language* (OPL), often used to solve mathematical programming problems, is an off-the-shelf optimisation package called `CPLEX` [112, 115]. `CPLEX` is a powerful solver capable of solving a wide range of problems (such as integer programming problems, large linear programming problems, convex and non-convex quadratic programming problems and convex quadratically constrained problems). Modelling can also be carried out in other programming languages and be linked to `CPLEX` by means of certain add-ons. Due to the wide use of `CPLEX` in academia and industry, it is easy to find support for the implementation of mathematical programming models in `CPLEX`.

Another popular optimisation engine is the `Gurobi` optimisation package [101]. `Gurobi` can also be employed to solve a variety of mathematical programs, including linear programs, MIPs, mixed integer quadratic programs, *etc.* Like `CPLEX`, `Gurobi` is also considered one of the leading public benchmark solvers in terms of computation time [100].

Excel Solver is yet another high-level optimisation package available as an add-on to Microsoft Excel [158]. This package employs the simplex method to solve linear programming programs and the generalised reduced gradient algorithm to solve nonlinear smooth problems. The standard version of Excel Solver has a limit on the number of variables (200) and constraints (depending on whether a linear or nonlinear model is considered) [92]. An add-on to Microsoft Excel, developed by Frontline Solvers [91], is capable of handling between 2000 and 8000 decision variables, depending on the type of problem under consideration.

The Matlab Optimization Toolbox [154] also has mathematical programming capabilities. This toolbox is capable of solving a variety of optimisation problems such as linear programming problems, MIPs, quadratic programming problems, nonlinear optimisation problems, and nonlinear least squares problems. The toolbox may be used to find solutions to either discrete or continuous optimisation problems and can even solve multi-objective optimisation problems [154].

A number of other optimisation packages also exist, such as Fico Xpress [84], AIMMS [6] and various functions available in Python [174] among others.

After considering the advantages and disadvantages of the various optimisation packages available, CPLEX and Gurobi were found to be the two optimisation software suites best suited to GMS models in this dissertation. Both of these packages contain a number of useful features and the performance of the two packages are similar in terms of computation time and model implementation ease. CPLEX was, however, chosen as the optimisation package employed for solving instances of the GMS models of §4.3.

5.2.2 CPLEX implementation

The linear GMS model derived in §4.3 is solved in this dissertation by means of the celebrated off-the-shelf software suite IBM ILOG CPLEX Optimization Studio [112, 115], also referred to colloquially as CPLEX. The name is derived from the simplex method which was initially implemented in the programming language C within this software suite. Recently, however, the suite has been updated to support other types of mathematical programming solution techniques and constructs, such as quadratic programming techniques, quadratically constrained programming algorithms, network flow algorithms, other MIP techniques and various heuristics [114].

In order to solve MIP models, CPLEX employs an algorithm that is based on the celebrated branch-and-cut method [138]. This algorithm may be customised as per the user's preference [113] which may result in the algorithm executing more effectively within the context of a particular problem instance. Emphasis may be placed on either proving optimality or achieving feasibility during the optimisation process without compromising the accuracy of the process. A useful functionality that CPLEX provides, is the possibility of specifying different methods of terminating the search process. These different termination criteria include limiting the processing time, limiting the size of the branching tree or specifying an acceptable tolerance associated with optimality [113]. The algorithms employed by CPLEX can generate thirteen different types of cutting planes, which depend on the nature of the model being solved [113].

Another useful aspect of the robust algorithms employed within the CPLEX optimiser, is that a lower bound on a minimisation objective function is calculated and presented. This lower bound may be interpreted as the best possible objective function value for which an integer feasible solution may potentially exist [113]. As the search for an optimal solution progresses, this value is constantly updated (increased). As the algorithm executes, a *relative gap* is calculated for each feasible solution obtained. This gap is the difference between the best lower bound thus

found and the upper bound on the minimum objective function value represented by the current solution, which is normalised and presented as a percentage [113]. One can interpret the relative gap as a guarantee on the difference between the objective function values of an optimal solution and the current best solution [114].

The implementation of the linear GMS model proposed in §4.3 is partitioned into three main parts of CPLEX code. The first is the initialisation of the model parameters and variables. The actual CPLEX implementation of parameters and variables coded for the purposes of this study is shown in Figure 5.2. The second part involves specifying the input data pertaining to the parameter values included in the model. This is achieved through data importation from an Excel spreadsheet which contains fields of data pertinent to the various parameters in the

```

//parameters
int n=...; // number of power generating units
int m=...; // number of planning periods
int o=...; // number of exclusion subsets

range u=1..n; // power generating unit range
range p=1..m; // planning period range
range v=1..m;
range w=1..o;

int C[u]=...; // capacity of each power generating unit
int d[u]=...; // duration of maintenance for each power generating unit
float e[u]=...; // earliest starting time for each power generating unit
float l[u]=...; // latest starting time for each power generating unit
float lam[u]=...; // lambda for failure model
int f[u][p][v]; // resource requirement for each power generating unit
int F[u][v]=...; // resource requirement for each power generating unit
int M[p]=...; // manpower available per planning period
int D[p]=...; // demand of system per week
int S[u]=...; // Previous service date of each power generating unit
int I[w]=...; // Min number of units allowed in simultaneous maintenance
int J1[w]=...; // First unit in exclusion set
int J2[w]=...; // Second unit in exclusion set

// variables
dvar boolean x[u][p];
dvar boolean y[u][p];

execute{ //generating 3d matrix for manpower requirements
for(var i in u){
  for(var j in p){
    for(var k in p){
      if(k-j<d[i] && k-j>=0){
        f[i][j][k] = F[i][k-j+1];
      }
      else
        {f[i][j][k] =0;}
    }
  }
}
}
}

```

FIGURE 5.2: Implementation of parameters and variables of the linear GMS model of §4.3 in CPLEX.

model. The code for achieving this in CPLEX is shown in Figure 5.3. The Excel spreadsheet should contain the required data (*e.g.* PGU capacities, required maintenance duration, earliest and latest starting dates, *etc.*) for the problem instance in columns. These columns should be named as shown in Figure 5.3 (*e.g.* Capacity, Duration, Earliest, Latest, *etc.*).

The final part of the model implementation in CPLEX involves the specification of the objective function and the accompanying constraints as described in §4.3.3. The actual CPLEX implementations of the linear objective function and model constraints are shown in Figure 5.4.

```
SheetConnection data("data.xlsx");
C from SheetRead(data,"Capacity");
d from SheetRead(data,"Duration");
e from SheetRead(data,"Earliest");
l from SheetRead(data,"Latest");
lam from SheetRead(data,"Alpha0");
D from SheetRead(data,"Demand");
M from SheetRead(data,"ManRequire");
S from SheetRead(data,"Previous");
F from SheetRead(data,"Manpower1");
I from SheetRead(data,"SimUnit");
J1 from SheetRead(data,"Exa");
J2 from SheetRead(data,"Exb");

SheetConnection results("results.xlsx");
x to SheetWrite(results,"x_result");
y to SheetWrite(results,"y_result");
```

FIGURE 5.3: *Parameter input specifications through Excel spreadsheets for implementation of the linear GMS model of §4.3 in CPLEX.*

Upon linearisation of the objective function in the nonlinear GMS model of §4.3, as described in §5.1, CPLEX may again be used to solve the model by means of its linear solver. Whereas the model constraints are in this case implemented exactly as elucidated in Figure 5.4, the alternative objective function implementation shown in Figure 5.5 is used instead. In this figure, the piecewise linear approximation for the 21-unit test system is presented as example. The optimal breakpoints shown in the figure are calculated in the following chapter.

5.3 Approximate model solution approach

In this section, the approximate solution methodology employed in this dissertation to solve instances of both the linear and nonlinear GMS models of §4.3 is described in some detail. The method of SA is employed for this purpose and was coded in R [175] in combination with RStudio [184]. R is an open source statistical programming language, modelled after the statistical language S, which provides the user with a variety of already implemented graphical and statistical techniques. Due to the open source nature of R, the global research community continually builds and extends libraries that are freely available to all R users and greatly enhance the functionality of the programming language. The language is also quality-validated by world industry leaders via Twitter [208], The New York Times [203] and Facebook [80]. R finally also provides the capability to experts to interpret, interact and visualise large quantities of data.

This section contains both a motivation for the choice of SA as the approximate solution methodology adopted in this dissertation and a description of the specific SA implementation employed.

```

// Objective function
maximize sum(i in u) ((C[i]/C_max) * (-lam[i] * ((sum(j in p) x[i][j]*j) - (S[i]-52))));

// Constraints

subject to {
  forall(i in u)
    Schedule_once:
      sum(j in p: e[i]<=j<=l[i]) x[i][j]==1;
  forall(i in u)
    Outside_window_x1:
      sum(j in p: j>l[i]) x[i][j]==0;
  forall(i in u)
    Outside_window_x2:
      sum(j in p: j<e[i]) x[i][j]==0;
  forall(i in u)
    Outside_window_y:
      sum(j in p: l[i]+d[i]-1<j<e[i]) y[i][j]==0;
  forall(i in u)
    Maintenance_duration:
      sum(j in p: e[i]<=j<=l[i]+d[i]-1) y[i][j]==d[i];
  forall(i in u)
    Contiguity1:
      x[i][1] >= y[i][1];
  forall(i in u, j in p: j>1)
    Contiguity2:
      y[i][j]-y[i][j-1]<=x[i][j];
  forall(j in p)
    Load_demand:
      D[j] <= sum(i in u) C[i]*(1-y[i][j]);
  forall(j in p)
    Manpower:
      sum(i in u) (sum(k in p) (f[i][k][j]*x[i][k])) <= M[j];
  forall(j in p, k in w)
    Exclusion:
      sum(i in u: J1[k]<=i<=J2[k]) y[i][j]<=I[k];
}

```

FIGURE 5.4: Implementation of the linear GMS model of §4.3 in CPLEX, including a specification of the linear objective function (4.6) and the model constraints (4.13)–(4.22).

```

// Objective function
maximize (6414.493)*z_1[1] + (537.5016*21+6414.493)*z_1[2] + (428.6654*35+8700.054)*z_1[3] + (211.0526*45+16316.5)*z_1[4]
+ (-159.867*52+33007.87)*z_1[5]
+ (6645.997)*z_2[1] + (923.6016*8+6645.997)*z_2[2] + (625.6345*19+9029.734)*z_2[3] + (406.255*33+13197.95)*z_2[4]
+ (182.4364*44+20583.96)*z_2[5] + (-149.87*52+35205.44)*z_2[6]
+ (2115.102)*z_3[1] + (195.717*15+2115.102)*z_3[2] + (159.8282*32+2653.433)*z_3[3] + (111.4623*44+4201.145)*z_3[4]
+ (27.59463*52+7891.32)*z_3[5]
+ (2103.345)*z_4[1] + (174.5276*27+2103.345)*z_4[2] + (136.8297*42+3121.188)*z_4[3] + (58.5225*52+6410.092)*z_4[4]
+ (7921.13)*z_5[1] + (1805.27*6+7921.13)*z_5[2] + (1150.547*13+11849.47)*z_5[3] + (658.2567*22+18249.24)*z_5[4]
+ (317.5491*33+25744.81)*z_5[5] + (19.0072*44+35596.69)*z_5[6] + (-397.01*52+53901.45)*z_5[7]
+ (7492.761)*z_6[1] + (615.604*27+7492.761)*z_6[2] + (458.1693*42+11743.5)*z_6[3] + (126.34*52+25680.33)*z_6[4]
+ (7579.267)*z_7[1] + (728.6752*14+7579.267)*z_7[2] + (567.0859*31+9841.518)*z_7[3] + (382.5833*43+15561.1)*z_7[4]
+ (65.72333*52+29186.08)*z_7[5]
+ (5569.836)*z_8[1] + (589.414*16+5569.836)*z_8[2] + (474.9068*35+7401.951)*z_8[3] + (257.099*45+15025.23)*z_8[4]
+ (-164.873*52+34013.96)*z_8[5]
+ (2422.859)*z_9[1] + (269.0689*22+2422.859)*z_9[2] + (223.5104*34+3425.146)*z_9[3] + (115.7113*42+7090.318)*z_9[4]
+ (-82.765*48+15426.32)*z_9[5] + (-371.374*52+29279.55)*z_9[6]
+ (1256.988)*z_10[1] + (136.1922*31+1256.988)*z_10[2] + (99.18231*44+2404.295)*z_10[3] + (9.38*52+6355.597)*z_10[4]
+ (598.0267)*z_11[1] + (89.1293*34+598.0267)*z_11[2] + (75.329*45+1067.237)*z_11[3] + (35.24743*52+2870.908)*z_11[4]
+ (568.816)*z_12[1] + (74.61745*33+568.816)*z_12[2] + (58.3195*45+1106.649)*z_12[3] + (11.10657*52+3231.23)*z_12[4]
+ (903.9351)*z_13[1] + (78.01329*17+903.9351)*z_13[2] + (66.39169*33+1101.502)*z_13[3] + (46.91136*44+1744.353)*z_13[4]
+ (13.62763*52+3208.838)*z_13[5]
+ (1097.388)*z_14[1] + (97.21856*17+1097.388)*z_14[2] + (80.24475*33+1385.943)*z_14[3] + (52.04836*44+2316.424)*z_14[4]
+ (2.459625*52+4498.329)*z_14[5]
+ (261.1722)*z_15[1] + (38.54894*34+261.1722)*z_15[2] + (31.54091*45+499.4451)*z_15[3] + (11.323*52+1409.251)*z_15[4]
+ (1320.841)*z_16[1] + (185.5885*33+1320.841)*z_16[2] + (152.7278*45+2405.247)*z_16[3] + (51.82929*52+6945.678)*z_16[4]
+ (738.4473)*z_17[1] + (247.6526*5+738.4473)*z_17[2] + (160.5082*11+1174.169)*z_17[3] + (86.54975*19+1987.712)*z_17[4]
+ (33.92609*30+2987.561)*z_17[5] + (-0.77792*42+4028.682)*z_17[6] + (-33.0708*52+5384.983)*z_17[7]
+ (572.6489)*z_18[1] + (76.78939*8+572.6489)*z_18[2] + (54.198*19+753.38)*z_18[3] + (37.98086*33+1061.506)*z_18[4]
+ (22.51309*44+1571.942)*z_18[5] + (0.287125*52+2549.885)*z_18[6]
+ (542.0859)*z_19[1] + (133.6791*46+542.0859)*z_19[2] + (80.3125*52+2996.951)*z_19[3]
+ (5756.701)*z_20[1] + (1048.657*7+5756.701)*z_20[2] + (647.305*17+8566.165)*z_20[3] + (353.3354*30+13563.65)*z_20[4]
+ (141.1531*43+19929.12)*z_20[5] + (-151.04*52+32493.42)*z_20[6]
+ (601.1975)*z_21[1] + (51.44887*20+601.1975)*z_21[2] + (43.9046*35+752.083)*z_21[3] + (28.9018*45+1277.181)*z_21[4]
+ (4.524286*52+2374.169)*z_21[5];

```

FIGURE 5.5: Implementation of the alternative objective function as a piecewise linear approximation of the nonlinear objective function (4.12).

5.3.1 Motivation for the selected approximate solution methodology

Existing models of the GMS problem in the literature are known for their high computational complexity due to the large dimensions typically associated with real-world instances of the problem. Exact solution approaches, such as dynamic programming techniques, exhibit a serious shortcoming in terms of being able to find model solutions within acceptable timeframes to such realistically sized problem instances [187, 219]. When solving large-scale model instances, the number of solutions grows rapidly as a function of the number of PGUs [187] (*i.e.* all $\binom{p}{u}$ candidate solutions have to be considered implicitly in order to find a solution). For typical parameter values of $p = 365$ and u of the order 50, this complexity renders exact solution of the GMS problem out of reach of mixed integer programming methods implemented in currently available software and hardware systems. Hence there is generally a need to pursue approximate GMS model solution techniques instead of exact methods.

In population-based approximate model solution approaches such as a GA, however, recombination operators often cause infeasibility of solutions. This disadvantage is eliminated by the trajectory-based SA techniques toward solving instances of the GMS problem. Employing other population-based approaches, such as PSO, often also involve long computation times and in some instances it is difficult to find sufficiently diverse starting solutions and search parameter values [140]. The method of ACO exhibits other drawbacks such as that the algorithmic implementation is somewhat more cumbersome than that of other metaheuristics (deciding on a suitable updating scheme for pheromone trails may, for example, prove difficult) [191].

The comparatively simpler method of SA is adopted in this dissertation for solving the GMS models of §4.3 due to its previously documented successes in the GMS domain [37, 187, 189]. This method has proven capable of providing high-quality solutions in acceptable time frames within the context of GMS. In some cases it has even outperformed certain genetic algorithmic implementations and has matched the best solutions found by ACO [81, 170, 189]. Furthermore, the method of SA is more flexible than other trajectory-based approximate solution approaches such as tabu search or local (improving) searches. It involves a comparatively simpler method of escaping from local optima than having to implement lists of tabu moves. It is finally also easy to implement the SA algorithm and to adapt it to a multi-objective solution approach if need be.

5.3.2 Simulated annealing implementation

This section contains a description of the particular implementation of the SA algorithm employed in this dissertation to obtain high-quality solutions to instances of the GMS models of §4.3. The section includes a discussion on the method of determination of initial solutions and the choice of an initial temperature for the algorithm, the cooling and reheating schedules employed, the constraint handling technique adopted, the epoch management protocol implemented, the neighbourhood move operator used and the termination criteria selected.

The initial solution

An initial solution is required in order to initialise the SA algorithm. The implementation of the SA algorithm employed in this dissertation randomly generates an initial solution. This is achieved by generating a random feasible maintenance commencement date, based on a uniform distribution, for each PGU within the maintenance window of that PGU.

The choice of initial temperature

In 1987, van Laarhoven *et al.* [211] presented a method for calculating an appropriate value for the initial temperature T_0 , called the average increase method. The value of the initial temperature may be determined from the so-called initial acceptance ratio

$$\chi_0 = \exp\left(-\frac{\overline{\Delta E}}{T_0}\right), \quad (5.22)$$

where $\overline{\Delta E}$ is the average deterioration in objective function values (*i.e.* change in energy) of the system and where χ_0 is the acceptance ratio during a pre-solution random walk through the solution space, starting from the randomly generated initial solution. This yields the initial temperature as

$$T_0 = -\frac{\overline{\Delta E}}{\ln(\chi_0)}. \quad (5.23)$$

The initial acceptance ratio χ_0 in (5.23) is the number of accepted solutions which exhibit a deterioration in objective function value divided by the number of attempted solutions which exhibit a deterioration in the objective function value during the pre-solution random walk. This ratio is typically set to a value between 0.5 and 0.8, as suggested by Buseti [38]. The length (number of iterations) of the pre-solution random walk is selected as 1 000 in this dissertation, using the initial solution as the starting point of the random walk. A pseudo-code description of the procedure for determining the initial temperature is given in Algorithm 5.1.

Algorithm 5.1: Determining initial temperature

Input : The initial solution matrix, the initial objective function value, the length of the pre-solution random walk, the acceptance percentage and a full data set specifying an instance of one of the models of §4.3.

Output: The initial temperature calculated using the average increase method.

```

1 counter ← 0;
2 CurrentObjValue ← InitialObjValue;
3 for i ← to WalkLength do
4   PreviousObjValue ← CurrentObjValue;
5    $\mathbf{x}$  ← EjectionChain ( $\mathbf{x}$ );
6   CurrentObjValue ← ObjValue ( $\mathbf{x}$ );
7   ChangelnEnergy ← PreviousObjValue – CurrentObjValue;
8   if ChangelnEnergy > 0 then
9     counter ← counter +1;
10    Increases [counter ] ← ChangelnEnergy;
11    ObjValueArray [counter ] ← CurrentObjValue;
12 InitialTemperature ← –mean(Increases)/log(AcceptPercentage);
13 return InitialTemperature
```

The method of constraint handling

The constraints included in the GMS models of §4.3 are either implemented as hard or soft constraints in the SA algorithm. More specifically, the maintenance window constraints and the maintenance duration constraints are implemented as hard constraints. Therefore, solutions

which violate any of these constraints are considered to be infeasible, and are therefore not acceptable during the search. The remaining constraints (the demand constraints, the resource availability constraints, the exclusion constraints and the precedence constraints) are all implemented as soft constraints. A solution which violates one of these constraints is considered to be feasible, but the objective function value of such a solution is penalised by a certain value so as to allow the SA search to pass through infeasible areas of the solution space, but discouraging it from remaining there.

A multiplicative penalty function approach is adopted in this dissertation to accommodate soft constraints of the GMS models. This approach involves penalisation of the objective function for a violation of a soft constraint by the magnitude of the violation *via* multiplication [190]. Consider, for example, a soft constraint set of the form

$$g_i(\mathbf{x}) \leq G_i, \quad i = 1, \dots, s,$$

where G_i is a constant limiting value, which is strictly positive, for the i -th soft constraint and s is the number of soft constraints [190]. A total scaled constraint violation value

$$\mathcal{G}(\mathbf{x}) = \sum_{i=1}^s \max \left\{ 0, \frac{g_i(\mathbf{x}) - G_i}{G_i} \right\}$$

may be calculated for the set of constraints. The scheduling criterion which seeks to minimise the probability of unit failure, presented in §4.3.2, involves minimisation. The objective function in (4.6) is furthermore always positive. Therefore, the multiplicative penalty function for a minimisation problem

$$\phi^-(\mathbf{x}) = \exp[\gamma(\mathcal{G}(\mathbf{x}))]$$

is adopted in the case of the linear GMS model of §4.3, where γ is a severity factor which is typically determined empirically. The penalised objective function for the minimisation problem may then be calculated as

$$Z(\mathbf{X}) \times \phi^-(\mathbf{x}), \quad (5.24)$$

where $Z(\mathbf{X})$ is as defined in (4.6). The product in (5.24) is minimised in an attempt to find the best feasible solution during the SA search.

The alternative scheduling criterion which seeks to maximise the expected energy production, also presented in §4.3.2, involves maximisation. Moreover, the objective function in (4.12) is also always positive. Therefore, the multiplicative penalty function for a maximisation problem

$$\phi^+(\mathbf{x}) = 2 - \exp[\gamma(\mathcal{G}(\mathbf{x}))]$$

is adopted in the case of the nonlinear GMS model of §4.3. The penalised objective function for the maximisation problem may then be calculated as

$$Z'(\mathbf{X}) \times \phi^+(\mathbf{x}), \quad (5.25)$$

where $Z'(\mathbf{X})$ is as defined in (4.12). The product in (5.25) is maximised in an attempt to find the best feasible solution during the SA search.

The epoch management protocol

Busetti [38] argued that the number of iterations spent in a single SA epoch k should be problem-dependent, rather than being a function of k . A Markov chain³ of length L_k may of course be

³A sequence of possible events described by a stochastic model in which the probability of an event occurring only depends on the state attained during the previous event [218].

used to determine the number of iterations during epoch k . Ideally, however, one would rather prefer to specify a minimum number A_{min} of move acceptances during any epoch before lowering the temperature and moving on to the next epoch. A problem with this approach is that as the temperature T_k approaches zero, the probability of accepting non-improving solutions decreases which, in turn, increases the expected number of iterations required to achieve the required A_{min} move acceptances per epoch. Therefore, an epoch is terminated once a certain number L of moves have been attempted, by reheating the temperature of the system to the initial temperature or according to some reheating schedule, or alternatively by decreasing the temperature according to a cooling schedule once A_{min} moves have been accepted, where A_{min} and L are pre-specified parameters satisfying $A_{min} < L$. Dreo *et al.* [62] proposed that $A_{min} = 12N$ and $L = 100N$, where N is a measure of the number of degrees of freedom in the problem instance, taken here as the number of PGUs in the system (*i.e.* $N = n$).

In order to determine the most effective length of an epoch for a given GMS problem instance, an epoch parameter is additionally introduced here. The epoch parameter is incorporated into the above epoch termination criteria by instead taking $A_{min} = 12n/\psi$ and $L = 100L/\psi$ in this dissertation. The epoch parameter is determined by an extensive parameter evaluation for each instance of the GMS problem. The ratio of A_{min} to L is, however, taken as proposed by Dreo *et al.* [62].

The cooling and reheating schedules

The geometric cooling schedule in (2.48) is implemented within the SA algorithmic implementation in this dissertation. This cooling schedule has proved to be very successful in solving various GMS problem instances in the literature [2, 127, 188, 189]. The geometric cooling schedule is also very easy to implement and converges to good solutions within a reasonable amount of computation time. The value of the cooling parameter α which appears in this schedule is determined during an extensive parameter evaluation for each of the problem instances considered later in this dissertation.

The reheating schedule adopted in the SA algorithmic implementation of this dissertation is based on a similar approach to that followed in the geometric schedule. Once L moves have been attempted during a single epoch without reaching the required A_{min} accepted moves, the temperature of the system is reheated. In such a case, the temperature is reheated according to the schedule

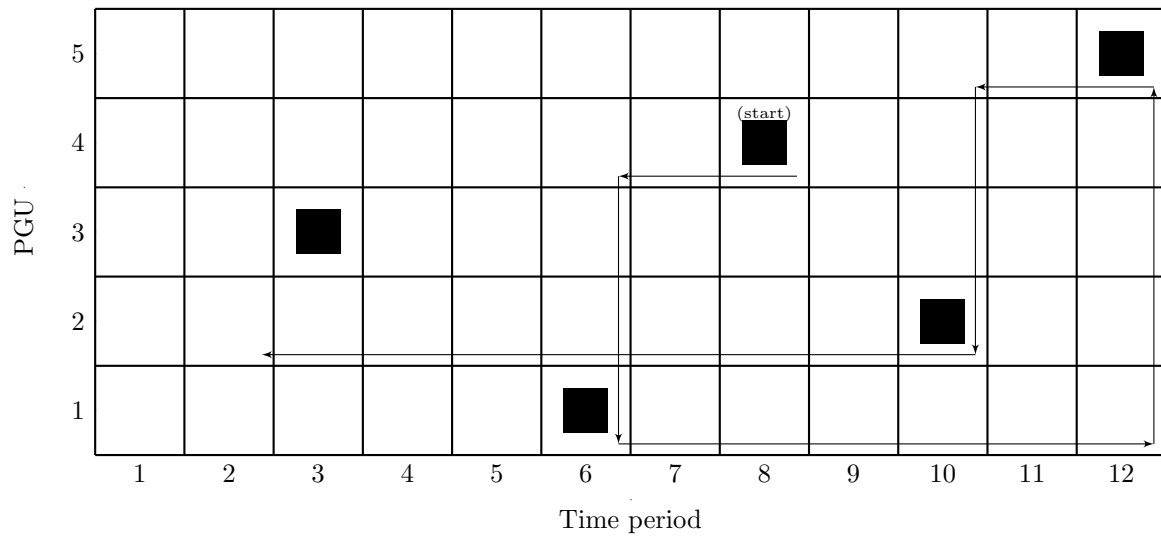
$$T_{k+1} = \frac{T_k}{\xi}, \quad k = 0, 1, 2, \dots, \quad (5.26)$$

where ξ is the *reheating parameter*, which is also determined by an extensive parameter evaluation for each instance of the GMS problem considered.

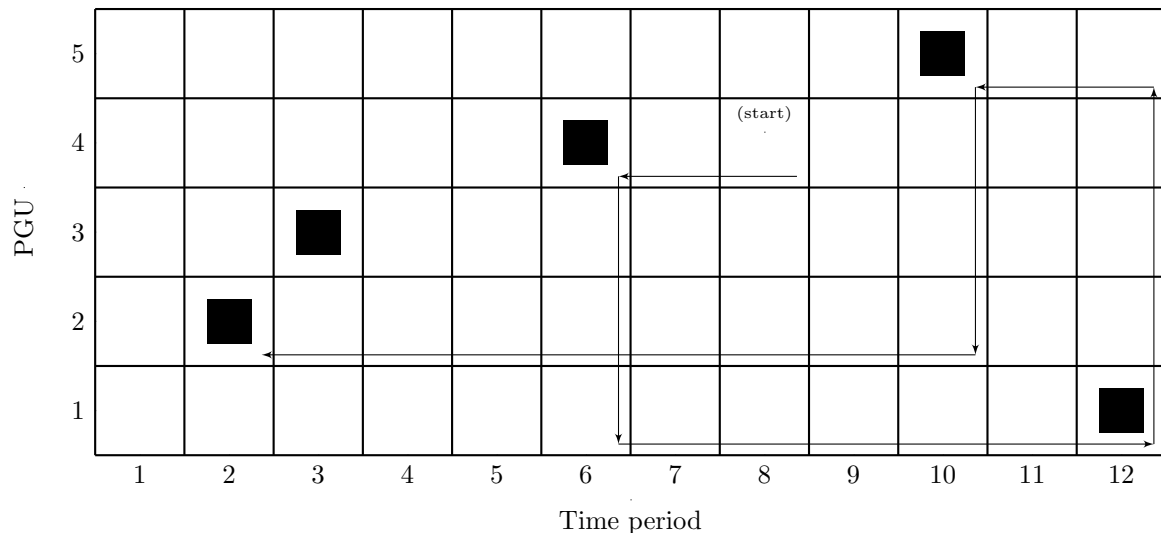
The neighbourhood move operator

The neighbourhood move operator implemented within the SA algorithmic implementation of this dissertation is the well-known *ejection chain* move operator [188, 189], which involves generating a list of PGUs whose maintenance period commencement times are altered randomly in succession within their allowable maintenance windows. More specifically, a PGU is selected randomly after which its maintenance commencement time is moved randomly without violating the maintenance window constraint of that PGU. Another PGU, whose planned maintenance is scheduled to start during the newly allocated maintenance time of the initially selected PGU is next selected randomly and its maintenance commencement time is moved to a random time

within the maintenance window of that PGU. This process is repeated until either of the following stopping criteria is met: no maintenance of a PGU is scheduled to start during the newly allocated maintenance time of the latest selected PGU in the sequence or chain, or a PGU in the chain has a maintenance starting time that corresponds to the initial starting time of the PGU that was initially selected to initiate the ejection chain. A graphical representation of the ejection chain neighbourhood move operator is shown in Figure 5.6, with Figure 5.6(a) containing an indication of the state of the schedule before the ejection chain is applied and Figure 5.6(b) containing an indication of the state of the schedule after the ejection chain has been applied. This neighbourhood move operator is affected by the implementation of Algorithm 5.2.



(a) Maintenance schedule before applying the ejection chain move operator



(b) Maintenance schedule after having applied the ejection chain move operator

FIGURE 5.6: Graphical representation of a small, hypothetical GMS before and after applying the ejection chain move operator to the corresponding decision matrix.

In other implementations of the method of SA within the context of GMS, a more basic move operator, often referred to as the *classical* move operator, has been implemented, which involves merely selecting a PGU randomly (according to a uniform distribution) and randomly moving the maintenance commencement time of the selected PGU to a new point in time within the

Algorithm 5.2: Ejection chain move operator

Input : The current solution matrix, the number of PGUs in the system, as well as earliest and latest starting times for the maintenance of each PGU.

Output: A new solution vector after having applied the ejection chain neighbourhood move operator to the current solution matrix.

```

1 ListOfMoves  $\leftarrow$   $\emptyset$ ;
2 MovingUnit  $\leftarrow$  rand(NumberOfUnits);
3 ListOfMoves [1]  $\leftarrow$  MovingUnit;
4 while  $i \leftarrow 1$  to (NumberOfUnits- 1) do
5     TempUnitList  $\leftarrow$   $\emptyset$ ;
6     SolutionVector [MovingUnit ]  $\leftarrow$  randBetween(EarliestStartTime, LatestStartTime);
7     TempNewTime  $\leftarrow$  SolutionVector [MovingUnit ];
8     for counter  $\leftarrow 1$  to NumberOfUnits do
9         if TempNewTime == SolutionVector [i] then
10             TempUnitList [counter ]  $\leftarrow i$ ;
11             counter  $\leftarrow$  counter + 1;
12     TempUnitList  $\leftarrow$  TempUnitList [!TempUnitList %in% MovingUnit ];
13     Len  $\leftarrow$  length(TempUnitList);
14     if Len == 0 then
15         Break
16     MovingUnit  $\leftarrow$  TempUnitList [randBetween(1,counter- 1)];
17      $i \leftarrow i + 1$ ;
18     ListOfMoves [i]  $\leftarrow$  MovingUnit;
19 return SolutionVector

```

maintenance window of that PGU [188, 189]. Due to the localised nature of the classical move operator, its use may pose difficulties for the algorithm in terms of escaping from local optima. Comparisons of the classical move operator with the ejection chain operator described above have shown that the ejection chain operator outperforms the classical move operator in most GMS problem instances [188] and therefore the ejection chain move operator is the only move operator implemented within the SA implementation of this dissertation.

End-of-search termination criteria

The implementation of the SA algorithm adopted here is subject to two termination criteria. The primary termination criterion corresponds to when the system is said to be in a frozen state. Once three consecutive reheats have occurred, the system is considered frozen, as A_{min} accepted solution could not be found for three consecutive epochs during the search. A secondary termination criterion is implemented only to ensure that the algorithm does not exceed a computation time of eight hours. The reason for choosing an 8-hour timeout period is that the calculations may thus be completed overnight. This timeout period was verified as sensible by Mr Bhongo Mdungo [157], a production assurance manager of the national power utility of South Africa, Eskom, through personal communication.

Once the SA algorithm has terminated, the best solution encountered during the entire search is returned for the given instance of one of the GMS models of §4.3, along with a graphical representation of the corresponding maintenance schedule.

5.3.3 Experimental design

In order to obtain the best results by means of the method of SA, a suitable combination of algorithmic parameter values has to be selected. This combination will ensure that the solution space is explored thoroughly and that the SA algorithm returns, on average, solutions of high quality for a selected instance of the GMS models of §4.3. The method of SA employed to solve instances of the GMS models of §4.3, as described in §5.3.3, requires specification of five parameter values. These parameters are the cooling parameter α , the reheating parameter ξ , the epoch parameter ψ for determining the length of an epoch, the acceptance ratio at the initial temperature χ_0 , and the severity factor γ by which the objective function is penalised for solutions which violate the soft constraints of the models of §4.3.

In order to conduct an SA parameter optimisation experiment, two sets of parameters are considered separately, in two different phases. During the first phase, the parameters to be evaluated are the acceptance ratio and the soft constraint violation severity factor. During the second phase of the experiment, the best combination of parameter values obtained during the first phase is adopted as constant values. The set of parameters to be considered (varied) during the second phase of the experiment are the cooling parameter, the reheating parameter and the epoch parameter.

TABLE 5.2: *Parameter values considered for the initial acceptance ratio and the soft constraint violation severity factor during the first phase of the parameter optimisation experiment.*

Parameter	Symbol	Very low	Low	Medium	High	Very high
Acceptance ratio	χ_0	0.4	0.5	0.6	0.7	0.8
Severity factor	γ	0.25	0.50	0.75	1.00	1.25

For each of the parameters varied during the first phase of the experiment, five values are considered. These five values correspond to very low, low, medium, high and very high parameter values for the initial acceptance ratio and the severity factor, respectively. The parameter values to be considered are shown in Table 5.2. In order to evaluate all the combinations of these parameters, $5^2 = 25$ parameter configurations have to be considered. For each of these parameter configurations, each model of §4.3 is solved 30 times by the method of SA in order to obtain the mean objective function value and the mean computation time per run. Each run is limited to either four or eight hours of computation time, depending on the size of the test instance considered. The values of the other three parameters which are not varied during the first phase, are kept constant at values which correspond to medium values. The medium values of these parameters may be found in the penultimate column of Table 5.3.

TABLE 5.3: *Parameter values considered for the cooling parameter, the reheating parameter and the epoch parameter during the second phase of the parameter optimisation experiment.*

Parameter	Symbol	Low	Medium	High
Cooling parameter	α	0.85	0.90	0.95
Reheating parameter	ξ	0.55	0.75	0.95
Epoch parameter	ψ	1	2	4

The way in which good parameter configurations are obtained during the second phase of the parameter optimisation experiment involves considering three values of each remaining parameter. These three values correspond to low, medium and high parameter configurations. Each of the resulting $3^3 = 27$ combinations of the parameters have to be evaluated in order to ensure that suitable parameter values are employed when eventually solving the GMS models of §4.3 in

Algorithm 5.3: Simulated annealing for GMS

Input : A full data set for an instance of the GMS models of §4 as well as all the parameters required for implementation of the SA algorithm.

Output: The best solution matrix (and corresponding objective function value) found during the SA search.

```

1 EpochNum  $\leftarrow$  0;
2 counter  $\leftarrow$  1;
3 Stuck  $\leftarrow$  0;
4 BestSol  $\leftarrow$  0;
5 Solutions  $\leftarrow$   $\emptyset$ ;
6 Solutions [counter]  $\leftarrow$  ObjValue ( $\mathbf{x}$ );
7 while (EpochNum  $\leq$  3) & (counter  $\leq$  NumOfIterations) do
8   counter  $\leftarrow$  counter +1;
9    $\mathbf{x} \leftarrow$  EjectionChain ( $\mathbf{x}$ );
10  CurrentObjValue  $\leftarrow$  ObjValue ( $\mathbf{x}$ );
11  if NumberAccepts  $\geq A_{min}$  then
12    Temp [counter]  $\leftarrow$   $\alpha$ Temp [counter -1];
13    Stuck  $\leftarrow$  0;
14    EpochNum  $\leftarrow$  0;
15  else
16    Temp [counter]  $\leftarrow$  Temp [counter -1];
17    Stuck  $\leftarrow$  Stuck +1;
18  if Stuck == L then
19    EpochNum  $\leftarrow$  EpochNum +1;
20    Stuck  $\leftarrow$  0;
21    Temp [counter]  $\leftarrow$  Temp[counter+1]/ $\xi$ ;
22  if Solutions [counter -1] > CurrentObjValue then
23    Solutions [counter]  $\leftarrow$  CurrentObjValue;
24    if Solutions [counter] < BestSol then
25      BestSol  $\leftarrow$  Solutions [counter];
26      BestSol_X  $\leftarrow$   $\mathbf{x}$ 
27  else if (exp((Solutions [counter -1]-CurrentObjValue )/Temp[counter]))  $\geq$ 
    RandBetween(0, 1) then
28    Solutions [counter]  $\leftarrow$  CurrentObjValue;
29  else
30    Solutions [counter]  $\leftarrow$  Solutions [counter -1];
31 return BestSol_X & BestSol

```

the context of a test instance. The parameter values considered for each of the three mentioned parameters during the second phase of the parameter optimisation experiment are shown in Table 5.3. For each of these parameter configurations, the GMS models of §4.3 are again solved 30 times by the method of SA in order to obtain the mean objective function value and the mean computation time per run. Each run is again limited to four or eight hours of computation time. In each case, the best parameter configuration obtained during the first phase of the parameter optimisation experiment is taken as constant values for all combinations of the parameters during the second phase.

5.4 Chapter summary

This chapter contained a description of the approaches adopted in this dissertation to obtain exact or approximate solutions to instances of the GMS models of §4.3.

The exact solution approach involves exploitation of the mathematical programming nature of the GMS problem and is implemented in CPLEX, which is capable of providing exact solutions to relatively small MIP instances with linear objective functions within a reasonable amount of time. The description of the exact solution methodology included a motivation for the selection of CPLEX as the exact solution platform and an overview of the implementation of the GMS models of §4.3 in CPLEX in §5.2.

In §5.3, a second, albeit approximate, solution approach was described, namely the metaheuristic method of SA. This description included an explanation of the reasoning behind selecting SA as approximate solution technique and the implementation of SA in the context of the GMS models of §4.3 in R. A detailed discussion was also included on the method of determining initial solutions and the initial temperature for the algorithm, the cooling and reheating schedule employed, the constraint handling technique implemented, the epoch management protocol adopted, the neighbourhood move operator selected and the termination criteria enforced. A pseudo-code summary of the SA algorithmic implementation employed in this dissertation is given in Algorithm 5.3. In §5.3.3, the experimental design that is to be followed to determine a suitable combination of parameter values for the SA algorithm was described. The experiment will be performed in two separate phases. The first phase involves variation of the initial acceptance ratio and the soft constraint violation severity factor, whereas the second phase involves variation of the cooling parameter, the reheating parameter and the epoch parameter.

Part III

Academic benchmark results

CHAPTER 6

Academic benchmark system data

Contents

6.1	The 21-unit system	101
6.2	The 32-unit IEEE-RTS	103
6.3	Chapter summary	109

The solution approaches described in §5 are applied later in this dissertation to solve the newly proposed GMS model of §4 in the context of two test systems from the GMS literature. The first system is the so-called *21-unit system* which was introduced by Dahal and McDonald [54]. The second is a system introduced by Schlünz [188] in 2011 which is based on the *Institute of Electrical Engineering and Electronic Engineering Reliability Test System* (IEEE-RTS) data set [8, 9]. The data and model constraints specified for each of these benchmark systems are described in detail in this chapter.

6.1 The 21-unit system

The 21-unit test system was derived by Dahal and McDonald [54] from a system presented by Yamayee [219] in 1983, but with additional constraints and some simplifications so as to represent an adequate GMS problem instance. The original objective in this GMS problem instance was to minimise the SSR levels (*i.e.* a reliability criterion). The constraints include maintenance window constraints for each PGU in the system, the availability of maintenance personnel during each week of the planning period, requirements on personnel during each time period of maintenance for every PGU and meeting the system load demand.

This test system contains twenty one PGUs. The specifications of these PGUs are presented in Table 6.1. The scheduling window spans fifty two weeks with a constant peak demand of 4739 MW for each week over the entire scheduling window. Each of the PGUs has an allowable maintenance window which is also presented in Table 6.1 along with the duration of the maintenance that has to be performed on the PGU. The maximum number of maintenance personnel available to perform maintenance during any week of the planning period is twenty. The assumed number of time periods that have elapsed between the end of the previous maintenance date of each PGU in the 21-unit test system and the start of the current planning horizon, is based on PGU maintenance timings in the best solution obtained by Schlünz [188] for this benchmark system and these time durations are presented in Table 6.2.

TABLE 6.1: Specifications and maintenance requirements for the 21-unit test system [54].

PGU	Capacity (MW)	Earliest starting time (week)	Latest starting time (week)	Duration (weeks)	Manpower required during each week of maintenance	Failure rate
1	555	1	20	7	10, 10, 5, 5, 5, 5, 3	0.0873
2	555	27	48	5	10, 10, 10, 5, 5	0.0873
3	180	1	25	2	15, 15	0.0873
4	180	1	26	1	20	0.0873
5	640	27	48	5	10, 10, 10, 10, 10	0.0873
6	640	1	24	3	15, 15, 15	0.0873
7	640	1	24	3	15, 15, 15	0.0873
8	555	27	47	6	10, 10, 10, 5, 5, 5	0.1178
9	276	1	17	10	3, 2, 2, 2, 2, 2, 2, 2, 3	0.1178
10	140	1	23	4	10, 10, 5, 5	0.1178
11	90	1	26	1	20	0.1527
12	76	27	50	3	10, 15, 15	0.1527
13	76	1	25	2	15, 15	0.0873
14	94	1	23	4	10, 10, 10, 10	0.0873
15	39	1	25	2	15, 15	0.1527
16	188	1	25	2	15, 15	0.1527
17	58	27	52	1	20	0.0873
18	48	27	51	2	15, 15	0.0873
19	137	27	52	1	15	0.3733
20	469	27	49	4	10, 10, 10, 10	0.0873
21	52	1	24	3	10, 10, 10	0.0873

In an ideal case, if failure times were available for each of the PGUs, which is not the case for the 21-unit test system, the failure times of each of the PGUs in the system could have been analysed as described in §3. Such data would first have been classified as either complete data, right-censored data, interval-censored data or left-censored data, as demonstrated in §3.5. Thereafter, one of the trend tests described in §3.6 would have been employed to determine whether or not a trend exists in the failure times, indicating whether the system is repairable or non-repairable. An appropriate life distribution model (in the case of a non-repairable system), as described in §3.3, or an appropriate repair model (in the case of a repairable system), as described in §3.4, would then have been selected for each of the PGUs in the system. Finally, the parameters (*i.e.* failure rates) of the selected model would have been calculated by employing one of the model parameter estimation methods described in §3.7.

Since failure times for the PGUs are not available, however, the failure rate of each of the PGUs was instead extrapolated from the RTS-79 system (described in more detail in the following section) by comparing the size and duration of each PGU in the two test systems. These approximate failure rates are also presented in Table 6.1.

In order to solve the nonlinear model described in §4.3, the objective function was linearised

TABLE 6.2: The assumed number of weeks that have elapsed between the previous PGU maintenance times and the start of the current planning horizon for the 21-unit test system.

PGU	1	2	3	4	5	6	7	8	9	10	11
x'_u	46	15	32	50	5	49	28	22	39	38	51
PGU	12	13	14	15	16	17	18	19	20	21	
x'_u	25	40	36	43	30	0	17	10	9	46	

by means of the piecewise linear approximation method described in §5.1. The breakpoints for each of the curves which represents the expected energy produced by a PGU in the 21-unit test system over the scheduling window were calculated and are presented in Table 6.3. The number of breakpoints in each approximation was selected as the smallest value for which the corresponding piecewise linear function accounts for more than 99.5% of the true expected energy production curve (in terms of the maximum deviation of the expected energy curve from the piecewise linear approximation function). The error proportion of the piecewise linear approximation function and the corresponding number of breakpoints for each PGU are also presented in Table 6.3. The true expected energy production curve of PGU 17, for example, exhibiting the largest number of interior breakpoints, and that of PGU 19, exhibiting the smallest number of interior breakpoints, are shown in Figure 6.1. The piecewise linear approximation of each of these curves is also shown in the figure together with the corresponding number of breakpoints.

TABLE 6.3: *Optimal piecewise linear approximation breakpoints for the 21-unit test system which result in an expected energy production approximation remaining within an error band of 0.5% of the true expected energy production curve for each PGU.*

PGU	Number of interior breakpoints	Error proportion	Interior breakpoints
1	3	0.00392	21, 35, 45
2	4	0.00423	8, 19, 33, 44
3	3	0.00346	15, 32, 44
4	2	0.00409	27, 42
5	5	0.00393	6, 13, 22, 33, 44
6	2	0.00484	27, 42
7	3	0.00406	14, 31, 43
8	3	0.00487	16, 35, 45
9	4	0.00348	22, 34, 42, 48
10	2	0.00275	31, 44
11	2	0.00223	34, 45
12	2	0.00425	33, 45
13	3	0.00296	17, 33, 44
14	3	0.00360	17, 33, 44
15	2	0.00266	34, 45
16	2	0.00315	33, 45
17	5	0.00401	5, 11, 19, 30, 42
18	4	0.00344	8, 19, 33, 44
19	1	0.00209	46
20	4	0.00487	7, 17, 30, 43
21	3	0.00285	20, 35, 45

6.2 The 32-unit IEEE-RTS

The IEEE-RTS GMS problem instance was published in 1979 [8], and is referred to in this dissertation as the RTS-79 system. This test system was developed by a subcommittee of the IEEE Power System Engineering Committee and its Application of Probability Methods Subcommittee. Before the publication of this test system, a need existed for a standardised power system for the purpose of reliability evaluation which eventually led to the development

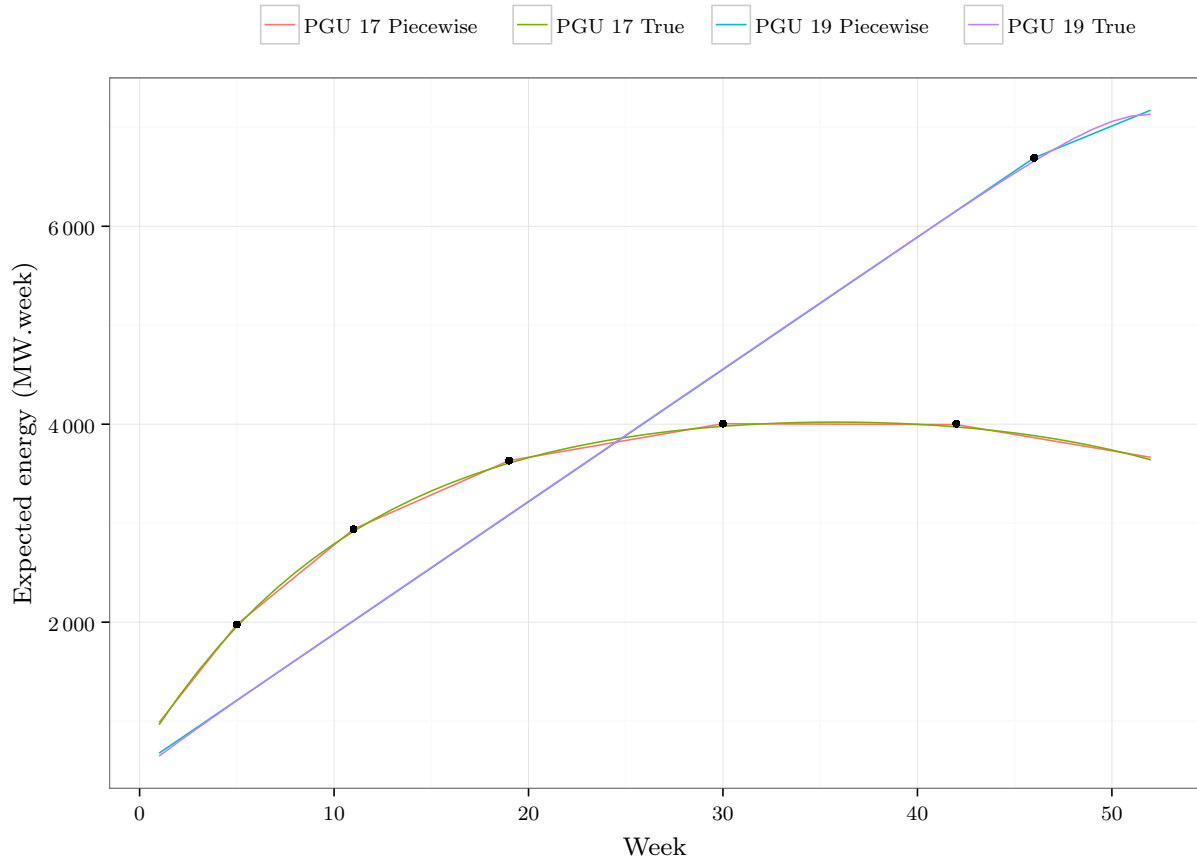


FIGURE 6.1: The true curves representing the expected energy production of two PGUs in the 21-unit test system together with the corresponding piecewise linear approximation of each curve. The curves represent the expected energy of the PGU with the largest number of breakpoints in Table 6.3 (PGU 17) as well as that of the PGU with the smallest number of breakpoints (PGU 19). The black dots indicate the corresponding breakpoints for each of the piecewise linear approximations.

of the RTS-79 system. Since then, revisions to the test system have been published (in 1986 and 1999). These revisions are referred to in this dissertation as the RTS-86 system [9] and the RTS-96 system [99], respectively.

A more recent test system, which was derived from the RTS-79 system as a result of the inclusion of a number of additional parameter values and constraints, was published by Schlünz [188] in 2011. This system is henceforth referred to as the IEEE-RTS and comprises maintenance window constraints, load demand constraints (which include a safety margin), resource requirements and availability constraints, as well as PGU maintenance exclusion constraints.

The system contains thirty two PGUs that each has to be scheduled once for planned maintenance within a scheduling window of fifty two weeks. The specifications of the IEEE-RTS, including the capacities of PGUs, earliest and latest starting times for maintenance of each PGU, PGU maintenance durations and manpower requirements associated with the maintenance of each PGU, are presented in Table 6.4. The manpower available is limited to twenty five during each week of the scheduling window.

The PGUs that form part of each of seven exclusion sets are presented in Table 6.5. The PGUs contained in these exclusion sets are PGUs within single power stations and therefore

TABLE 6.4: Specifications and maintenance requirements for the IEEE-RTS [188].

PGU	Capacity (MW)	Earliest starting time (week)	Latest starting time (week)	Duration (weeks)	Manpower required during each week of maintenance	Failure rate
1	20	1	25	2	7, 7	0.3733
2	20	1	25	2	7, 7	0.3733
3	76	1	24	3	12, 10, 10	0.0857
4	76	27	50	3	12, 10, 10	0.0857
5	20	1	25	2	7, 7	0.3733
6	20	27	51	2	7, 7	0.3733
7	76	1	24	3	12, 10, 10	0.0857
8	76	27	50	3	12, 10, 10	0.0857
9	100	1	50	3	10, 10, 15	0.1400
10	100	1	50	3	10, 10, 15	0.1400
11	100	1	50	3	15, 10, 10	0.1400
12	197	1	23	4	8, 10, 10, 8	0.1768
13	197	1	23	4	8, 10, 10, 8	0.1768
14	197	27	49	4	8, 10, 10, 8	0.1768
15	12	1	51	2	4, 4	0.0571
16	12	1	51	2	4, 4	0.0571
17	12	1	51	2	4, 4	0.0571
18	12	1	51	2	4, 4	0.0571
19	12	1	51	2	4, 4	0.0571
20	155	1	23	4	5, 15, 10, 10	0.1750
21	155	27	49	4	5, 15, 10, 10	0.1750
22	400	1	21	6	15, 10, 10, 10, 10, 5	0.1527
23	400	27	47	6	15, 10, 10, 10, 10, 5	0.1527
24	50	1	51	2	6, 6	0.0857
25	50	1	51	2	6, 6	0.0857
26	50	1	51	2	6, 6	0.0857
27	50	1	51	2	6, 6	0.0857
28	50	1	51	2	6, 6	0.0857
29	50	1	51	2	6, 6	0.0857
30	155	1	23	4	12, 12, 8, 8	0.1750
31	155	1	49	4	12, 12, 8, 8	0.1750
32	350	1	48	5	5, 10, 15, 15, 5	0.1461

TABLE 6.5: Seven exclusion data sets for the IEEE-RTS [188].

Exclusion set	Maximum	Units
1	2	1, 2, 3, 4
2	2	5, 6, 7, 8
3	1	9, 10, 11
4	1	12, 13, 14
5	3	15, 16, 17, 18, 19, 20
6	3	24, 25, 26, 27, 28, 29
7	1	30, 31, 32

only a limited number of PGUs are allowed to be in simultaneous maintenance in each of the seven cases.

TABLE 6.6: Demand requirement per week for the IEEE-RTS [188].

Week	Demand (MW)	Week	Demand (MW)	Week	Demand (MW)	Week	Demand (MW)
1	2 457	14	2 138	27	2 152	40	2 063
2	2 565	15	2 055	28	2 326	41	2 118
3	2 502	16	2 280	29	2 283	42	2 120
4	2 377	17	2 149	30	2 508	43	2 280
5	2 508	18	2 385	31	2 058	44	2 511
6	2 397	19	2 480	32	2 212	45	2 522
7	2 371	20	2 508	33	2 280	46	2 591
8	2 297	21	2 440	34	2 078	47	2 679
9	2 109	22	2 311	35	2 069	48	2 537
10	2 100	23	2 565	36	2 009	49	2 685
11	2 038	24	2 528	37	2 223	50	2 765
12	2 072	25	2 554	38	1 981	51	2 850
13	2 006	26	2 454	39	2 063	52	2 713

The load demand of the test system, presented in Table 6.6, exhibits a typical two-seasonal peak demand characteristic with the peak load being reached during Week 51. The load demand requires a safety margin of 15%, as indicated in Figure 6.2, which also contains the load demand for all fifty two weeks (including the safety margin). Finally, the presumed number of time periods that have elapsed between the ends of the previous PGU maintenance instances and the start of the current planning horizon in the IEEE-RTS are again based on the PGU maintenance timings in the best solution obtained by Schlünz [188] for this test system and these time durations are presented in Table 6.7.

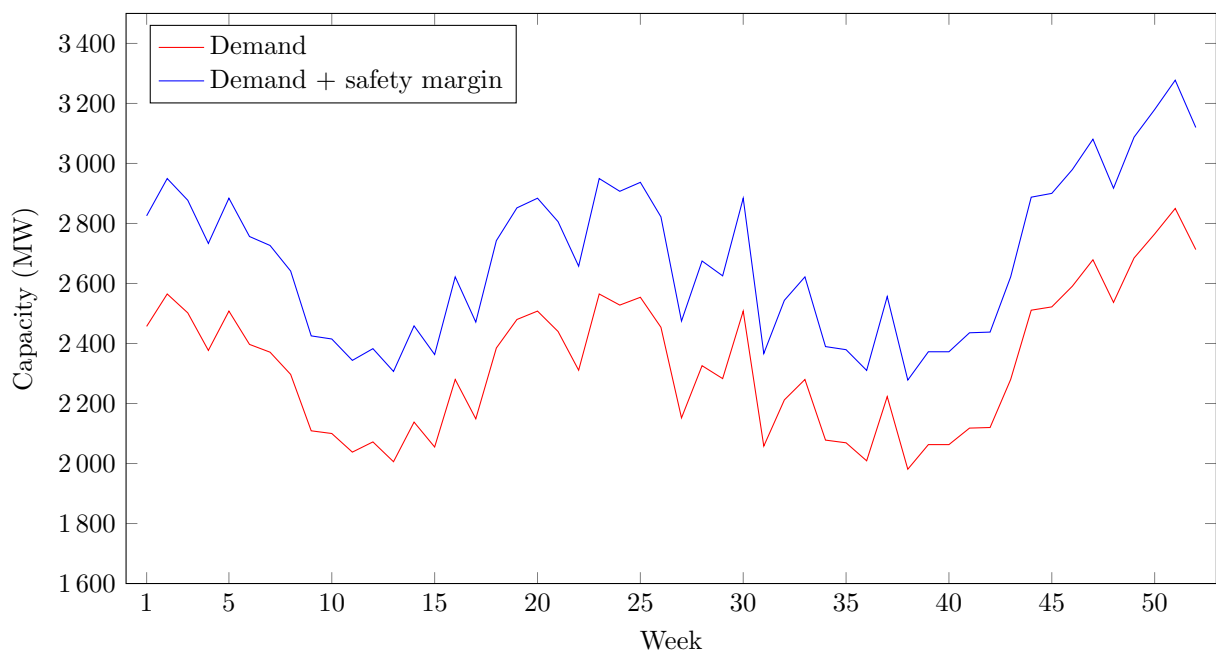


FIGURE 6.2: IEEE-RTS demand including safety margin of 15% [188].

TABLE 6.7: The assumed number of time periods that have elapsed between the previous PGU maintenance times and the start of the current planning horizon for the IEEE-RTS.

PGU	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
x'_u	46	27	51	8	49	21	30	16	11	25	43	38	48	15	9	26
PGU	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
x'_u	18	18	27	37	25	44	21	18	10	26	35	29	16	40	33	14

TABLE 6.8: Optimal piecewise linear approximation breakpoints for the IEEE-RTS which result in an expected energy production approximation remaining within an error band of 0.5% of the true expected energy production curve for each PGU.

PGU	Number of interior breakpoints	Error proportion	Interior breakpoints
1	1	0.2631	47
2	1	0.2632	47
3	2	0.48	27, 42
4	4	0.48	7, 16, 29, 42
5	1	0.2631	47
6	1	0.2647	47
7	3	0.3886	14, 31, 43
8	4	0.3731	8, 19, 33, 44
9	4	0.4005	6, 16, 34, 45
10	3	0.3268	16, 36, 46
11	2	0.3471	33, 45
12	2	0.3409	36, 46
13	2	0.3317	36, 46
14	3	0.4408	10, 36, 46
15	4	0.3985	8, 18, 29, 41
16	3	0.4357	12, 27, 41
17	4	0.3293	9, 19, 31, 42
18	4	0.3293	9, 19, 31, 42
19	3	0.4271	12, 27, 41
20	2	0.348	36, 46
21	2	0.4553	35, 46
22	2	0.4997	34, 45
23	3	0.4906	16, 37, 46
24	4	0.3345	8, 19, 33, 44
25	4	0.4341	7, 16, 29, 42
26	3	0.4092	13, 30, 43
27	3	0.327	16, 33, 44
28	3	0.2984	17, 33, 44
29	4	0.355	8, 19, 32, 43
30	2	0.3419	36, 46
31	2	0.3633	36, 46
32	4	0.3953	8, 22, 38, 46

As for the 21-unit test system, the same method of determining failure rates of the PGUs would have been adopted in the case of the IEEE-RTS, as described in §3, had previous failure times of the PGUs been available. Since this is not the case, however, the failure rate of each PGU was derived from the MTTF provided for each type of PGU in the RTS-79 system [8, 9].

In order to solve the nonlinear model described in §4.3, the objective function was again linearised by means of a piecewise linear approximation, as described in §5.1. The piecewise linear approximation breakpoints for each of the of the expected energy production curves of the PGUs in the 32-unit IEEE-RTS were calculated and are presented in Table 6.8. The number of breakpoints were again selected in such a manner such that the corresponding piecewise linear approximation achieves an accuracy of more than 99.5% with respect to the true expected energy production curve. The error proportion of the piecewise linear approximation and the corresponding number of breakpoints for each PGU are also presented in Table 6.8. The true expected energy production curves of PGU 4, exhibiting the largest number of interior breakpoints, and that of PGU 1 in Table 6.8, exhibiting the smallest number of interior breakpoints, are shown in Figure 6.3. The piecewise linear approximations of these curves are also shown in the figure together with the corresponding number of breakpoints.

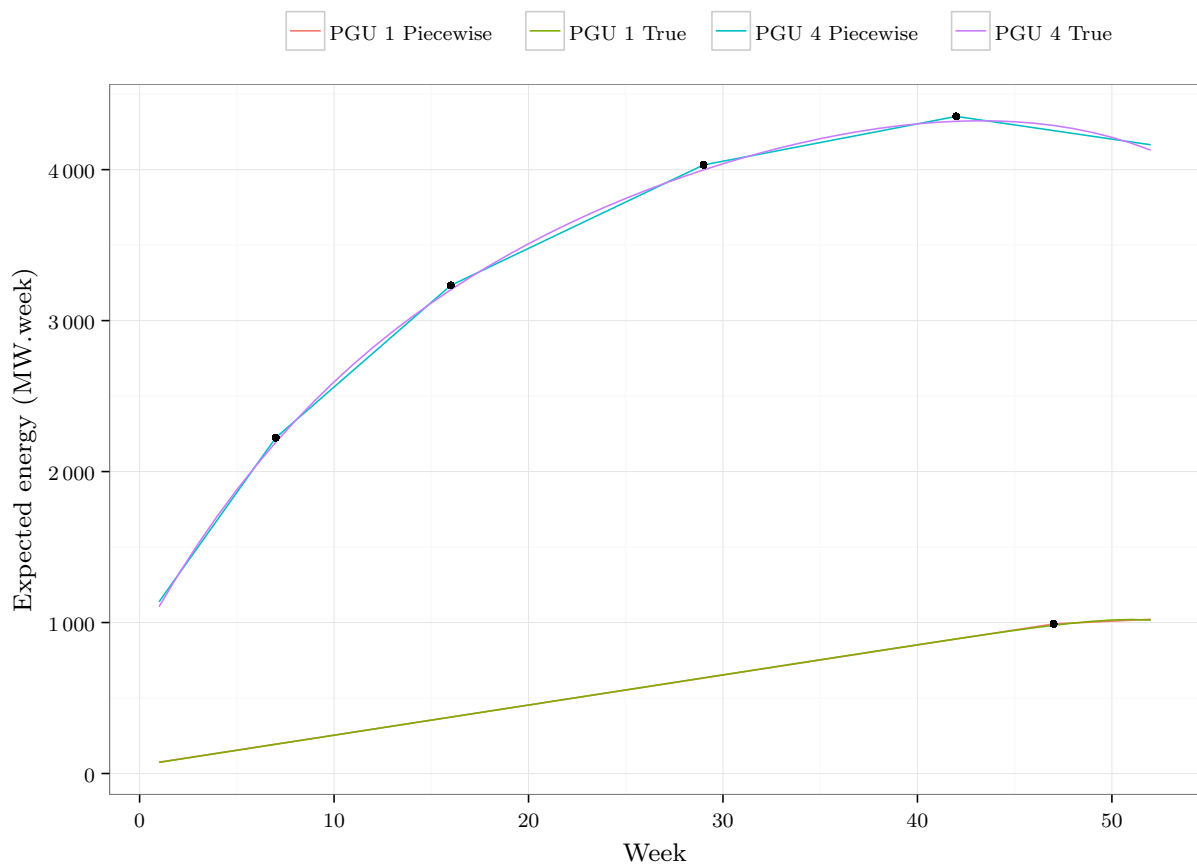


FIGURE 6.3: The true expected energy production curves of two PGUs in the IEEE-RTS together with the corresponding piecewise linear approximation of each curve. The curves represent the expected energy of the PGU with the largest number of breakpoints in Table 6.8 (PGU 4) as well as that of the PGU with the smallest number of breakpoints (PGU 1). The black dots indicate the corresponding breakpoints for each of the piecewise linear approximations.

6.3 Chapter summary

Two test systems in terms of which the effectiveness of the GMS models proposed in §4 and the solution approaches described in §5 are to be tested later in this dissertation were described in this chapter, namely a 21-unit test system and the well-known 32-unit IEEE-RTS. The 21-unit test system is a very basic power system containing twenty one PGUs and exhibiting a constant demand over fifty two one-week planning periods into which the annual scheduling window is discretised, and was presented in §6.1. This test system does not include the specification of a safety margin on the peak demand and also includes no exclusion or precedence constraint specifications. The larger IEEE-RTS was presented in §6.2. This system contains thirty two PGUs and exhibits a varying peak demand, which may be attributed to seasonal demand. This system also contains a 15% safety margin specification over and above the required peak demand of the system. The 32-unit IEEE-RTS has seven exclusion sets which form part of the problem instance specification.

CHAPTER 7

Minimising probability of unit failure

Contents

7.1	Exact solution results	111
7.2	Approximate solution results	124
7.3	Chapter summary	153

The results obtained for the two academic benchmark systems, presented in §6, when adopting the minimisation of the probability of unit failure GMS objective function, as presented in §4.3, is presented in this chapter. This presentation includes the results obtained by means of the exact and the approximate solution approaches described in §5.

7.1 Exact solution results

Exact solutions to the linear GMS model of §4.3 are presented in this section within the context of the two GMS test systems reviewed in §6. These exact solutions are also contrasted with solutions from the literature resulting from the adoption of another GMS objective within the class of reliability criteria. A personal computer, with an Intel CoreTM i7-4770 processor and 8 GB RAM running at 3.4 GHz within a MicrosoftTM Windows 7 64-bit operating system was used to perform all the computational evaluations reported in this chapter.

7.1.1 The 21-unit system

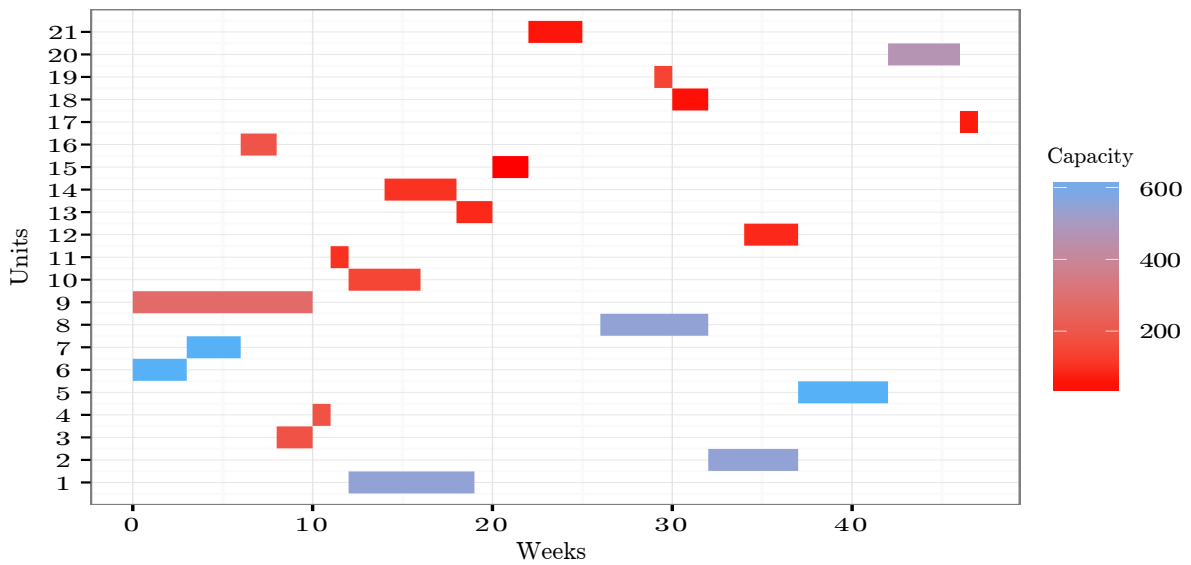
This section contains a presentation of an exact solution obtained by CPLEX for the linear model of §4.3 as applied to the 21-unit test system [54]. The solution is compared to a solution obtained in the literature upon adoption of another reliability scheduling criterion (minimisation of the SSR). The feasibility of an exact solution approach for the linear model in §4.3 within the context of the 21-unit test system is also analysed in the form of a sensitivity analysis involving various relaxations of the demand and maintenance scheduling window constraints.

Numerical results

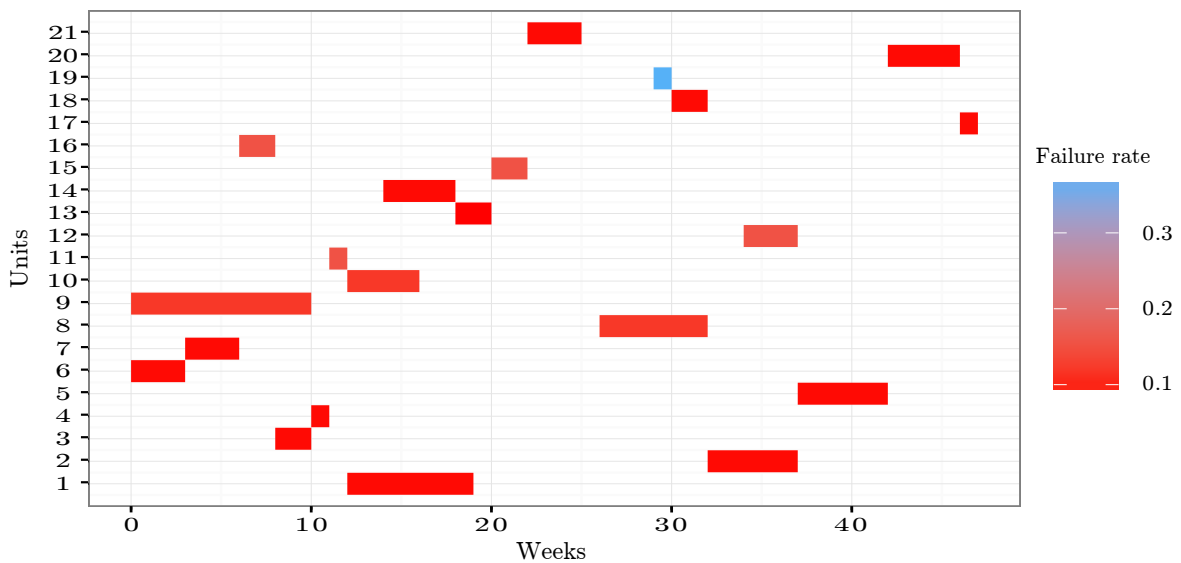
An exact solution to the linear GMS model of §4.3 is obtained for the 21-unit test system by CPLEX within 23 seconds. The optimal decision variable values of this solution are given in integer decision vector form by $\mathbf{x} = [13, 33, 9, 11, 38, 1, 4, 27, 1, 13, 12, 35, 19, 15, 21, 7, 47, 31, 30, 43, 23]$,

which corresponds to an optimal objective function value of 43.542. A graphical representation of this optimal maintenance schedule is presented with the colour scale in Figure 7.1(a) indicating the capacity (in MW) of each PGU and the colour scale in Figure 7.1(b) indicating the failure rate of each PGU. The manpower required over the duration of the scheduling window to implement the optimal solution in Figure 7.1 is shown in Figure 7.2(a). The available capacity over the duration of the scheduling window associated with the optimal solution in Figure 7.1 is shown in Figure 7.2(b). In Figure 7.3, the probability of failure for each of the PGUs is presented, with the maintenance starting at the dates indicated by means of black dots.

The CPLEX model implemented, as shown in Figures 5.2–5.4, terminates once the first optimal solution is found. In order to obtain a complete set of optimal solutions for the 21-unit test

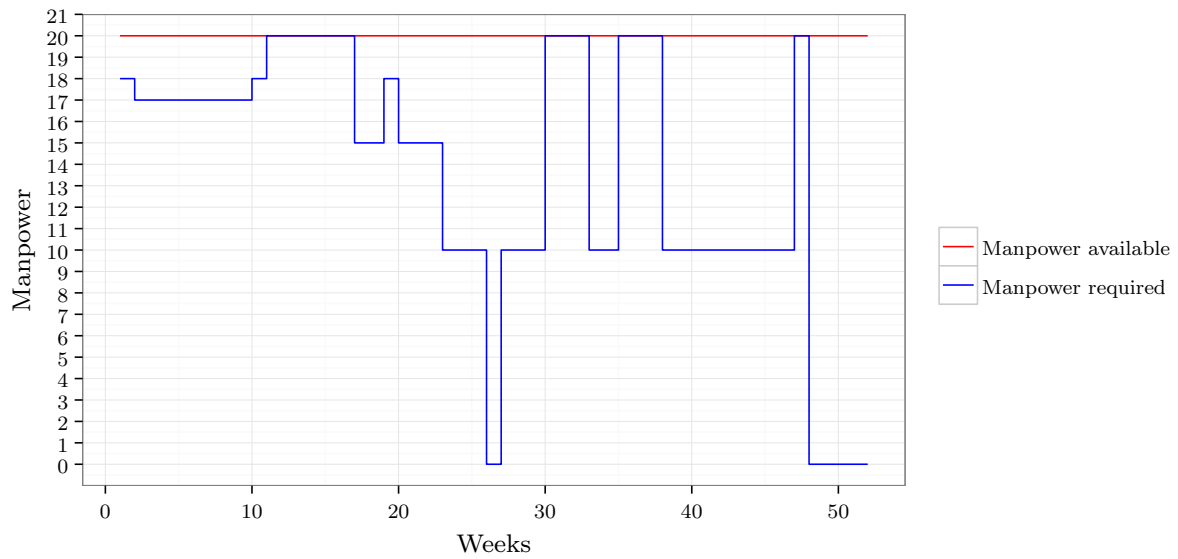


(a) An optimal maintenance schedule with the colour scale indicating the capacity of the PGUs of the 21-unit test system

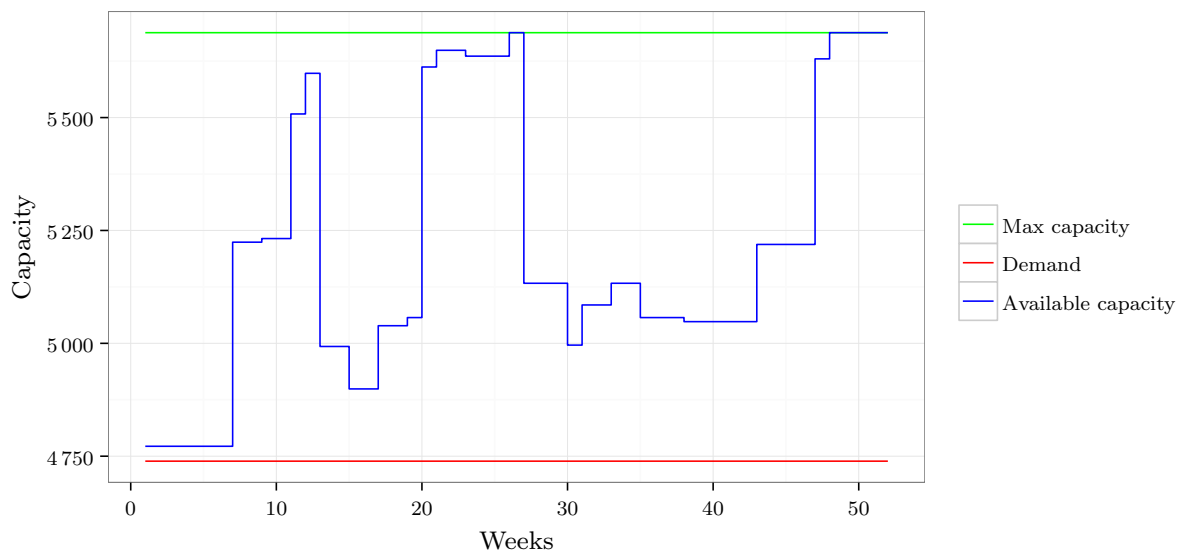


(b) An optimal maintenance schedule with the colour scale indicating the failure rate of the PGUs of the 21-unit test system

FIGURE 7.1: An optimal solution to the linear model of §4.3 for the 21-unit test system.



(a) The manpower required over the scheduling window corresponding to the optimal maintenance schedule in Figure 7.1



(b) The system capacity over the scheduling window corresponding to the optimal maintenance schedule in Figure 7.1

FIGURE 7.2: Evaluation of the manpower required and the capacity available over the duration of the scheduling window for the 21-unit test system.

system, however, a total of 50 seconds of computation time is required. Only one alternative optimal solution¹ to the 21-unit test system is thus found.

A combination of Figures 7.1, 7.2 and 7.3 may be used to analyse the optimal solution to the linear GMS model in §4.3 obtained for the 21-unit test system. For instance, the influence of weighting the probability of failure by the rated capacity of each PGU in the newly proposed scheduling criterion (objective function) is clear in Figure 7.1(a) where two of the largest PGUs in the power system, namely PGU 6 and PGU 7, are scheduled for maintenance to start during

¹This alternative is $\mathbf{x} = [13, 33, 9, 11, 38, 4, 1, 27, 1, 13, 12, 35, 19, 15, 27, 7, 47, 31, 30, 43, 23]$.

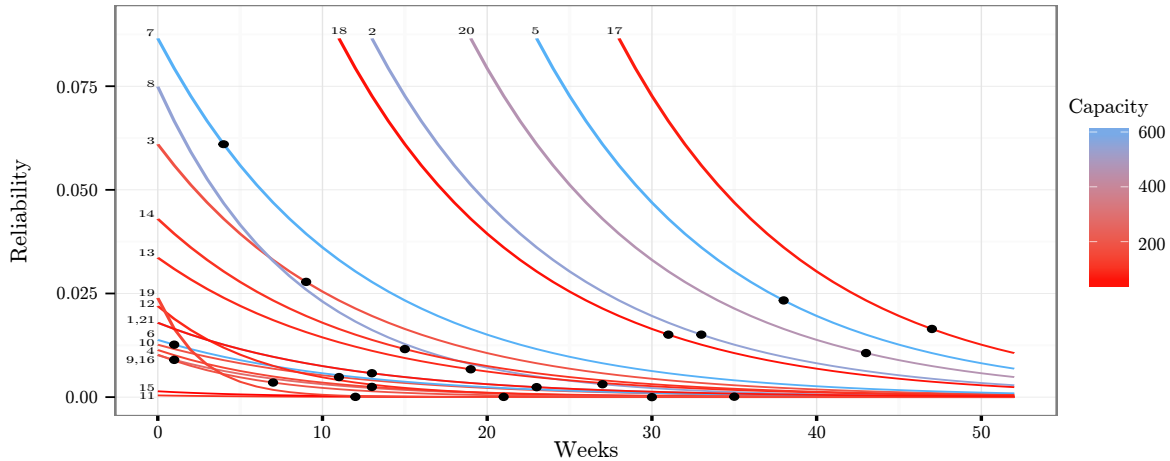


FIGURE 7.3: Graphical representation of the reliability of each PGU of the 21-unit test system with the starting time of planned maintenance indicated by the dot as per the optimal solution shown in Figure 7.1.

planning periods 1 and 4, respectively. Once the maintenance of PGU 6 is completed, the maintenance of PGU 7 is scheduled to commence immediately. It is furthermore clear that good solutions to the linear GMS model of §4.3 will seek to schedule PGUs with larger capacities for maintenance early during the scheduling window. This may be seen in Figure 7.2(b) where, during the early stages of the scheduling window, the available capacity is very close to the demand after which it increases toward the middle of the scheduling window. The available capacity is observed to diminish again during planning period 27, which is typically when the second set of maintenance window constraints start. At the end of the scheduling window, it may also be observed that no maintenance is scheduled as all the PGUs have already been serviced and so the maximum capacity is available (*i.e.* no manpower is required for maintenance scheduling), as indicated in Figure 7.2(a). The same effect may be observed during planning period 26 when no planned maintenance is scheduled, because all the PGUs which have maintenance windows within the first half of the scheduling window, have already been serviced.

The newly proposed GMS objective function does not only give preference to the maintenance of PGUs with large capacities, but also to PGUs with large failure rates, as may be seen in Figure 7.1(b). The PGU with the largest failure rate is PGU 19 which only has a capacity of 137 MW and has an earliest maintenance starting time of 27. The optimal solution shows that the maintenance of PGU 19 is scheduled to commence during planning period 30, which is earlier than the maintenance starting time of PGUs 2, 5, 8 and 20 (which all have larger capacities than PGU 19). The effects of failure rates on maintenance scheduling is better expressed in Figure 7.3 where it may be seen that PGUs with high failure probabilities are typically scheduled for maintenance earlier than PGUs with low failure probabilities.

Comparison with results from the literature

The effects of taking into account the probability of failure as well as the rated capacity of each PGU, as in the newly proposed objective function of §4.3.2, may profitably be analysed by comparing the results reported above with results found in the literature when adopting other reliability scheduling criteria, such as minimisation of the SSR. The aforementioned results are therefore compared in this section with the results obtained by Schlünz and Van Vuuren [188] who adopted the SSR scheduling criterion in (2.8). The results of Schlünz and Van Vuuren [188]

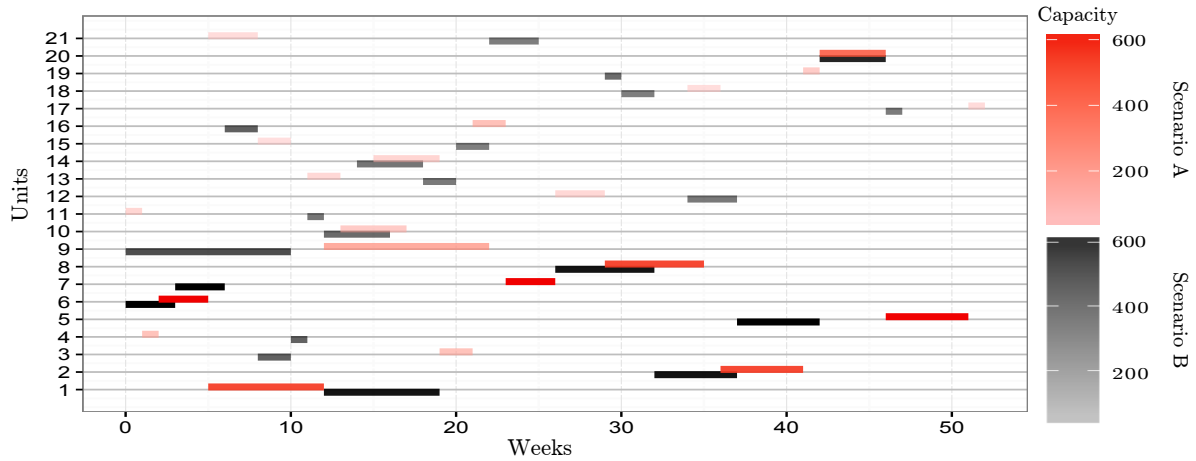
(referred to here as Scenario A) are compared with the results reported above (referred to here as Scenario B). Both GMS objectives adopted in Scenarios A and B reside within the class of the reliability scheduling criteria and the results of Schlünz and Van Vuuren[188] for the 21-unit test system (with the minimisation of SSR as objective) represent the best results in the literature for this particular scheduling criterion and problem instance combination. A graphical representation of the two maintenance schedules for Scenarios A and B and their effects on the manpower required and the available capacity for the 21-unit test system is shown in Figure 7.4. In Figure 7.4(a), the maintenance schedules of the two scenarios are compared, with the colour scale indicating the capacity of the PGUs in the system. The effects of the two maintenance schedules on the manpower required and the available capacity are shown in Figures 7.4(b) and 7.4(c), respectively. A comparison between the results for the two scenarios in terms of both scheduling objectives is presented in Table 7.1. The optimal solution of Scenario B performs 17.519% worse than that of Scenario A in terms of the SSR scheduling objective. A similar observation is made when taking the best solution obtained for the newly proposed objective. In this case, the solution obtained for Scenario B performs 10.34% better than that for Scenario A. A reason for this is that the two objectives conflict with each other in terms of when maintenance should be scheduled for the PGUs. In Scenario A, the objective function aims to spread the maintenance of PGUs with large capacities out over the entire scheduling window in order to levelise the reserves of the system. The objective function of Scenario B, on the other hand, aims to schedule planned maintenance for the PGUs as early as possible during the scheduling window.

TABLE 7.1: Comparison between objective function values associated with the maintenance schedules in Figure 7.4(a) for Scenarios A and B in the context of the 21-unit test system. The asterisk indicates an optimal solution. The percentage change values are computed for the solution of Scenario B relative to that of Scenario A.

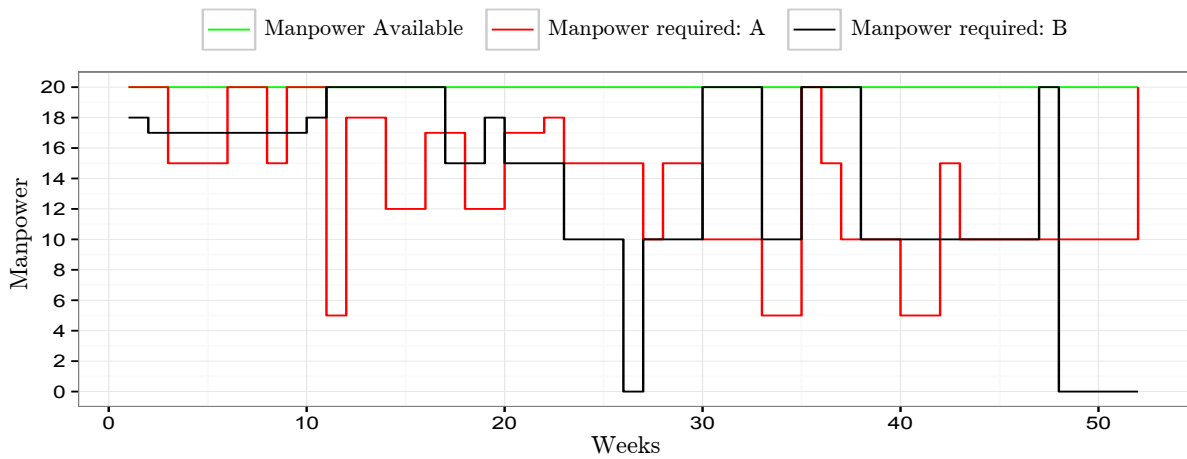
Scenario	SSR (MW ²)	Measure of probability of failure
A	13 664 879	48.042
B	16 567 361	43.542*
Percentage change	+17.519%	-10.34%

It may be seen in Figure 7.4(a) that PGUs with large capacities are scheduled as early as possible, within their respective PGU maintenance windows. It may also be seen in Figure 7.4(a) that, compared to Scenario A, some PGUs with higher failure rates are scheduled for planned maintenance earlier during the scheduling window. PGU 9, for example, is scheduled for maintenance during the first week of the scheduling window due to its 0.1178 failure rate, which is 13 weeks earlier than in Scenario A. A similar observation may be made for PGU 16, which is scheduled for maintenance 15 weeks earlier in Scenario B than in Scenario A. PGU 19 has the highest failure rate of all the PGUs in the system, but has an earliest starting time for maintenance at 27 weeks. In Scenario B, planned maintenance of PGU 19 is scheduled to start during week 30, which is 12 weeks before the planned maintenance of PGU 19 in Scenario A.

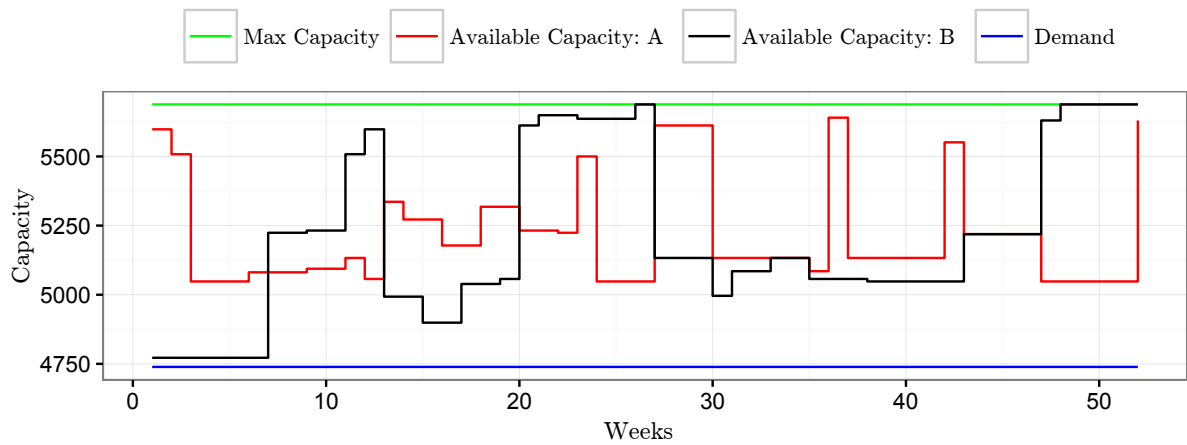
The different effects of the two scheduling objectives may also be observed in Figure 7.4(b). In Scenario A, the manpower required for planned maintenance is spread out over the entire scheduling window. During each planning period of the problem instance, manpower is required as there is no planning period within the scheduling window during which no PGUs are in maintenance. In Scenario B, on the other hand, the maximum amount of manpower is often required during the early stages of the scheduling window. It is also observed that at the end



(a) Two maintenance schedules for the 21-unit test system corresponding to different GMS criteria



(b) The manpower required over the duration of the scheduling window for the two maintenance schedules in Figure 7.4(a)



(c) The available system capacity over the duration of the scheduling window for the two maintenance schedules in Figure 7.4(a)

FIGURE 7.4: Comparison between the maintenance schedules of Scenarios A and B for the 21-unit test system.

of both halves of the scheduling window (*e.g.* weeks 1–27 and weeks 28–52) no manpower is required as no maintenance is scheduled during these times.

Similar observations may also be made in respect of Figure 7.4(c). In Scenario A, the available capacity never falls below a 6% gap between the demand and the available capacity. The available capacity mainly occurs in the middle of the graph and, compared to Scenario B, fewer jumps are observed in the available capacity. In Scenario B, on the other hand, the available system capacity drops down to 0.696% above the demand during the early stages of the scheduling window.

Sensitivity analysis

An exact approach towards solving the linear GMS model of §4.3 for large power systems or very unconstrained systems is not expected to be feasible. The feasibility of an exact solution approach by CPLEX is of course also influenced by the nature of the objective function (*e.g.* linear or nonlinear). It was demonstrated in §7.1.1 that employing such an exact model solution approach in the context of the 21-unit test system is feasible.

In order to analyse the effects of alterations in the system specifications on the exact model solution approach, six cases are analysed in this section in terms of the computation time required by CPLEX to solve the linear model of §4.3. These cases involve combinations of increasing the peak demand of the system by a certain margin and relaxing the maintenance window constraints to have an earliest starting time of 1 and a latest starting time of 53 less the duration of maintenance of each PGU. The first case is the original 21-unit test system which is considered as a reference case for the other five cases. The second case involves a 3% increase in the peak demand, but adheres to the original test system's maintenance window constraints. The third case involves a 6.5% increase in the peak demand, but also adheres to the original maintenance window constraints. In the fourth case, the peak demand is kept as specified for the original 21-unit test system, but the maintenance window constraints are relaxed as described above. The fifth case involves a 3% increase in the peak demand and relaxed maintenance window constraints. Finally, the sixth case involves an increase in the peak demand of 6.5% and relaxed maintenance window constraints. Various statistics pertaining to optimal solutions of the linear model in §4.3 are shown for these six cases in Table 7.2.

Case 1 requires 23 seconds of computing time by CPLEX to obtain an optimal solution. In Case 2 the demand is increased by 3%, which requires 26 seconds of computing time and yields a 1.470% worsening of the objective function value. This is a small increase in computing time

TABLE 7.2: Various statistics pertaining to optimal solutions of the linear model of §4.3 in a sensitivity analysis in respect of demand and PGU maintenance windows for the 21-unit test system. The last column contains gap values with respect to provable lower bounds on the minimum objective function value. An asterisk denotes that a time-out budget of 4 hours of computation time was reached by CPLEX.

Cases	Demand (%)	Maintenance window	Objective function value	Time (s)	Gap (%)
1	100	Original	43.542	23	0
2	103	Original	44.182	26	0
3	106.5	Original	44.409	28	0
4	100	Relaxed	37.768	2 264	0
5	103	Relaxed	39.018	4 929	0
6	106.5	Relaxed	40.784*	14 400	5.03

and an optimal solution is still obtained. Furthermore, when increasing the demand by 6.5%, in Case 3, a computation time of 28 seconds is required to obtain the optimal solution which results in a 1.991% worsening of the objective function value compared to that in Case 1. It is observed that increasing the demand has a small impact on the computation time required to solve the linear model of §4.3 for the 21-unit test system. This computation time increases by only 5 seconds over the course of a 6.5% increase in demand.

When the maintenance window constraints are relaxed, however, the number of optimal solutions to the linear model of §4.3 increases drastically, and so does the required computation time. Case 4, in which the demand is kept as specified for the original 21-unit test system and the maintenance window constraints are relaxed, results in a large increase in computation time. A total of 2 264 seconds of processing time is required in this case to obtain the first optimal solution, which is 13.261% better than that in Case 1. A further increase in computation time is observed in Case 5, where the demand is increased by 3% and the maintenance window constraints are relaxed. The first optimal solution is obtained within 4 929 seconds of computation time (which is more than double the computation time required for Case 4) and yields an objective function value that is 10.390% better than that of Case 1. Finally, in Case 6, it is found that no optimal solution can be obtained within the time-out budget of 14 400 seconds of processing time. A gap of 5.03% is obtained between the best objective function value (of 40.784) and the largest provable lower bound on the objective function within four hours of computation time.

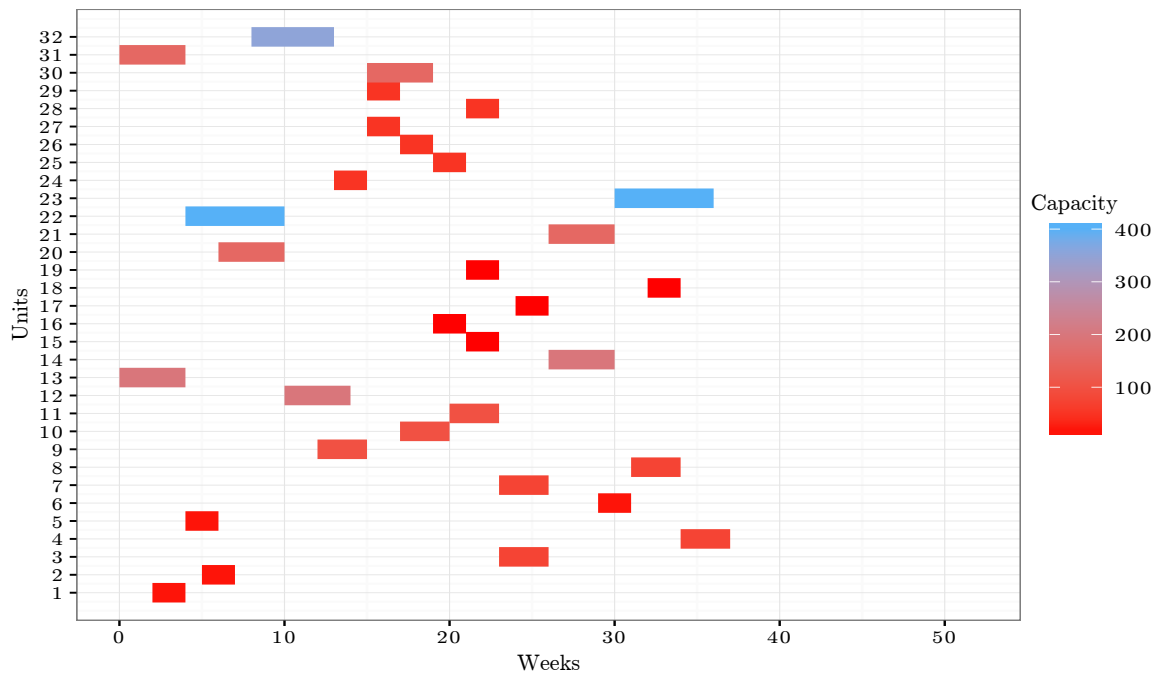
7.1.2 The IEEE-RTS

In this section, exact solutions to the linear GMS model of §4.3 obtained by CPLEX for the IEEE-RTS [188] are presented and compared with solutions in the literature obtained when scheduling according to another reliability scheduling criterion (minimisation of the SSR). The practical feasibility of an exact solution approach for the IEEE-RTS is also analysed in the form of a sensitivity analysis.

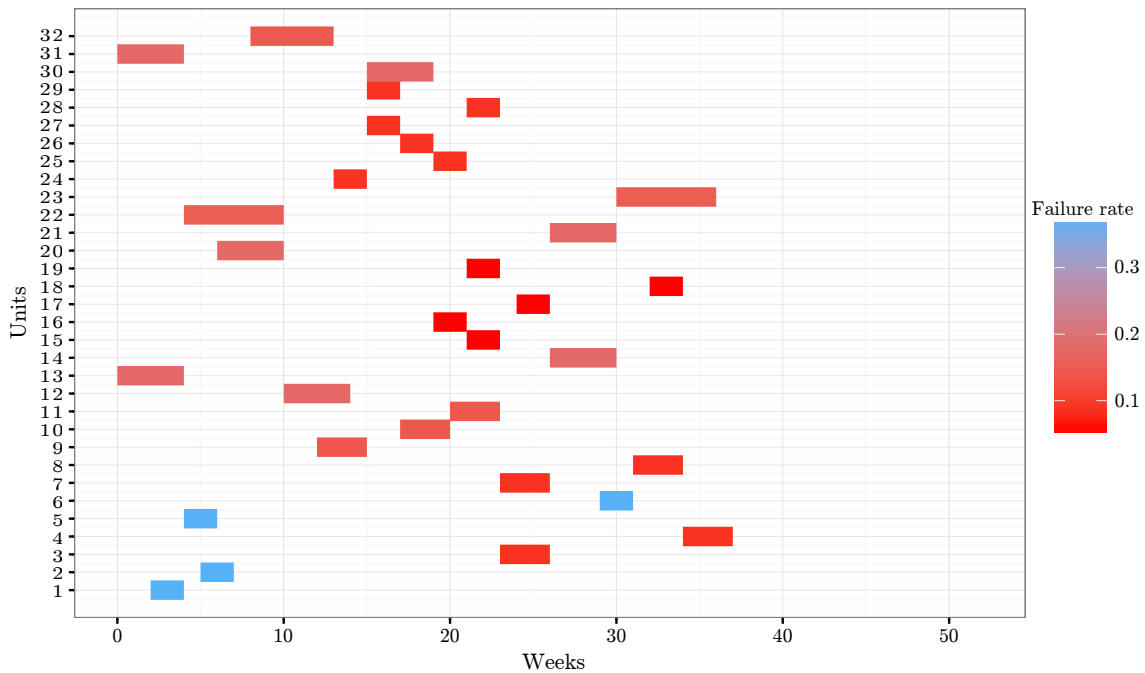
Numerical results

An exact solution to the linear GMS model of §4.3 is obtained for the IEEE-RTS by CPLEX within 382 seconds. The optimal decision variable values of this solution are given in integer decision vector form by $\mathbf{x} = [3, 6, 24, 35, 5, 30, 24, 32, 13, 18, 21, 11, 1, 27, 22, 20, 25, 33, 22, 7, 27, 5, 31, 14, 20, 18, 16, 22, 16, 16, 1, 9]$, which corresponds to an optimal objective function value of 57.886. A graphical representation of the optimal maintenance schedule is presented in Figure 7.5 with the colour scale in Figure 7.5(a) indicating the capacity (in MW) of each PGU and the colour scale in Figure 7.5(b) indicating the failure rate of each PGU. The manpower required over the duration of the scheduling window to implement the optimal solution in Figure 7.5 is shown in Figure 7.6(a). The available capacity over the duration of the scheduling window associated with the optimal solution is further shown in Figure 7.6(b).

As mentioned before, the implemented CPLEX model terminates once the first optimal solution is found. In order to obtain a complete set of alternative optimal solutions for the IEEE-RTS, a total of 2 127 seconds of computation time is required during which a total 4 147 200 optimal solutions are found for the IEEE-RTS. These alternative solutions are not reported here individually due to the large number of alternative solutions. All of the 4 147 200 optimal solutions may be found by exchanging the maintenance commencement dates within certain groups of PGUs in the system. The PGUs in each of these groups have the exact same specifications in terms of capacity, maintenance duration, earliest and latest maintenance commencement dates and



(a) An optimal maintenance schedule with the colour scale indicating the rated capacity of the PGUs of the IEEE-RTS

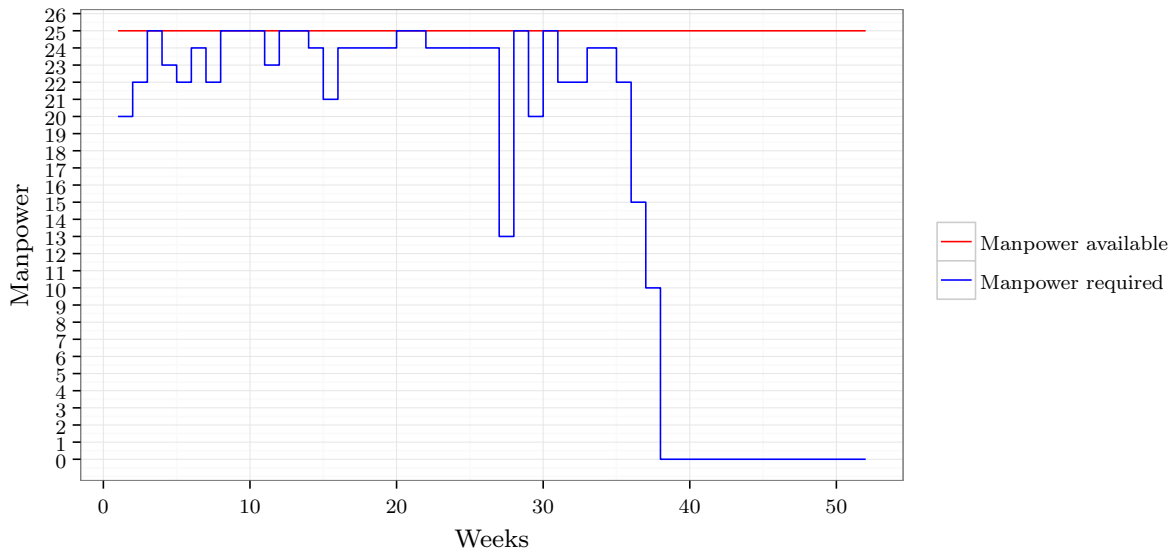


(b) An optimal maintenance schedule with the colour scale indicating the failure rate of the PGUs of the IEEE-RTS

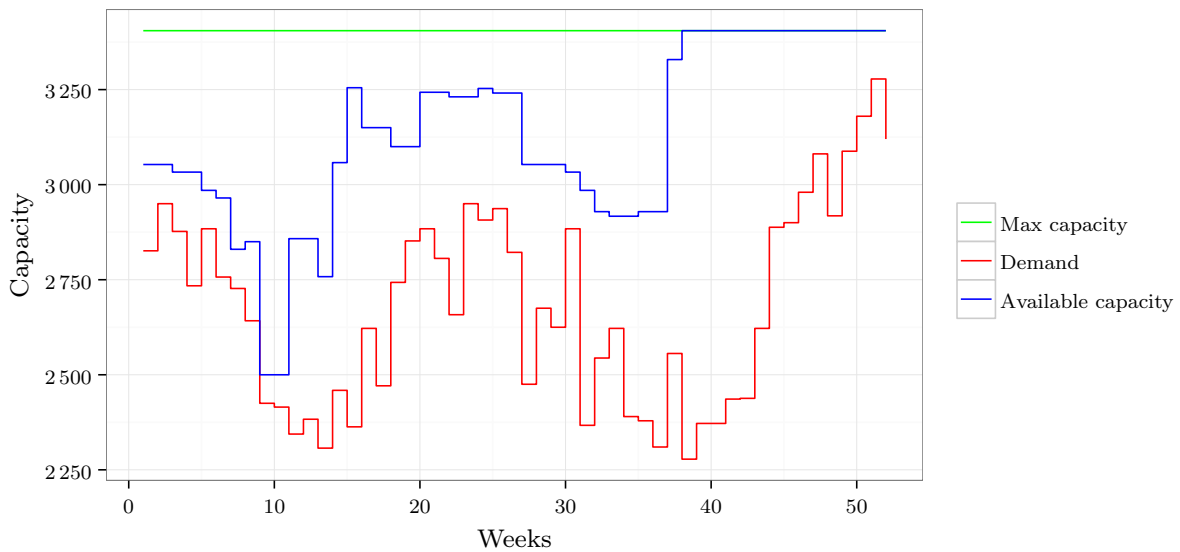
FIGURE 7.5: An optimal solution to the linear model of §4.3 for the IEEE-RTS.

failure rates, and hence these PGUs are interchangeable in any optimal maintenance schedule. There are six of these interchangeable groups of PGUs, as presented in Table 7.3.

Multiplying the number of interchangeable combinations of the six groups presented in Table 7.3, one arrives at the total number of alternative optimal solutions of 4 147 200. These groups also correspond more or less to the groups of PGUs that form exclusion sets, as presented in Table 6.5.



(a) The manpower required over the scheduling window corresponding to the optimal maintenance schedule in Figure 7.5



(b) The system capacity over the scheduling window corresponding to the optimal maintenance schedule in Figure 7.1

FIGURE 7.6: Evaluation of the manpower required and system capacity available over the duration of the scheduling window for the IEEE-RTS.

Figures 7.5 and 7.6 may be used to analyse the optimal solutions obtained for the IEEE-RTS. For this test system, a clear trade-off is observed in terms of when maintenance is scheduled between the rated capacity of a PGU and the failure rate of a PGU. In Figure 7.5(a), it is observed that PGUs 13, 20, 22 and 31, which have large rated capacities, are scheduled for maintenance early during the scheduling window. Along with these PGUs, PGUs 1, 2 and 5 are also scheduled for planned maintenance early during the scheduling window, although in this case due to their high failure rates. It is also observed that maintenance of all the PGUs is completed during planning period 38, which is 14 planning periods before the end of the scheduling window. This is attributed to the fact that the objective function seeks to schedule planned maintenance as

TABLE 7.3: Six groups of PGUs which have identical specifications. Any two PGUs within the same group are interchangeable in an optimal maintenance schedule for the IEEE-RTS.

Group	PGUs	Number of combinations
1	1,2	2
2	9,10,11	6
3	12,13	2
4	15,16,17,18,19	120
5	24,25,26,27,28,29	720
6	30,31	2

early as possible, within the respective maintenance window constraints of the PGUs, in order to minimise the probability of failure.

The fact that no maintenance is scheduled after planning period 38, translates to the manpower and capacity depicted in Figures 7.6(a) and 7.6(b), respectively. After planning period 38, it may be observed in Figure 7.6(a) that no manpower is required. Similarly, for the capacity in Figure 7.6(b), it may be seen that after planning period 38 the total capacity of the system is available.

Comparison with results from the literature

As for the 21-unit test system, the IEEE-RTS maintenance schedule obtained by solving the linear GMS model of §4.3 may better be analysed by comparing it to a maintenance schedule proposed by Schlünz and Van Vuuren [188] for the IEEE-RTS. Schlünz and Van Vuuren adopted minimisation of the SSR as scheduling criterion in (2.8) instead of minimising the risk of PGU failure as is pursued in the linear model of this dissertation. The schedule obtained by Schlünz and Van Vuuren [188] (referred to here as Scenario C) may be compared directly with the schedule above (referred to here as Scenario D) due to both objectives residing within the class of the reliability scheduling criteria. The maintenance schedule by Schlünz and Van Vuuren [188] is the best-quality schedule available in the literature for the IEEE-RTS under the SSR scheduling criterion, and was found by employing the method of SA. A graphical representation of the two maintenance schedules of Scenarios C and D are shown in Figure 7.7 with the colour scale indicating the capacities of the PGUs in the system. The effects of the two maintenance schedules on the manpower required and the system capacity available are shown in Figures 7.8(a) and 7.8(b), respectively.

A comparison between the schedules of the two scenarios in terms of both objective functions is shown in Table 7.4. It may be seen in this table that the schedule of Scenario D performs 7.91% worse in terms of minimisation of the SSR objective than Scenario C. The result obtained in Scenario C, on the other hand, performs 15.34% worse in terms of the newly proposed linear scheduling objective than the schedule of Scenario D. Again this is indicative of the conflicting nature of the two scheduling objectives, as described in §7.1.1.

The conflicting nature of the two objective functions in Scenarios C and D is clearly visible in Figure 7.8 where the manpower required and the available capacity are compared. In Figure 7.8(a), it may be observed that throughout the duration of the scheduling window, the required manpower for Scenario C is on average, between 10 and 20, with the manpower reaching the maximum amount of manpower available during a few planning periods only. For Scenario D, on the other hand, it may be observed that during the early stages of the maintenance

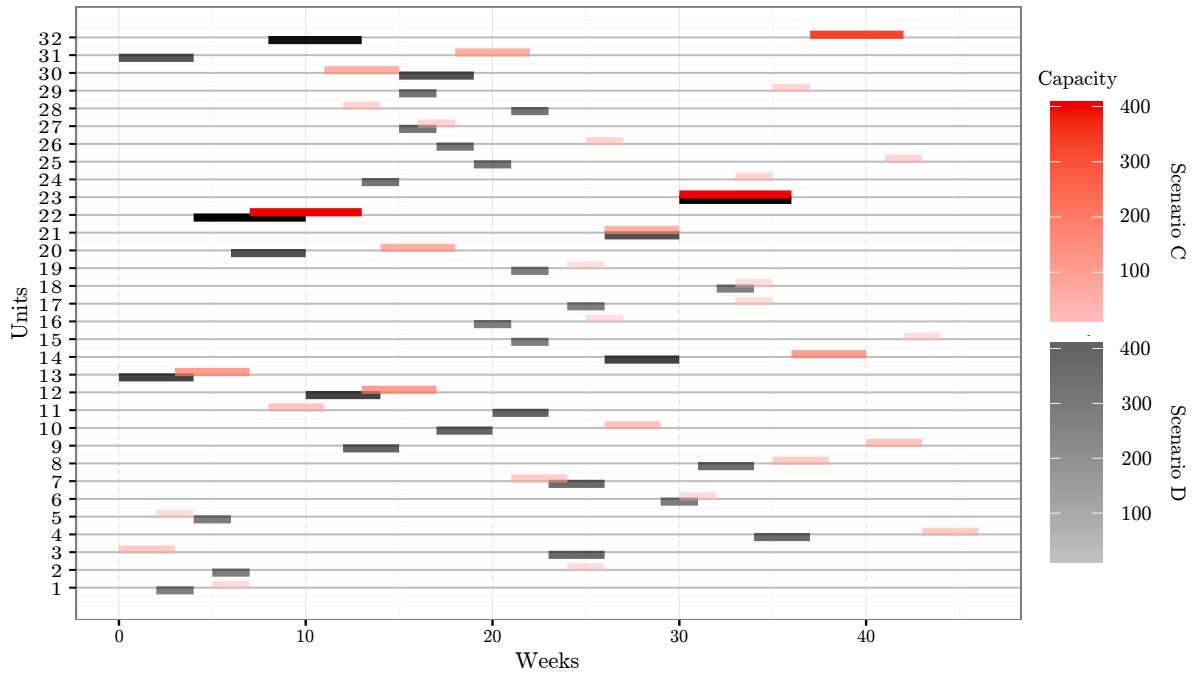


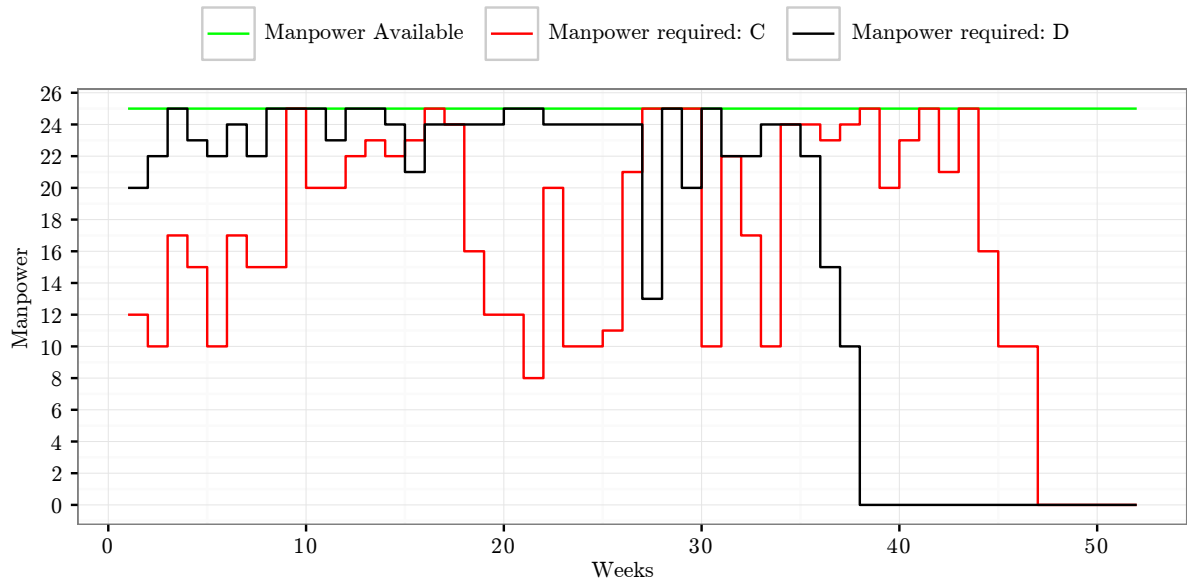
FIGURE 7.7: Comparison between the maintenance schedules of Scenarios C and D for the IEEE-RTS.

window, the manpower required is between 20 and 25, and that it always remains close to the maximum amount of manpower available.

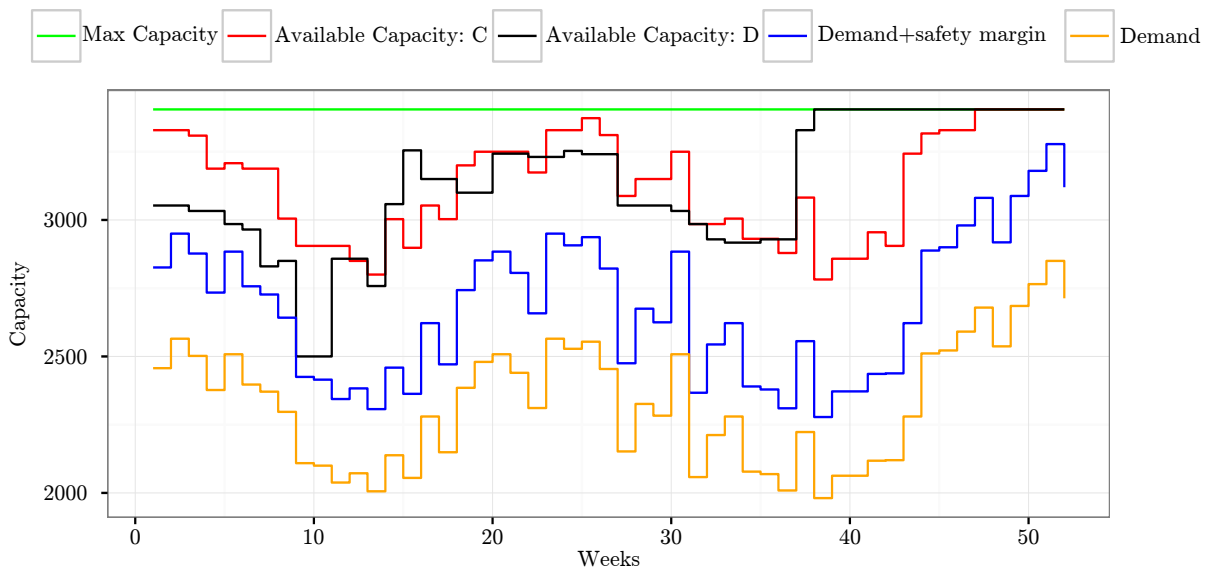
The SSR scheduling objective in Scenario C, seeks to level the reserve load over all the planning periods of the maintenance schedule. This is clearly observable in Figure 7.8(b) where the available capacity mimics the increases and decreases in the peak demand (together with the required safety margin). The scheduling objective in Scenario D, maximising the weighted probability that no PGU fails during the scheduling window, seeks to schedule planned maintenance for the PGUs in the system as soon as possible within the constraints of the system. This is also observable in Figure 7.8(b) where, during the early stages of the scheduling window, the available system capacity is close to the peak demand (together with the required safety margin). As fewer PGUs require maintenance toward the end of the maintenance schedule, the gap between the peak demand and available system capacity starts to increase.

TABLE 7.4: Comparison between objective function values associated with the maintenance schedules in Figure 7.7 for Scenarios C and D in the context of the IEEE-RTS. The asterisk indicates an optimal solution. The percentage change values are computed for the solution of Scenario D relative to that of Scenario C.

Scenario	SSR (MW ²)	Measure of probability of failure
C	33 627 292	66.741
D	36 514 792	57.866*
Percentage change	+7.91%	-15.34%



(a) The manpower required over the duration of the scheduling window for the two maintenance schedules in Figure 7.7



(b) The available system capacity over the duration of the scheduling window for the two maintenance schedules in Figure 7.7

FIGURE 7.8: Comparison between the manpower required and the available system capacity of Scenarios C and D for the IEEE-RTS.

Sensitivity analysis

It was demonstrated in §7.1.2 that employing an exact solution approach towards solving the linear GMS model of §4.3 in the context of the IEEE-RTS is feasible.

In order to analyse the effects of alterations in the system specifications on the practical feasibility of the exact solution approach, four cases are analysed in this section in terms of the time required to solve the linear model of §4.3 for the IEEE-RTS using CPLEX. In each of these cases, CPLEX is allowed 28 800 seconds (8 hours) of computation time. These cases involve combinations of increasing the peak demand of the system by a certain percentage and relaxing the maintenance window constraints to have an earliest starting time of 1 and a latest starting time of 53 less the duration of each PGU, as in §7.1.1. The first case is the original IEEE-RTS which is considered as a reference case for the other three cases. The second case involves a 3% increase in the peak demand, but adheres to the original test system's maintenance window constraints. In the third case, the peak demand is kept as specified for the original IEEE-RTS, but the maintenance window constraints are relaxed. Finally, the fourth case involves both a 3% increase in the peak demand and a relaxation of the maintenance window constraints. The objective function values and required computation times for these cases are shown in Table 7.5.

TABLE 7.5: Various statistics pertaining to optimal solutions to the linear model of §4.3 in a sensitivity analysis in respect of demand and PGU maintenance windows for the IEEE-RTS. The last column contains gap values with respect to provable lower bounds on the minimum objective function value. An asterisk denotes that a time-out budget of 8 hours of computation time was reached by CPLEX.

Cases	Demand (%)	Maintenance window	Objective function value	Time (s)	Gap (%)
1	100	Original	57.886	382	0
2	103	Original	57.886	296	0
3	100	Relaxed	55.169	28 800	2.33
4	103	Relaxed	55.827	28 800	3.31

As may be seen in the table, Case 1 requires 382 seconds of computing time to obtain an optimal solution, whereas Case 2 requires a computation time of 296 seconds, which is less than that required for Case 1, and results in the same objective function value as in Case 1. The reason for the decrease in computation time might be because there exist fewer feasible solutions that have to be evaluated in order to obtain an optimal solution for the problem instance. The reason for observing no change in the objective function values of Cases 1 and 2 is that even with the increased demand, the same maintenance schedule is optimal for Case 2 as for Case 1. For Case 3, it is observed that no optimal solution can be found in the allowed 28 800 seconds of computation time. The percentage gap obtained within the 28 800 seconds of processing time is 2.33%. In the final case, an optimal solution can also not be found in the allocated time. After 28 800 seconds of computation time, the percentage gap obtained for Case 4 is 3.31%.

7.2 Approximate solution results

In this section, an experimental design is carried out according to which suitable parameter values for the SA algorithm may be selected for the linear model of §4.3. This takes the form of an extensive parameter optimisation experiment in the contexts of both the 21-unit test system [54] and the 32-unit IEEE-RTS [188] described in §6. Thereafter, a presentation follows of the results obtained by the SA algorithm for these two test systems when adopting the

best parameter value combination uncovered during the parameter optimisation experiment. The approximate solutions thus obtained are finally compared with the results obtained by the exact solution approach described in §7.1. The SA solution approach described in §5.3 was implemented in the software package R [175] in combination with RStudio [184] as the *integrated development environment* (IDE) for R.

7.2.1 The 21-unit system

This section is devoted to an application of the parameter optimisation experiment described in §5.3.3 within the context of the linear GMS objective of §4.3 for the 21-unit test system of §7.1.1. The best algorithmic parameter combination thus uncovered is then employed to solve the test instance approximately.

Parameter optimisation experiment

In this section, a parameter optimisation experiment is performed for the 21-unit test system in two separate phases according to the design described in §5.3.3. The first phase of the experiment involves variation of the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ . During the second phase of the experiment, the parameters varied are the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ .

Phase 1: Initial acceptance ratio and soft constraint violation severity factor

The mean optimality gaps (measured as percentages relative to the optimal objective function values obtained by CPLEX) for the feasible incumbents returned during the first phase of the parameter optimisation experiment are shown in Table 7.6. The mean computation times involved in evaluating combinations of these parameter values are shown in Table 7.7, including computation times expended during runs that returned infeasible incumbents. The numbers of times (out of 30) that an infeasible incumbent was returned during an SA search run are finally shown in Table 7.8.

TABLE 7.6: Mean optimality gaps for all parameter combinations of the first phase of the parameter optimisation experiment involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ for the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion.

χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	0.168	2.739	3.917	3.754	4.736
0.5	1.248	3.905	3.559	4.238	4.703
0.6	1.059	2.921	4.020	3.797	4.594
0.7	1.523	3.280	3.985	4.280	4.434
0.8	0.364	3.330	3.996	4.578	5.030

In these results a correlation is observed between the mean optimality gap and the soft constraint violation severity factor. As the severity factor increases, it is observed that the mean optimality gap also increases. This may also be seen in Figure 7.9(a) where a box plot comparison is presented of the mean optimality gap as a function of the soft constraint violation severity factor. The reason for this correlation may be that the search space contains many disjoint

TABLE 7.7: Mean computation times required for all parameter combinations of the first phase of the parameter optimisation experiment involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ for the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion.

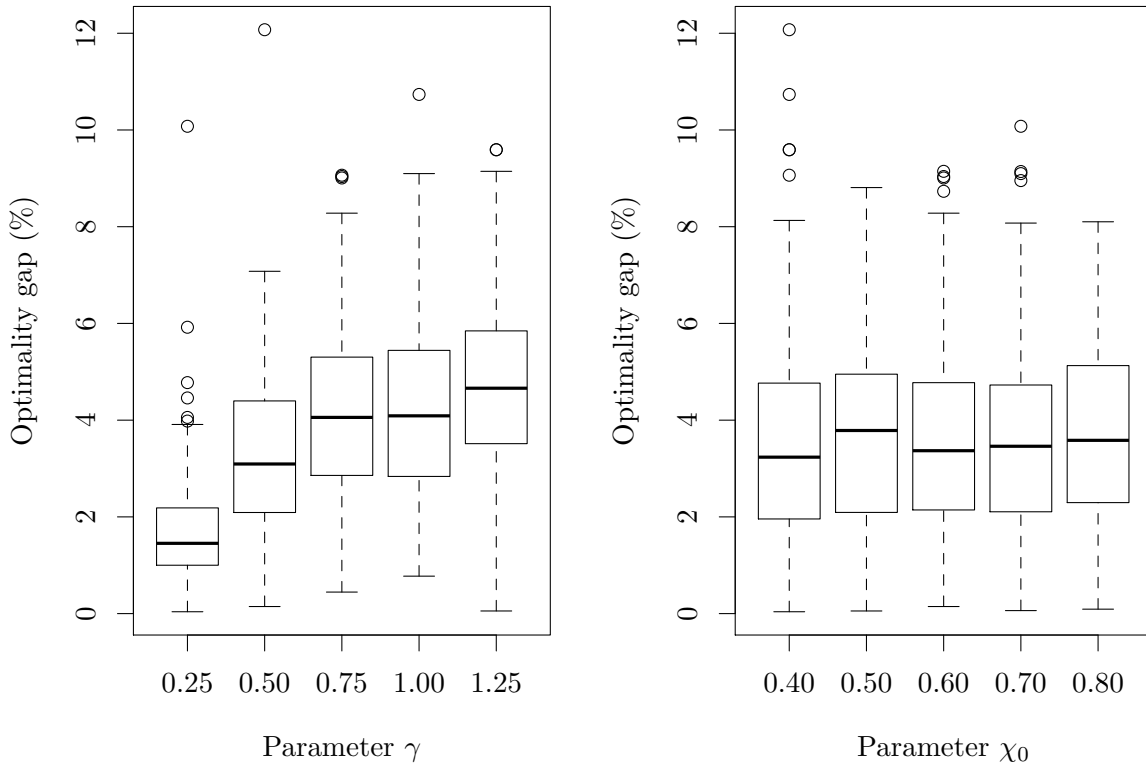
χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	11144	9797	7604	9596	8525
0.5	11481	9327	8086	8982	10459
0.6	10636	9290	8917	8296	8515
0.7	11526	9479	9805	8925	8142
0.8	10439	10488	10206	8116	9464

TABLE 7.8: Number of infeasible incumbents (out of 30) returned for all parameter combinations of the first phase of the parameter optimisation experiment involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ for the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion.

χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	22 (73.33%)	15 (50.00%)	8 (26.67%)	4 (13.33%)	2 (6.67%)
0.5	24 (80.00%)	9 (30.00%)	7 (23.33%)	5 (16.67%)	3 (10.00%)
0.6	26 (86.67%)	17 (56.67%)	6 (20.00%)	2 (6.67%)	1 (3.33%)
0.7	24 (80.00%)	17 (56.67%)	10 (33.33%)	4 (13.33%)	3 (10.00%)
0.8	25 (83.33%)	12 (40.00%)	3 (10.00%)	4 (13.33%)	2 (6.67%)

pockets of feasible regions. Therefore, in order to move from one such pocket of feasibility to a different pocket of feasibility, infeasible solutions have to be accepted during the transition in view of the fact that the SA algorithm is a trajectory-based metaheuristic. This type of transition is, however, less probable when adopting a larger soft constraint violation severity factor. It is therefore observed that the smallest value of the soft constraint violation severity factor leads to the discovery of feasible incumbents during the SA search that are close to optimal. It is, however, also observed that this parameter combination results in many outliers. The effect of the initial acceptance ratio on the mean optimality gap seems to be less prominent. Almost no increase or decrease is observed in the mean optimality gap as the initial acceptance ratio is increased. This is clearly visible in the box plot comparison presented in Figure 7.9(b). It is observed that the spreads of the mean optimality gaps for all five of the initial acceptance ratio values are fairly similar. The results obtained in respect of the initial acceptance ratio are similar. This similarity in mean optimality gaps may be due to the fact that the SA algorithm typically terminates as a result of the three-consecutive-epoch termination criterion and not the maximum allowable computation time. The algorithm therefore has enough time to reach an acceptably good incumbent solution, even when the initial temperature is higher in some cases (as is the case for a higher initial acceptance ratio).

In terms of the computation time expended to evaluate each of the parameter combinations presented in Table 7.7, some level of correlation is noticeable between the soft constraint violation severity factor and the mean required computation time. As the soft constraint violation severity factor increases, a decrease in the average computation time is observed. This downward trend is visible in Figure 7.10(a), which contains a plot of the computation time as a function of the



(a) Box plot comparison of the optimality gap as a function of the soft constraint violation severity factor

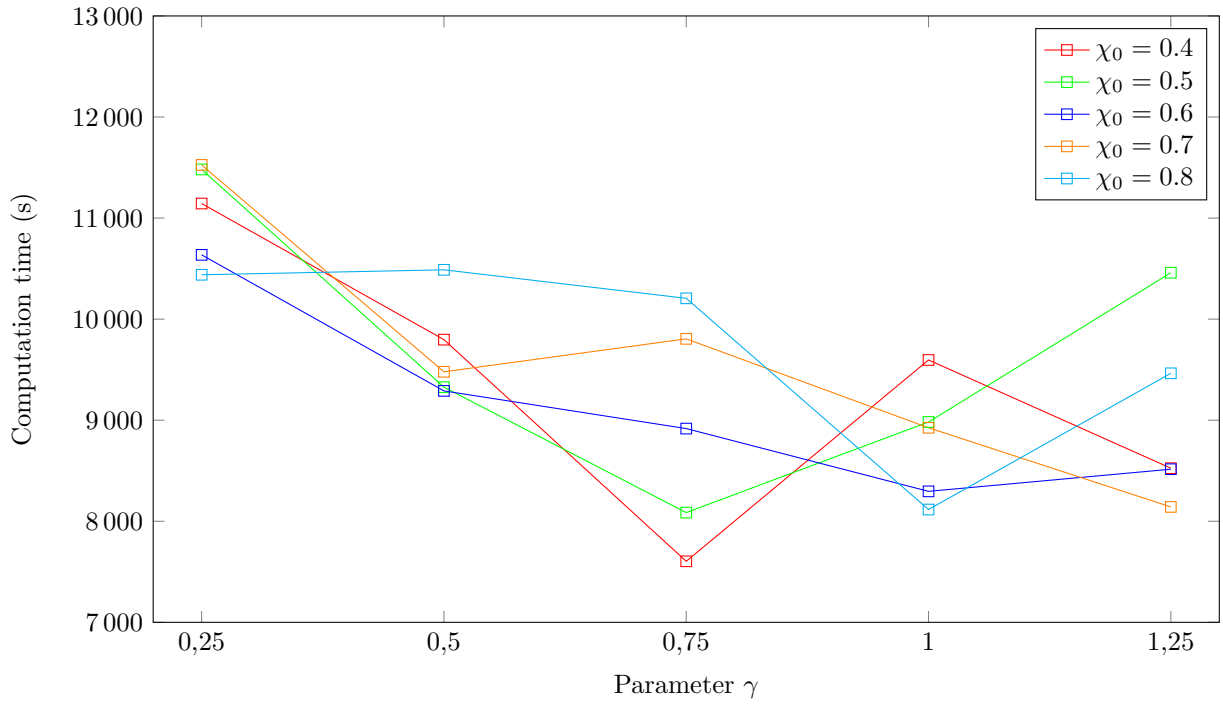
(b) Box plot comparison of the optimality gap as a function of the initial acceptance ratio

FIGURE 7.9: Box plot comparison of the optimality gaps for the first phase of the parameter optimisation experiment involving the soft constraint violation severity factor γ and the initial acceptance ratio χ_0 for the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion.

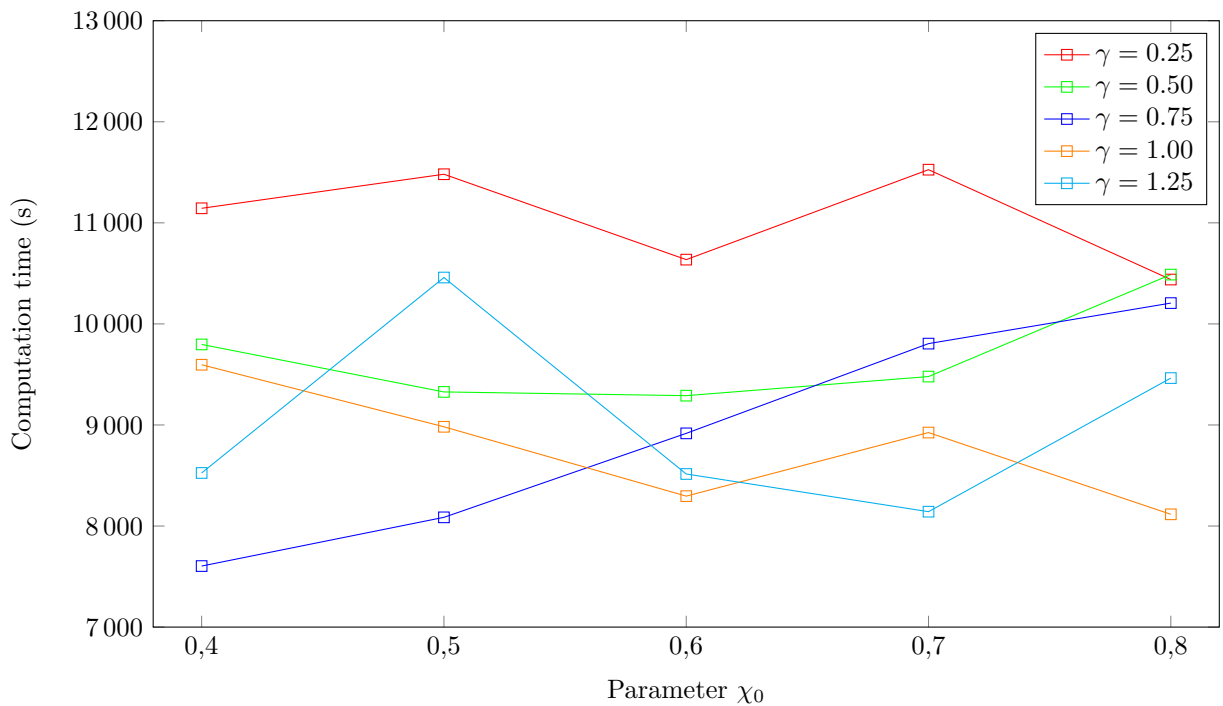
soft constraint violation severity factor for the different initial acceptance ratios. As for the initial acceptance ratio, no clear correlation is observed in the mean computation time when this parameter is varied, as may be seen in Figure 7.10(b).

It is observed in Table 7.8 that as the soft constraint violation severity factor increases, the number of infeasible incumbents decreases. A very large proportion of infeasible incumbents are observed for the small soft constraint violation severity factor value of 0.25. It is, however, observed that as the soft constraint violation severity factor increases, the number of infeasible incumbents returned (out of the 30) decreases to the extent that in some cases there are only one or two such infeasible solutions. This trend is also visible in Figure 7.11(a), where the number of infeasible incumbents returned is presented as a function of the soft constraint violation severity factor for different initial acceptance ratios. An almost exponential decay in the number of infeasible incumbents is observed as the soft constraint violation severity factor is decreased. For the initial acceptance ratio, however, it is observed, as previously also noted, that no significant change results in the number of infeasible incumbents returned as the value of the initial acceptance ratio varies. This may also be observed in Figure 7.11(b), where the number of infeasible incumbents is presented as a function of the initial acceptance ratio for different soft constraint violation severity factor values.

From the results obtained during the first phase of the parameter optimisation experiment, which included variation of the soft constraint violation severity factor and the initial acceptance ratio,

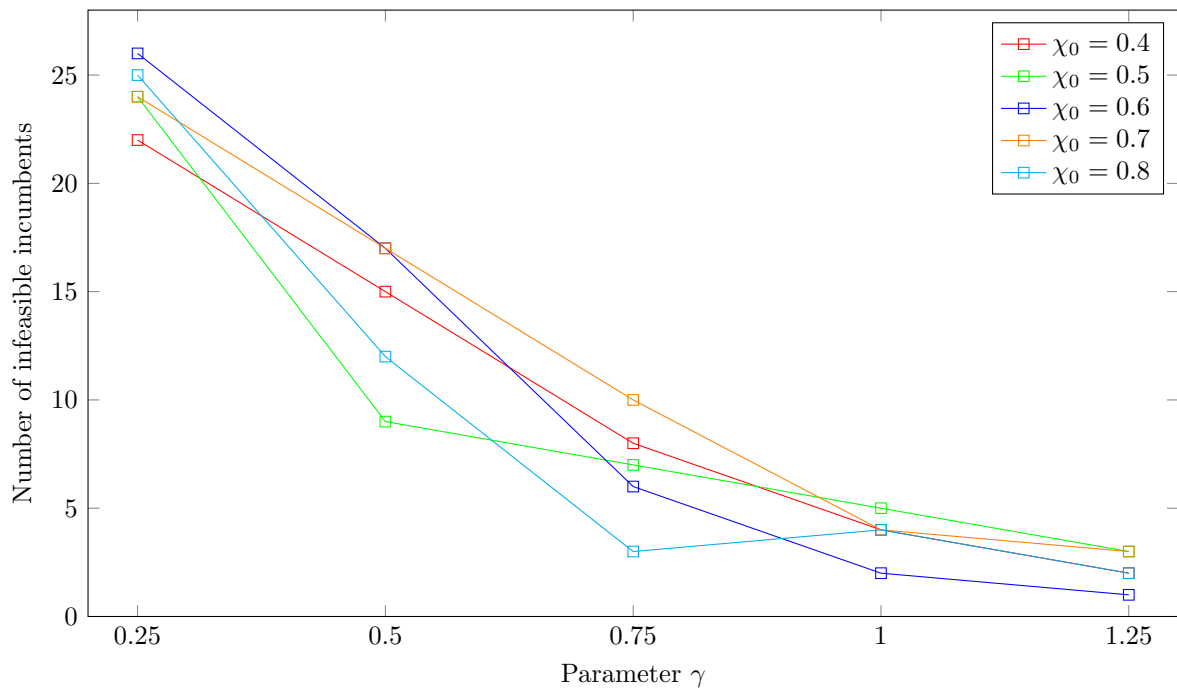


(a) The required computation time as a function of the soft constraint violation severity factor γ for different values of the initial acceptance ratio χ_0

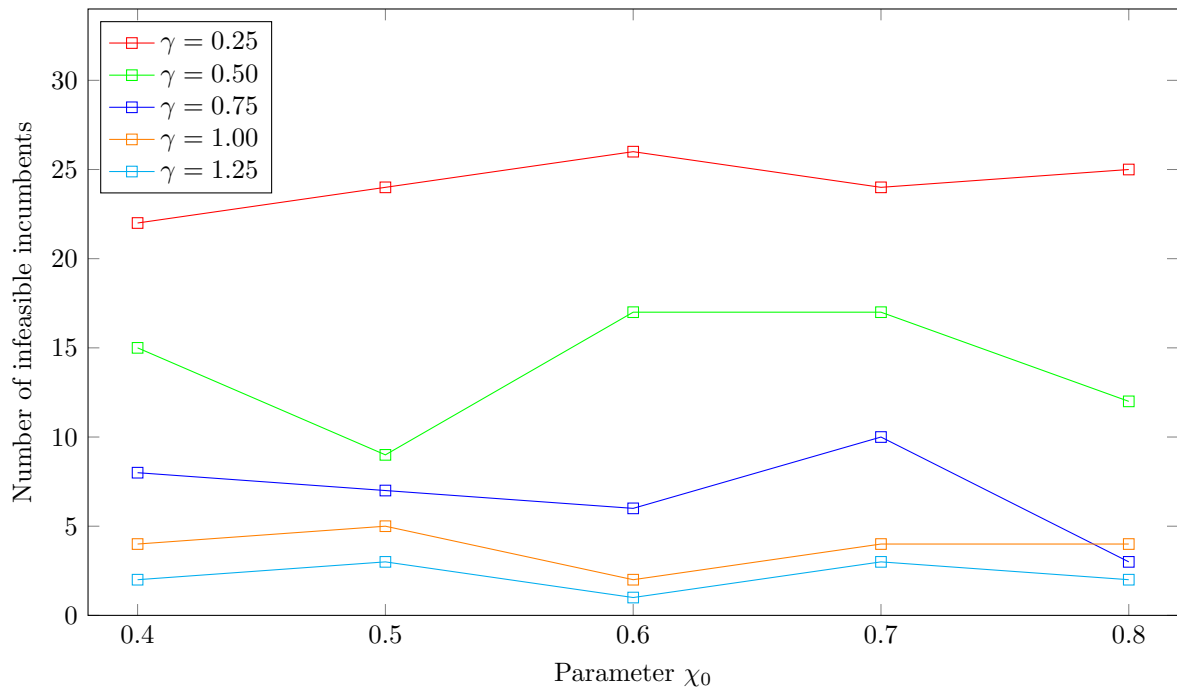


(b) The required computation time as a function of the initial acceptance ratio χ_0 for different values of the soft constraint violation severity factor γ

FIGURE 7.10: The required computation time as functions of the soft constraint violation severity factor γ and the initial acceptance ratio χ_0 when solving the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion.



(a) The number of infeasible incumbents returned (out of 30) as a function of the soft constraint violation severity factor γ for different values of the initial acceptance ratio χ_0



(b) The number of infeasible incumbents returned (out of 30) as a function of the initial acceptance ratio χ_0 for different values of the soft constraint violation severity factor γ

FIGURE 7.11: The number of infeasible incumbents returned (out of 30) as functions of the soft constraint violation severity factor γ and the initial acceptance ratio χ_0 when solving the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion.

TABLE 7.9: Selected parameter combination for the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ , as obtained from the first phase of the parameter optimisation experiment for the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion.

χ_0	γ	Optimality gap (%)	Computation time (s)	Infeasibilities
0.6	1	3.797	8 296	2 (out of 30)

a value of 1 is selected for the soft constraint violation severity factor and a value of 0.6 is selected for the initial acceptance ratio, as shown in Table 7.9. For this combination of parameter values, an acceptable mean optimality gap of 3.7966% is obtained. The mean computation time of 8 296 seconds required to solve the 21-unit test system is also acceptable as it is lower than the maximum amount of computation time allowed for the SA search, namely 14 400 seconds. A small proportion of infeasible incumbents is also returned (out of the 30 runs) by this parameter value combination. This combination is therefore adopted as the final parameter values in the context of solving the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion. These values for the soft constraint violation severity factor and the initial acceptance ratio are also adopted during the second phase of the parameter optimisation experiment involving variation of the cooling parameter, the reheating parameter and the epoch parameter values.

Phase 2: Cooling parameter, reheating parameter and epoch parameter

The mean optimality gaps (again measured as a percentage relative to the optimal objective function value obtained by CPLEX) for the feasible incumbents returned during the second phase of the parameter optimisation experiment are shown in Table 7.10, while the mean computation times required for the evaluation of the combinations of these parameter values are shown in Table 7.11. These times again include computation times expended during runs that returned infeasible incumbents. The numbers of times (out of 30) that an infeasible incumbent was returned during an SA search run are shown in Table 7.12. Furthermore, box plot comparisons of the mean optimality gaps are presented as functions of the cooling parameter, the reheating parameter and the epoch parameter in Figures 7.12(a), 7.12(b) and 7.12(c), respectively.

TABLE 7.10: Mean optimality gaps for all the combinations of parameter values during of the second phase of the parameter optimisation experiment involving variation of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ for the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion.

		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	3.1737	4.4756	5.8918
		2	3.7254	4.7713	6.7069
		4	3.6560	5.9728	6.7189
	0.9	1	3.1378	3.9413	5.9558
		2	3.8702	4.6027	6.0513
		4	3.2622	5.3155	6.9846
	0.95	1	4.8377	4.3726	4.4784
		2	4.6179	3.6225	5.6923
		4	4.3564	4.3830	6.7840

TABLE 7.11: Mean computation times required for all the combinations of parameter values during the second phase of the parameter optimisation experiment involving variation of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ for the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion.

		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	14400	8148	2220
		2	14243	4898	777
		4	12409	469	197
	0.9	1	14400	13148	4258
		2	14400	7176	3450
		4	13940	2343	352
	0.95	1	14400	14400	10827
		2	14400	14400	3204
		4	14400	10981	1020

TABLE 7.12: Number of infeasible incumbents (out of 30) returned for all the combinations of parameter values during the second phase of the parameter optimisation experiment involving variations of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ for the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion.

		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	2 (6.67%)	2 (6.67%)	3 (10.00%)
		2	2 (6.67%)	3 (10.00%)	14 (46.67%)
		4	3 (10.00%)	11 (36.67%)	15 (50.00%)
	0.9	1	4 (13.33%)	4 (13.33%)	3 (10.00%)
		2	2 (6.67%)	4 (13.33%)	9 (30.00%)
		4	2 (6.67%)	6 (20.00%)	17 (56.67%)
	0.95	1	7 (23.33%)	5 (16.67%)	6 (20.00%)
		2	4 (13.33%)	3 (10.00%)	8 (26.67%)
		4	6 (20.00%)	5 (16.67%)	13 (43.33%)

From these results, it may be observed that there is not much of a correlation between the cooling parameter and average optimality gap. As the cooling parameter increases from 0.85 to 0.95, no clear change is observed in the average optimality gap. This is observed in Table 7.10 — as one moves down the table, no clear change in the average optimality gaps are observed. This may also be seen in Figure 7.12(a) where it is observed that the optimality gaps remain relatively constant as a function of the cooling parameter. The mean optimality gaps corresponding to the cooling parameter values of 0.85, 0.90 and 0.95 are 5.602%, 5.209% and 5.096%, respectively, which is a small difference. From Figure 7.12(a) it is, however, observed that as the cooling parameter increases, the spreads of the optimality gaps decrease and fewer outliers are observed.

The correlation between the reheating parameter and the mean optimality gap is much more prominent than that between the mean optimality gap and the cooling parameter. As the reheating parameter increases in Table 7.10 (*i.e.* moving from left to right in the table), an increase in the mean optimality gap is observed. This is also observed in the box plot in Figure 7.12(b), which compares the mean optimality gaps for the three different reheating parameter values of 0.55, 0.75 and 0.95. Here a clear increase in the mean optimality gap is observed as the reheating parameter increases. The mean optimality gaps corresponding to the reheating parameter values of 0.55, 0.75 and 0.95 are 3.907%, 4.970% and 7.050%, respectively. It is also observed in

Figure 7.12(b) that as the reheating parameter increases, an undesirable increase results in the spreads and the number of outliers of the optimality gaps.

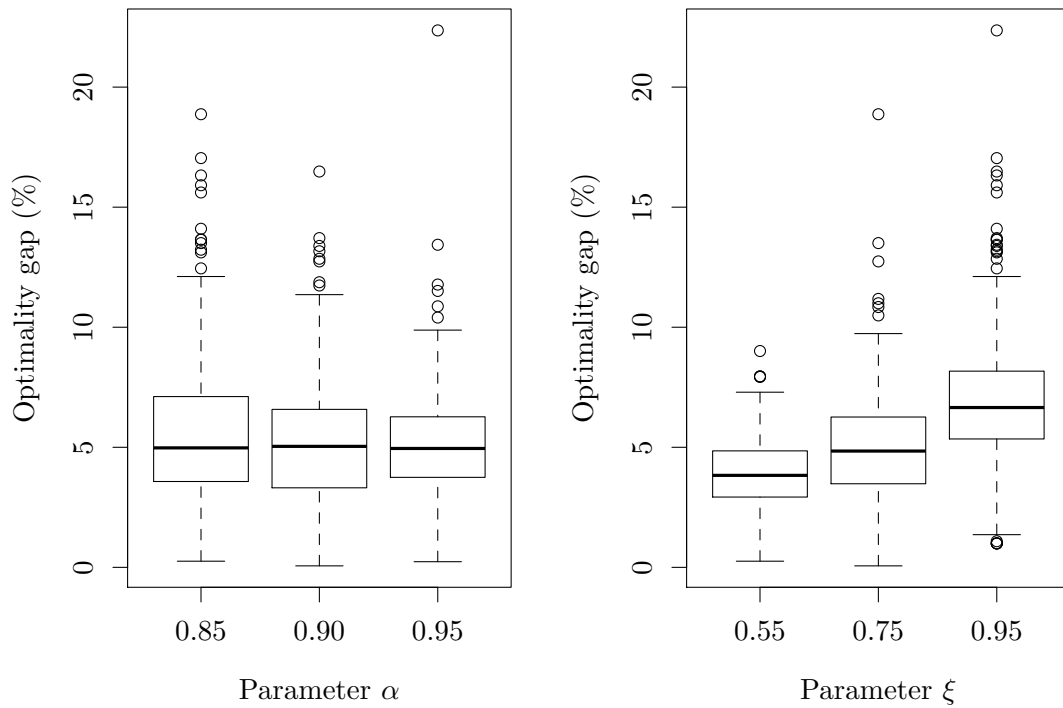
An analysis of the epoch parameter in terms of the mean optimality gap shows a gradual increase in the optimality gap as the epoch parameter increases. This is not so easily observed in Table 7.10, but it is clear in Figure 7.12(c). In this box plot, it may be seen that as the epoch parameter increases from 1 to 4, the optimality gap increases gradually. The mean optimality gaps for epoch parameter values of 1, 2 and 4 are 4.620%, 5.206% and 6.080%, respectively. It is also important to note that as the epoch parameter increases, so does the spread and the number of outliers of the optimality gaps. The effects of these three parameter combinations on the required computation time are reported in Table 7.11. The mean computation time required to solve the 21-unit test system for each combination of the three parameter values is presented in Figure 7.13. Here it may be observed that as the cooling parameter increases, an increase in the required computation time results. The mean computation time required for the cooling parameter values of 0.85, 0.90 and 0.95 are 6418 seconds, 8163 seconds and 10893 seconds, respectively. This increase in computation time may be the result of a larger cooling parameter providing a slower decay in the temperature of the SA algorithm which may, in turn, cause the algorithm to terminate after a longer time.

In Table 7.11 and Figure 7.13, the required computation time is observed to decrease as the reheating parameter increases. It is clear, as one moves from right to left in the table, that the computation time decreases. The average computation time required for evaluation of the reheating parameter values of 0.55, 0.75 and 0.95 are 14110 seconds, 8440 seconds and 2923 seconds, respectively, which represents a very large variation. One reason for observing very short computation times for large reheating parameter values may be that the increase in temperature of the SA algorithm caused by a large reheating parameter is very small and so the algorithm may terminate easier due to the three-consecutive-reheating termination criterion.

The effect that the epoch parameter has on the computation time required by the SA algorithm to solve the 21-unit test system may also be observed in Table 7.11 and Figure 7.13. As the epoch parameter value increases, the required computation time decreases. The mean computation time required for epoch parameter values of 1, 2 and 4 are 10689 seconds, 8550 seconds and 6235 seconds, respectively, which again represents a large variation. A reason for this observation may be that a larger epoch parameter value causes a shorter epoch on average which, in turn, results in a decrease in the number of iterations required before cooling or reheating is performed. This may cause the entire SA algorithm to terminate faster, because the temperature will decrease very rapidly and three consecutive reheats will be achieved early on during the algorithmic execution.

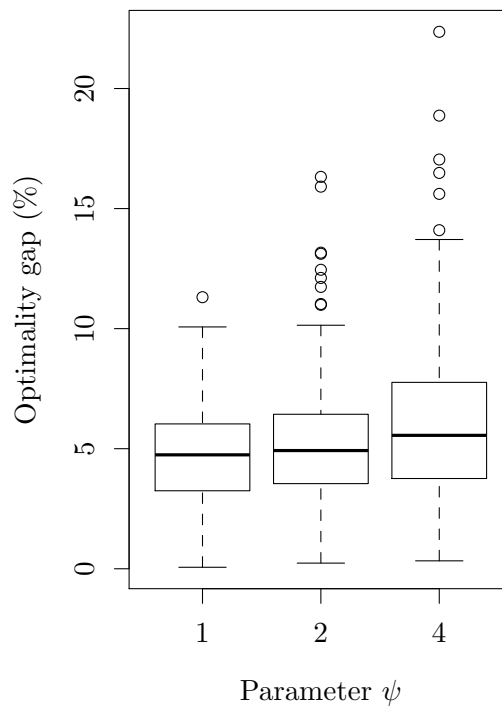
Finally, the number of infeasible incumbents returned (out of 30) during execution of the SA algorithm is reported in Table 7.12. The mean number of infeasible incumbents returned during each search run is presented in Figure 7.14. Here no significant change in the number of infeasible incumbents returned is observed when varying the cooling parameter. The mean number of infeasible incumbents returned (out of 30) for cooling parameter values of 0.85, 0.90 and 0.95 are 6.111, 5.667 and 6.333, respectively. In terms of the reheating parameter and epoch parameter, an almost exponential increase in the number of infeasible incumbents returned is observed as these parameter values increase. The average number of infeasible incumbents returned (out of 30) for reheating parameter values of 0.55, 0.75 and 0.95 are 3.556, 4.778 and 9.778, respectively, whereas the average number of infeasible incumbents returned (out of 30) for the epoch parameter values of 1, 2 and 4 are 4, 5.444 and 8.667, respectively.

The aim of the parameter optimisation experiment in this section was to obtain a suitable combination of parameters which may be used in the approximate solution approach (*i.e.* the



(a) Box plot comparison of the optimality gap as a function of the cooling parameter

(b) Box plot comparison of the optimality gap as a function of the reheating parameter



(c) Box plot comparison of the optimality gap as a function of the epoch parameter

FIGURE 7.12: Box plot comparison of the optimality gaps obtained during the second phase of the parameter optimisation experiment involving variation of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ for the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion.

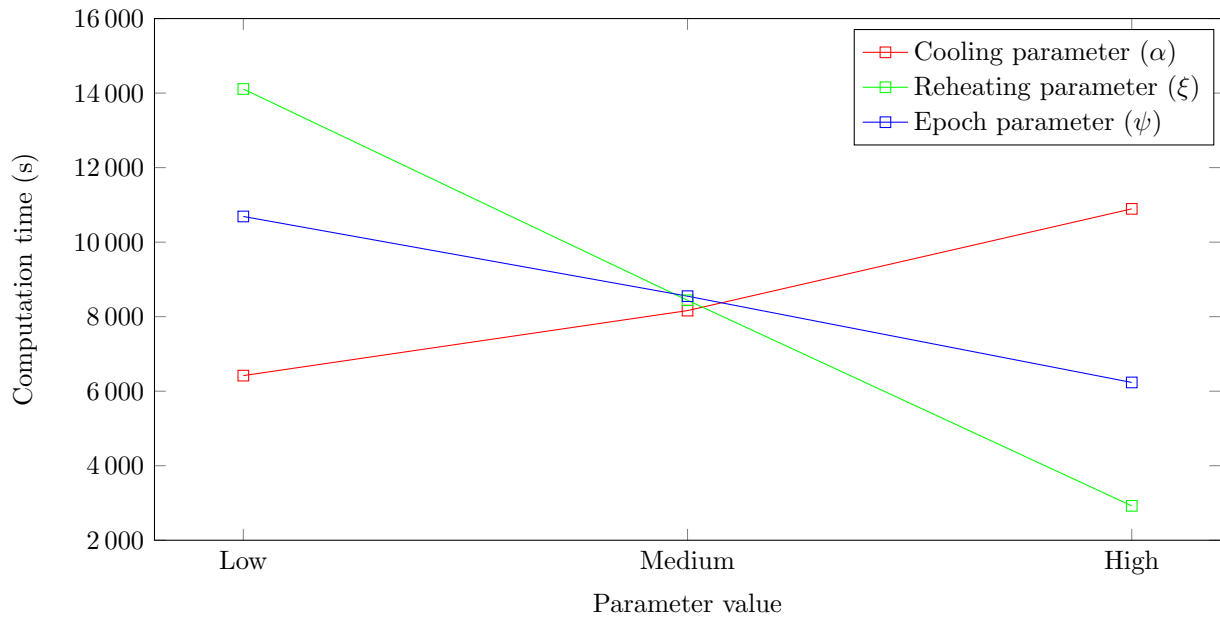


FIGURE 7.13: The computation time required to solve the 21-unit test system as a function of the cooling parameter, the reheating parameter and the epoch parameter with minimisation of the probability of unit failure as scheduling criterion.

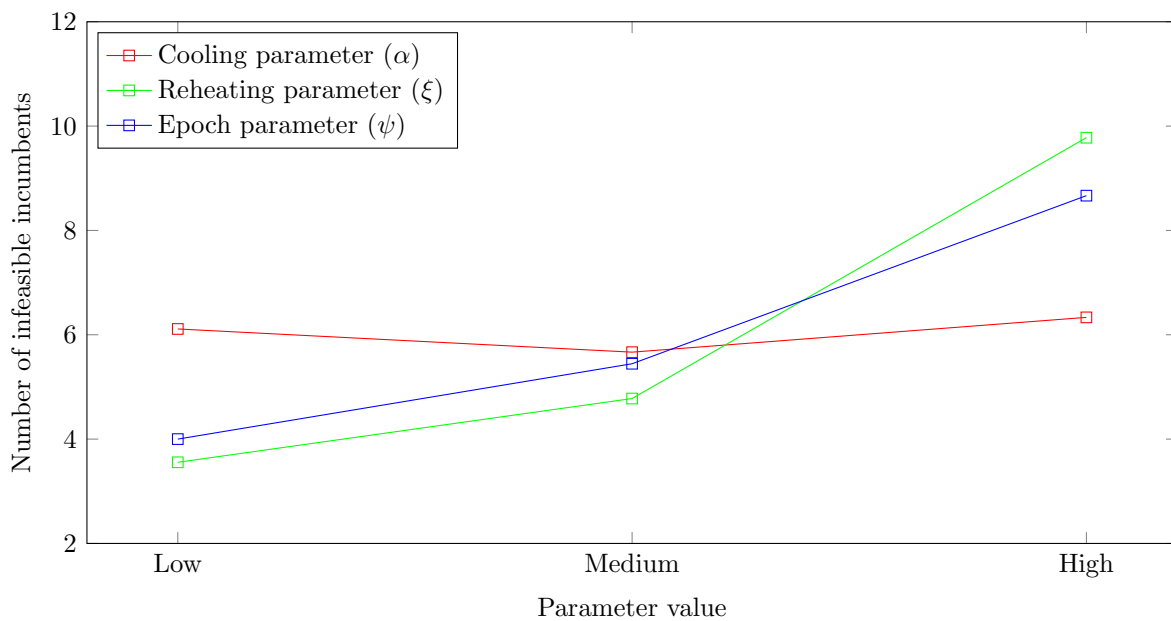


FIGURE 7.14: The mean number of infeasible incumbents (out of 30) as a function of the cooling parameter, the reheating parameter and the epoch parameter for the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion.

method of SA) so as to obtain good GMS solutions for the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion. In Figure 7.12(b), it may be observed that the large value for the reheating parameter achieves the largest mean optimality gap and that many outliers are observed in this case. This parameter value also produced the largest mean number of infeasible incumbents. The reheating parameter value of 0.95 was therefore eliminated from the parameter values considered as candidates for the final set of parameters in the context of the 21-unit test system. It was furthermore observed that a reheating parameter of 0.55 results in the longest required computation time (an average of 14 110 seconds). For this reason, this parameter value was also eliminated from the parameter values considered as candidates for the final set of parameters for the 21-unit test system. The only reheating parameter value left, is therefore 0.75, which achieves an acceptable mean optimality gap value as well as an acceptable mean computation time required by the SA algorithm. This parameter value also returns, on average, a small number of infeasible incumbents.

Furthermore, the cooling parameter values of 0.85 and 0.95 were also eliminated from the candidates considered due to both of these parameter values returning, on average, more than six infeasible incumbents out of thirty. Hence a cooling parameter value of 0.9 was selected as the best candidate for the 21-unit test system. Finally, in terms of the results obtained for the epoch parameter in combination with the value of 0.90 for the cooling parameter and 0.75 for the reheating parameter, it was observed that the larger value (*i.e.* 4) returned six infeasible incumbents out of thirty, which is an undesirably large proportion. Of the remaining epoch parameter values, the value 1 was selected as the final epoch parameter as this value results in a smaller mean optimality gap as well as returning a smaller number of infeasible incumbents of four out of thirty. The final set of parameters adopted in the SA algorithm for solving the 21-unit test system is shown in Table 7.13.

TABLE 7.13: *The complete set of parameters selected for the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion, as well as the mean optimality gap, the required computation time and the number of infeasible incumbents associated with these parameter values during the second phase of the parameter optimisation experiment.*

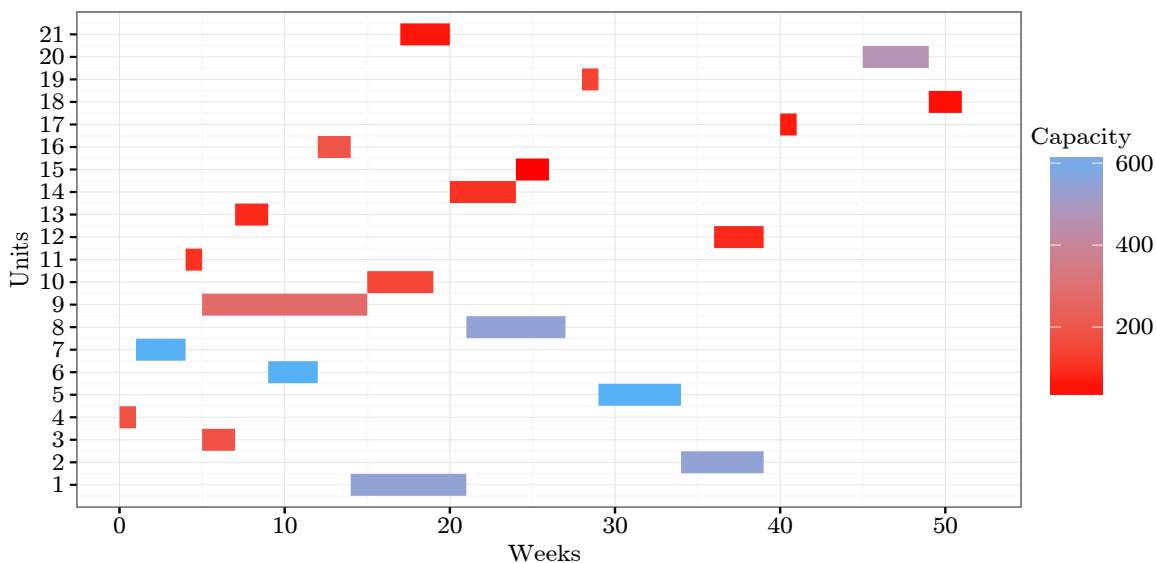
χ_0	γ	α	ξ	ψ	Optimality gap (%)	Computation time (s)	Infeasibilities
0.6	1	0.9	0.75	1	3.941	13 148	4 (out of 30)

Numerical results

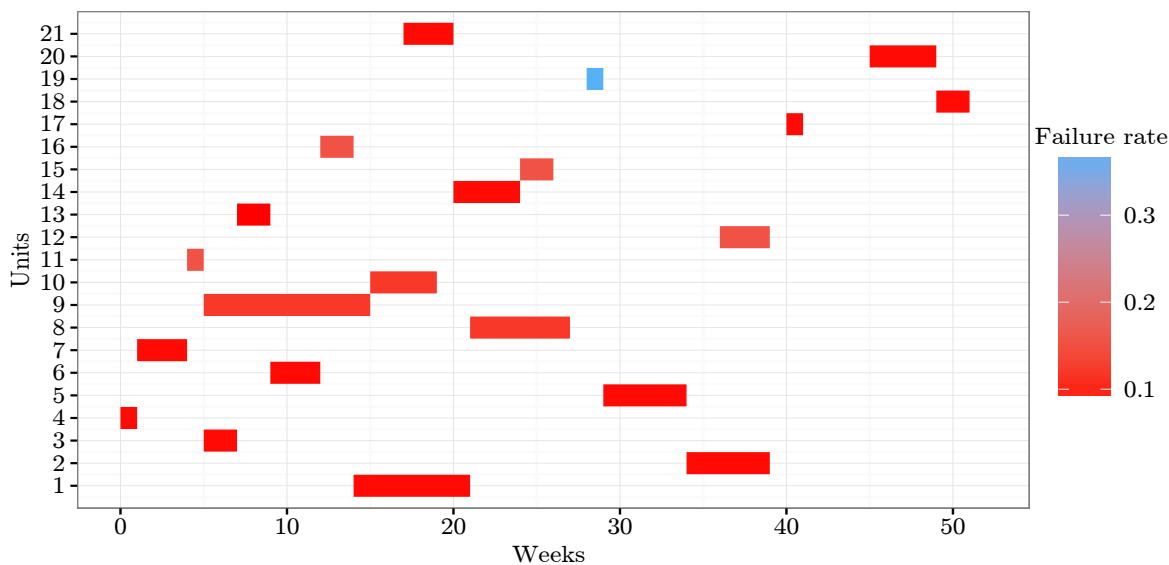
An approximate solution to the linear GMS model of §4.3 was obtained for the 21-unit test system by the SA algorithm with the parameter combinations as specified in Table 7.13. The decision variable values of the incumbent are given in integer decision vector form by $\mathbf{x} = [15, 35, 6, 1, 30, 10, 2, 22, 6, 16, 5, 37, 8, 21, 25, 13, 41, 50, 29, 46, 18]$, which corresponds to an objective function value of 43.572 (0.0689% worse than that of the optimal solution for the 21-unit test system, as reported in §7.1.1).

A graphical representation of this maintenance schedule is presented in Figure 7.15 with the colour scale in Figure 7.15(a) indicating the capacity (in MW) of each PGU and the colour scale in Figure 7.15(b) indicating the failure rate of each PGU. The manpower required over the duration of the scheduling window in order to implement the solution in Figure 7.15 is shown in Figure 7.16(a), while the available system capacity over the duration of the scheduling window associated with this solution is shown in Figure 7.16(b).

The solution may be analysed by comparing it with the optimal solution obtained in §7.1.1 for the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion. A graphical representation of the incumbent solution returned by the approximate



(a) The incumbent returned by the SA algorithm for the linear model of §4.3, with the colour scale indicating the rated capacity of the PGUs

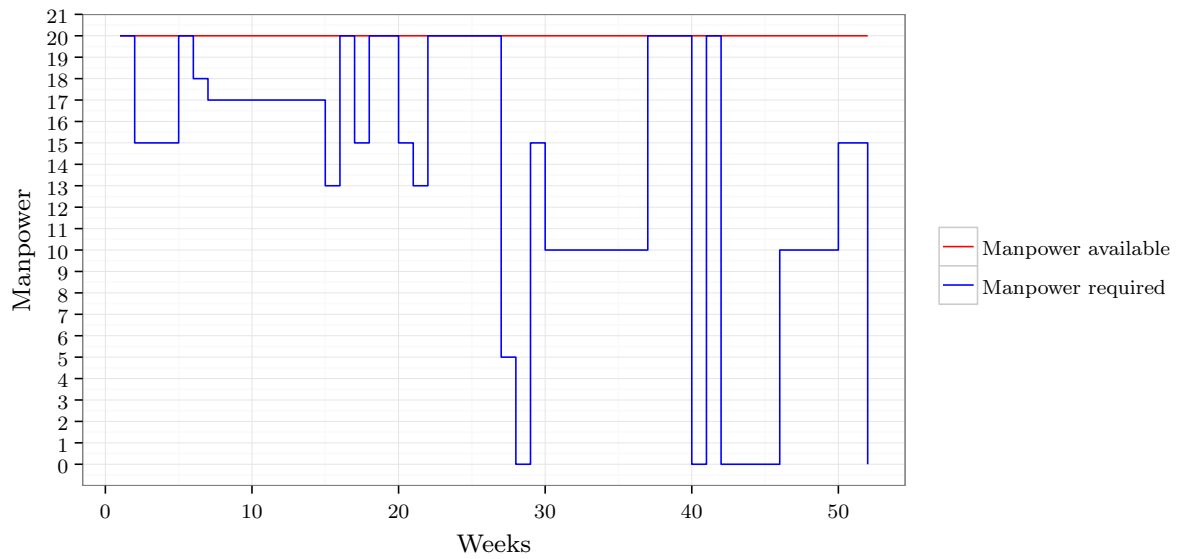


(b) The incumbent returned by the SA algorithm for the linear model of §4.3, with the colour scale indicating the failure rate of the PGUs

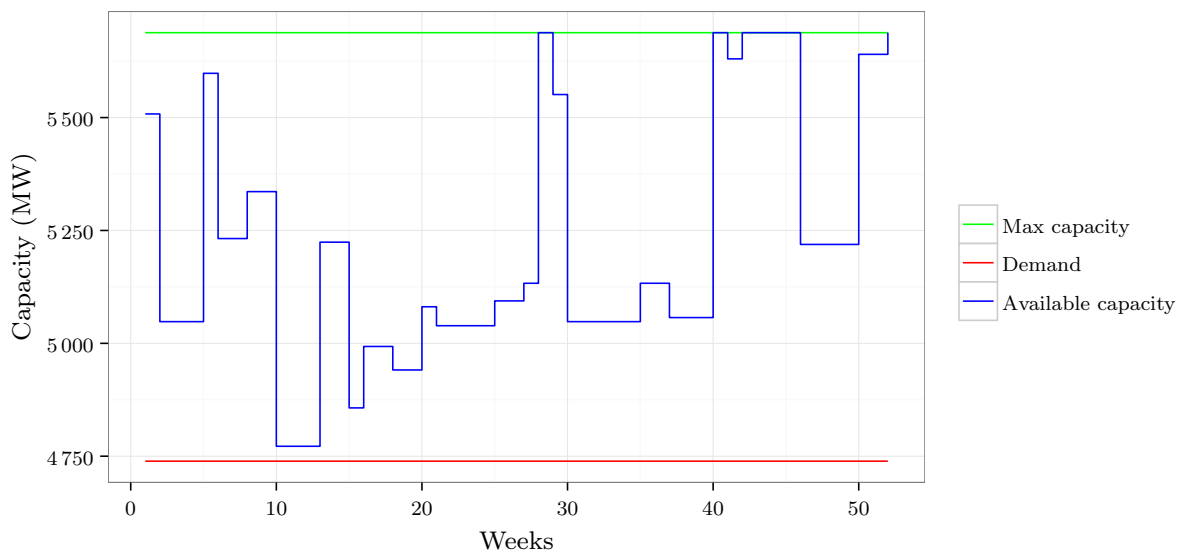
FIGURE 7.15: The incumbent returned by the SA algorithm for the linear model of §4.3, in the context of the 21-unit test system for parameter values as indicated in Table 7.13 with minimisation of the probability of unit failure as scheduling criterion.

solution methodology of §5.3 (*i.e.* the method of SA) and the optimal solution obtained by the exact solution approach described in §7.1 (*i.e.* by means of CPLEX) is shown in Figure 7.17 for the 21-unit test system. The maintenance schedules of the two solutions are compared in the figure, with the colour scale indicating the rated capacity of the PGUs in the system. The effects of the two maintenance schedules on the manpower required and the available system capacity are shown in Figures 7.18(a) and 7.18(b), respectively.

Comparing these two maintenance schedules, a reasonable amount of differences are observed in the scheduled commencement times of the PGUs, although the objective function values only



(a) The manpower required over the scheduling window corresponding to the maintenance schedule in Figure 7.15



(b) The system capacity over the scheduling window corresponding to the maintenance schedule in Figure 7.15

FIGURE 7.16: Evaluation of the manpower required and system capacity available over the duration of the scheduling window for the 21-unit test maintenance schedule in Figure 7.15.

differ by 0.0689%. Comparing the available system capacity in Figure 7.18(b), it is seen for the solution obtained by the exact solution approach that the system capacity is very close to the demand during the early stages of the maintenance window. This is due to the proposed objective function giving preference to maintenance of PGUs with large rated capacities, resulting in these PGUs being scheduled for maintenance early during the scheduling window and the available system capacity being low. This is not, however, the case for the approximate solution approach. It is observed that the system capacity associated with the approximate maintenance schedule only drops drastically during planning period 10, when the difference between the demand and available system capacity is very small. Figure 7.18(b) also reveals that in

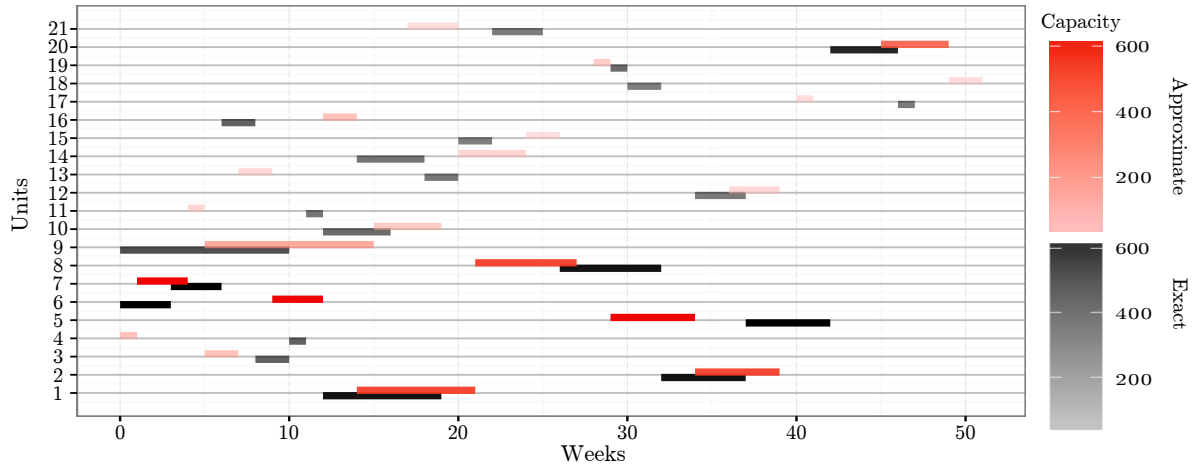


FIGURE 7.17: Two maintenance schedules for the 21-unit test system returned by two different approaches towards solving the linear GMS model of §4.3.

the approximate solution, the manpower constraints are binding during a number of planning periods early during the scheduling window. The manpower required for implementation of the approximate solution approach is zero during time period 40 and again during time period 42 onwards as no maintenance is scheduled during these time periods. This causes maintenance of all the PGUs in the system only to be completed during planning period 52 according to the approximate solution approach, whereas maintenance on all the PGUs are already completed after planning period 48 according to the optimal solution.

7.2.2 The IEEE-RTS

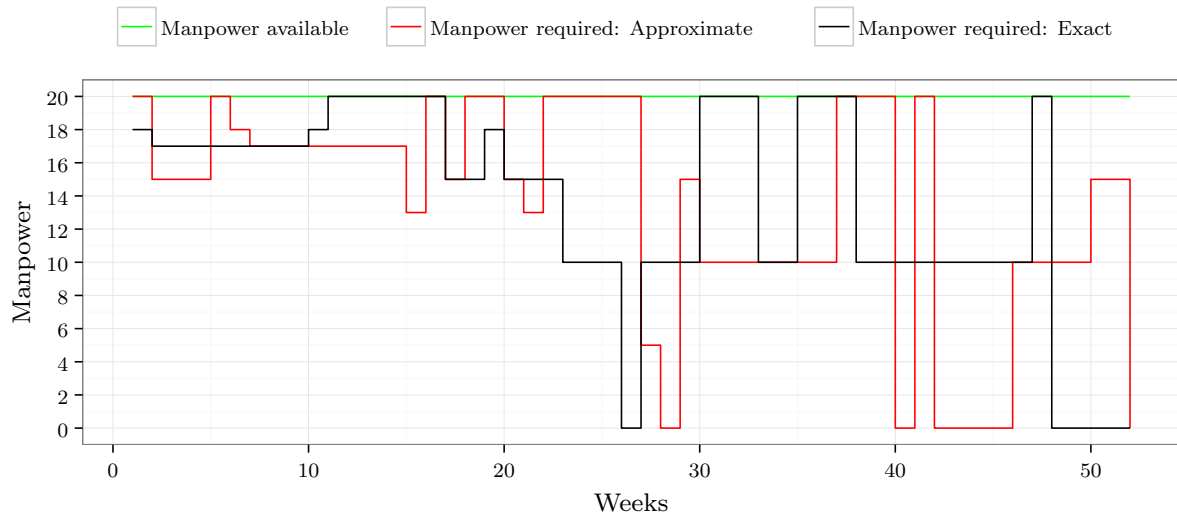
This section is devoted to an application of the parameter optimisation experiment described in §5.3.3, this time within the context of the IEEE-RTS. The best algorithmic parameter combination thus uncovered is then employed to solve the test instance approximately.

Parameter optimisation experiment

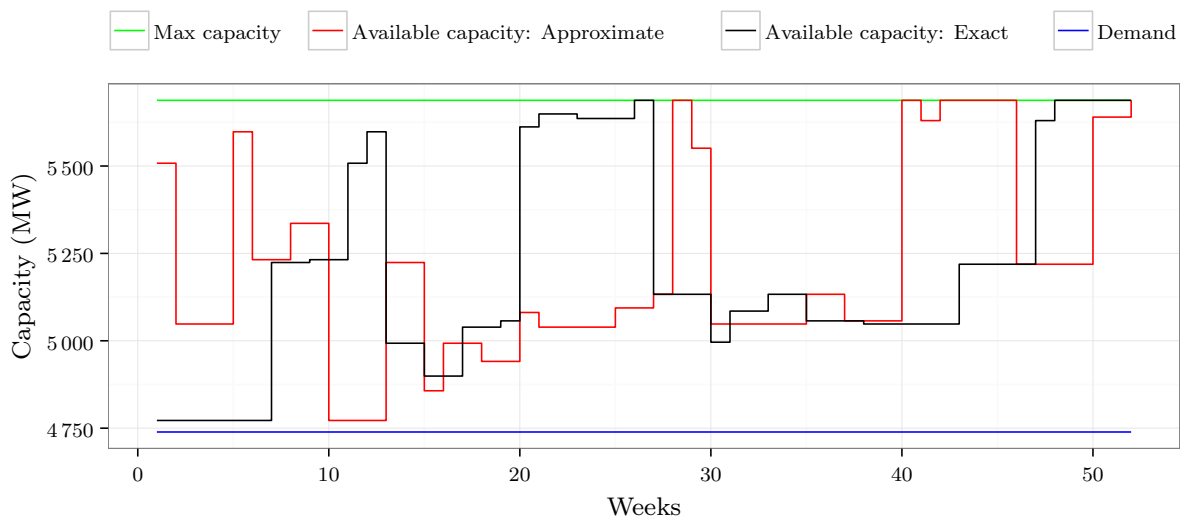
In this section, a parameter optimisation experiment is again performed for the IEEE-RTS in two separate phases according to the design described in §5.3.3. The first phase of the experiment involves the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ . During the second phase of the experiment the parameters considered are the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ .

Phase 1: Initial acceptance ratio and soft constraint violation severity factor

The mean optimality gap percentages (relative to the optimal objective function value obtained by CPLEX) for the feasible incumbents returned during the first phase of the parameter optimisation experiment are shown in Table 7.14. The mean computation time for each of the combinations of these parameters is shown in Table 7.15, which includes the computation time expended during runs that returned infeasible incumbents. The number of times (out of 30) that an infeasible incumbent was returned are shown in Table 7.16.



(a) The required manpower over the scheduling window corresponding to the maintenance schedules in Figure 7.17



(b) The system capacity over the scheduling window corresponding to the maintenance schedules in Figure 7.17

FIGURE 7.18: Comparison of the manpower required and the available system capacity associated with the schedules in Figure 7.17 for the 21-unit test system with minimisation of the probability of unit failure as scheduling criterion.

In these results a slight correlation is observed between the mean optimality gap and the soft constraint violation severity factor. As the severity factor increases, it is observed that the mean optimality gap also increases. In Figure 7.19(a), however, where a box plot comparison is presented of the mean optimality gap as a function of the soft constraint violation severity factor, it is observed that the mean optimality gaps associated with the soft constraint violation severity factor values of 1.00 and 1.25 are both lower than that associated with the value 0.75. It should, however, be noted that the difference between the mean optimality gap of the worst performing value (*i.e.* 0.75) and best performing value (*i.e.* 0.0.25) of the soft constraint violation severity

TABLE 7.14: Mean optimality gaps for all parameter combinations of the first phase of the parameter optimisation experiment involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ for the IEEE-RTS with minimisation of the probability of unit failure as scheduling criterion.

χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	2.115	2.507	2.616	2.672	2.756
0.5	2.170	2.572	2.604	2.228	2.781
0.6	2.144	2.337	2.666	2.702	2.714
0.7	2.332	2.585	2.800	2.535	2.523
0.8	2.037	2.475	2.513	2.627	2.599

TABLE 7.15: Mean computation times required for all parameter combinations of the first phase of the parameter optimisation experiment involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ for the IEEE-RTS with minimisation of the probability of unit failure as scheduling criterion.

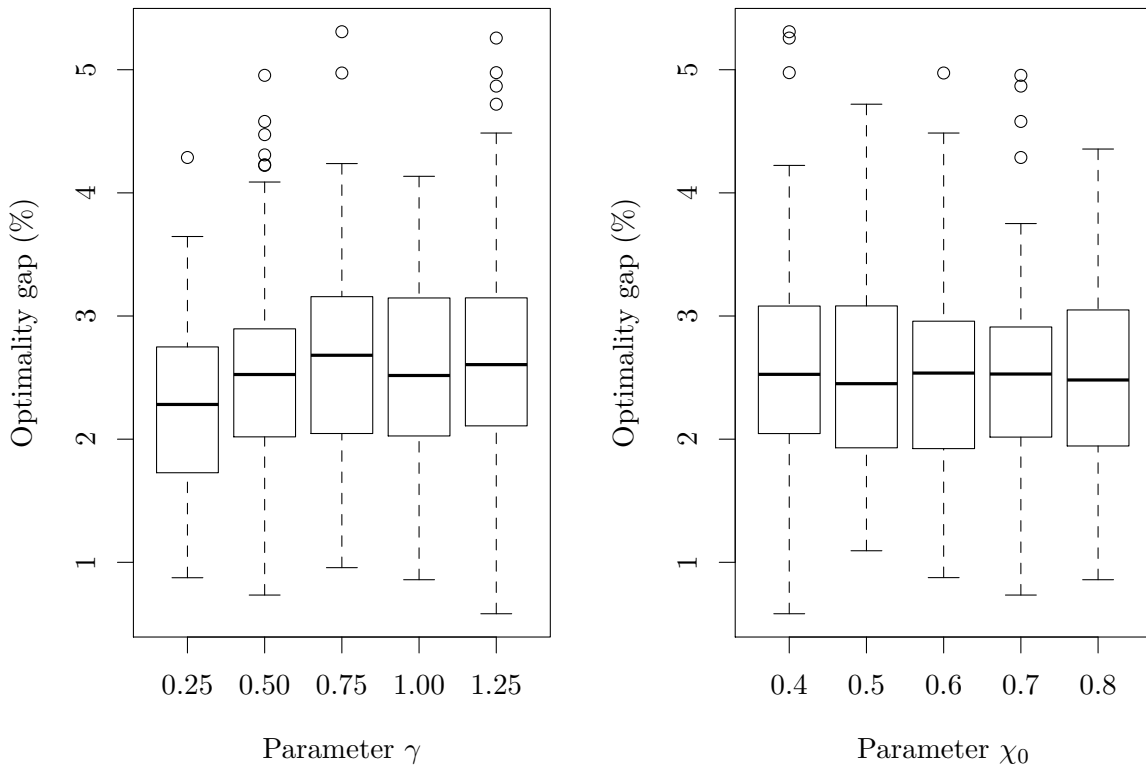
χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	26022	28074	27169	28800	28275
0.5	28595	28375	28261	28800	28472
0.6	27817	28795	26376	28419	28800
0.7	27264	28800	26671	28552	28776
0.8	27149	27668	28184	28134	28800

factor is only 0.331%. As for the 21-unit test system, the reason for this slight correlation may be that the search space contains disjoint pockets of feasible regions, but not as many as in the 21-unit test system. Therefore, in order to move from one such pocket of feasibility to a different pocket of feasibility, infeasible solutions have to be accepted during the transition in view of the fact that the SA algorithm is a trajectory-based metaheuristic. This type of transition is, however, less probable when adopting a larger soft constraint violation severity factor. It is therefore observed that the smallest value of the soft constraint violation severity factor leads to the discovery of feasible incumbents during the SA search that are close to optimal.

The effect of the initial acceptance ratio on the mean optimality gap seems to be less prominent. Almost no increase or decrease is observed in the mean optimality gap as the initial acceptance ratio is increased. This is clearly visible in the box plot comparison presented in Figure 7.19(b). It is observed that the spreads of the mean optimality gaps for all five of the initial acceptance ratio values are fairly similar, with the initial acceptance values of 0.4 and 0.7 exhibiting a

TABLE 7.16: Number of infeasible incumbents (out of 30) returned for all parameter combinations of the first phase of the parameter optimisation experiment involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ for the IEEE-RTS with minimisation of the probability of unit failure as scheduling criterion.

χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	13 (43.33%)	2 (6.67%)	3 (10.00%)	0 (0.00%)	0 (0.00%)
0.5	13 (43.33%)	8 (26.67%)	0 (0.00%)	1 (3.33%)	0 (0.00%)
0.6	9 (30.00%)	6 (20.00%)	5 (16.67%)	1 (3.33%)	0 (0.00%)
0.7	12 (40.00%)	8 (26.67%)	0 (0.00%)	2 (6.67%)	0 (0.00%)
0.8	11 (36.67%)	2 (6.67%)	4 (13.33%)	2 (6.67%)	1 (3.33%)



(a) Box plot comparison of the optimality gap as a function of the soft constraint violation severity factor

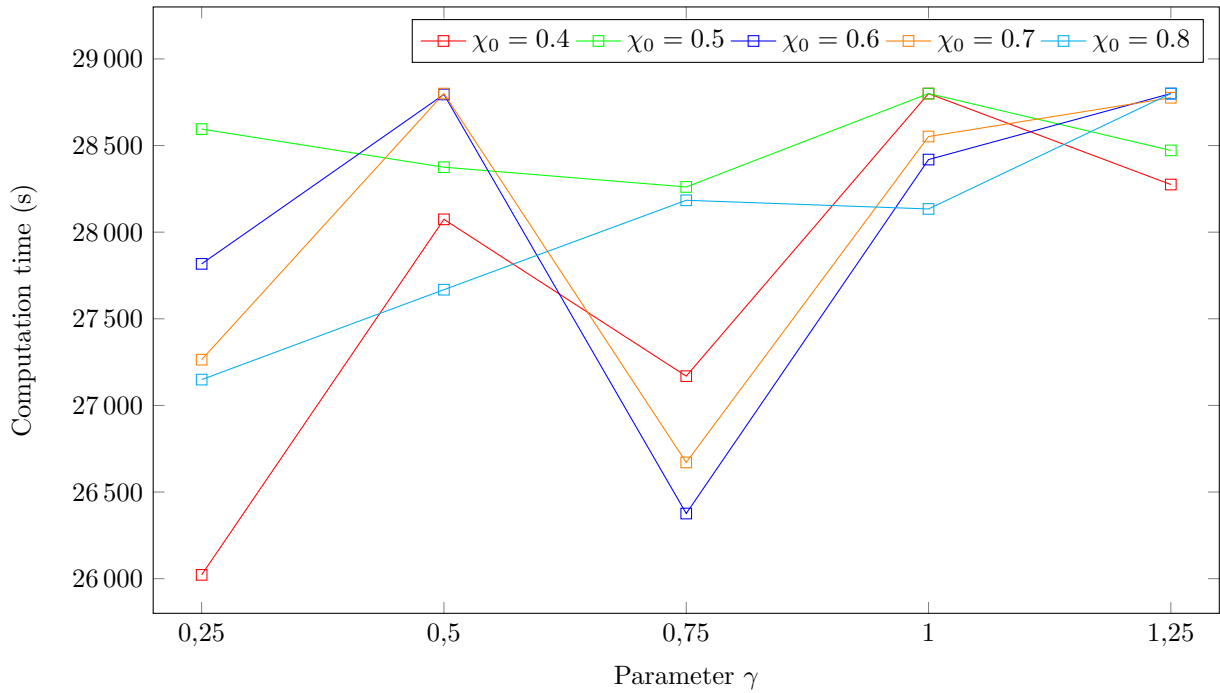
(b) Box plot comparison of the optimality gap as a function of the initial acceptance ratio

FIGURE 7.19: Box plot comparison of the optimality gaps for the first phase of the parameter optimisation experiment involving the soft constraint violation severity factor γ and the initial acceptance ratio χ_0 for the IEEE-RTS with minimisation of the probability of unit failure as scheduling criterion.

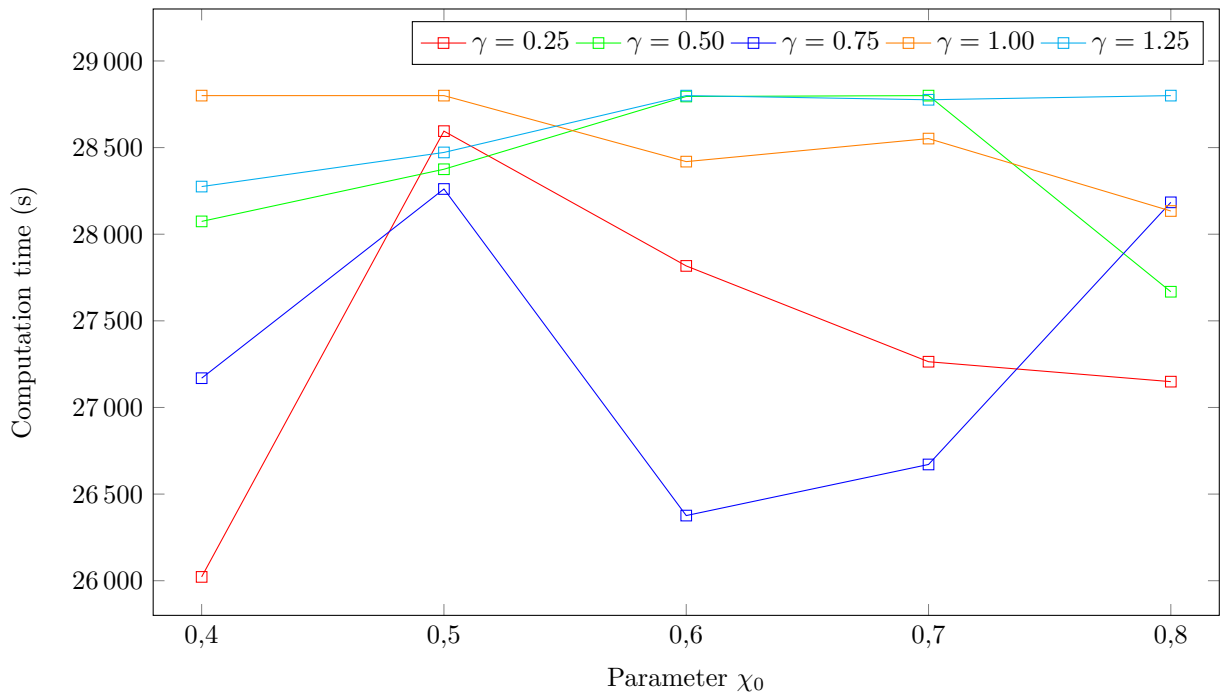
number of outliers. The results obtained in respect of the initial acceptance ratio are similar. This similarity in mean optimality gaps may be due to the fact that the SA algorithm typically terminates as a result of the three-consecutive-epoch termination criterion and not the maximum allowable computation time. The algorithm therefore has enough time to reach an acceptably good incumbent solution, even when the initial temperature is higher in some cases (as is the case for a higher initial acceptance ratio).

In terms of the computation time expended to evaluate each of the parameter combinations presented in Table 7.15, some level of correlation is noticeable between the soft constraint violation severity factor and the mean required computation time. As the soft constraint violation severity factor increases, a slight increase in the average computation time is observed for all values of the initial acceptance ratio, except for 0.5, in which case no clear correlation is observed. This slight upward trend is visible in Figure 7.20(a), which contains a plot of the computation time as a function of the soft constraint violation severity factor for the different initial acceptance ratios. It is also observed that for most values of the initial acceptance ratio, a decrease in computation time is observed for the soft constraint violation severity factor value of 0.75. As for the initial acceptance ratio, no clear correlation is observed in the mean computation time when this parameter is varied, as may be seen in Figure 7.20(b).

The numbers of infeasible incumbents returned (out of the 30 test runs) for each of the 25



(a) The required computation time as a function of the soft constraint violation severity factor for different values of the initial acceptance ratio χ_0



(b) The required computation time as a function of the initial acceptance ratio for different values of the soft constraint violation severity factor γ

FIGURE 7.20: The required computation time as a function of the soft constraint violation severity factor γ and the initial acceptance ratio χ_0 for the IEEE-RTS with minimisation of the probability of unit failure as scheduling criterion.

combinations of the soft constraint violation severity factor and the initial acceptance ratio are shown in Table 7.16. It is observed in this table that as the soft constraint violation severity factor increases, the number of infeasible incumbents decreases. A very large proportion of infeasible incumbents are observed for the small soft constraint violation severity factor value of 0.25. It is, however, observed that as the soft constraint violation severity factor increases, the number of infeasible incumbents returned (out of the 30) decreases to the extent that in some cases there are no such infeasible solutions. This trend is also clearly visible in Figure 7.21(a), where the number of infeasible incumbents returned is presented as a function of the soft constraint violation severity factor for different initial acceptance ratios. An almost exponential decay in the number of infeasible incumbents is observed as the soft constraint violation severity factor is decreased. For the initial acceptance ratio, however, it is observed, as previously also noted, that no significant change results in the number of infeasible incumbents returned as the value of the initial acceptance ratio varies. This may also be observed in Figure 7.21(b), where the number of infeasible incumbents is presented as a function of the initial acceptance ratio for different soft constraint violation severity factor values.

TABLE 7.17: Selected parameter combination for the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ , as obtained from the first phase of the parameter optimisation experiment for the IEEE-RTS with minimisation of the probability of unit failure as scheduling criterion.

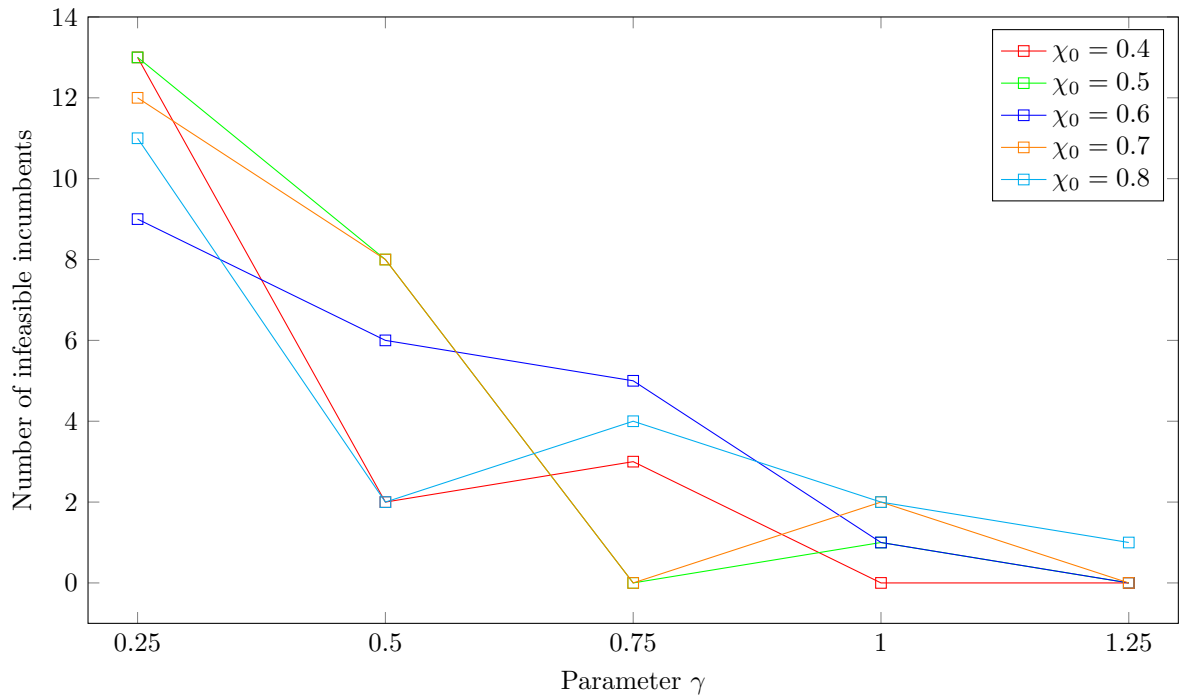
χ_0	γ	Optimality gap (%)	Computation time (s)	Infeasibilities
0.5	0.75	2.604	28 261	0

From the results obtained during the first phase of the parameter optimisation experiment, which included variation of the soft constraint violation severity factor and the initial acceptance ratio, a value of 0.75 is selected for the soft constraint violation severity factor and a value of 0.5 is selected for the initial acceptance ratio, as shown in Table 7.17. For this combination of parameter values, an acceptable mean optimality gap of 2.604% is obtained. The mean computation time of 28 261 seconds required to solve the IEEE-RTS is also acceptable as it is lower than the maximum amount of computation time allowed for the SA search, namely 28 800 seconds. No infeasible incumbents are also returned (in any of the 30 runs) for this parameter value combination. This combination is therefore adopted as the final parameter values in the context of solving the IEEE-RTS. These values for the soft constraint violation severity factor and the initial acceptance ratio are also adopted during the second phase of the parameter optimisation experiment involving variation of the cooling parameter, the reheating parameter and the epoch parameter values.

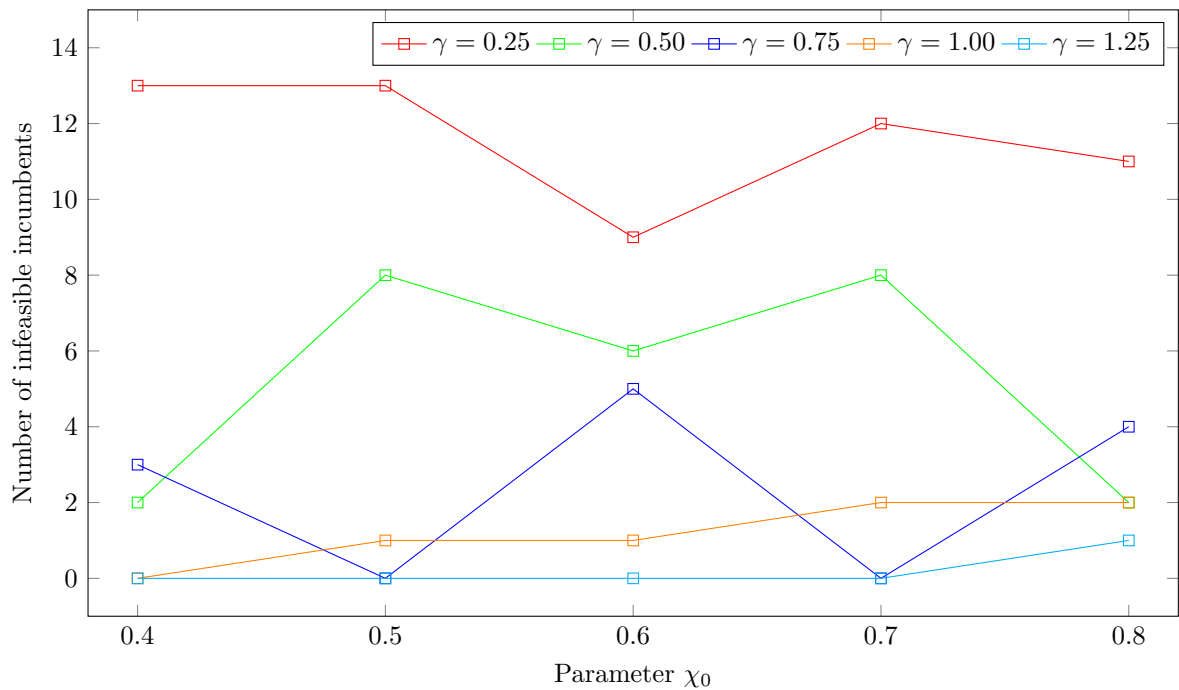
Phase 2: Cooling parameter, reheating parameter and epoch parameter

The mean optimality gaps (again measured as percentages relative to the optimal objective function values obtained by CPLEX) for the feasible incumbents returned during the second phase of the parameter optimisation experiment, are shown in Table 7.18, while the mean computation times required for the evaluation of the combinations of these parameter values are shown in Table 7.19. These times again include computation times expended during runs that returned infeasible incumbents. The numbers of times (out of 30) that an infeasible incumbent was returned during an SA search run are shown in Table 7.20. Furthermore, box plot comparisons of the mean optimality gaps are presented as functions of the cooling parameter, the reheating parameter and the epoch parameter in Figures 7.22(a), 7.22(b) and 7.22(c), respectively.

From these results, it may be observed that there is not much of a correlation between the cooling parameter and average optimality gap. As the cooling parameter increases from 0.85 to



(a) The number of infeasible incumbents returned (out of 30) as a function of the soft constraint violation severity factor for different values of the initial acceptance ratio χ_0



(b) The number of infeasible incumbents returned (out of 30) as a function of the initial acceptance ratio for different values of the soft constraint violation severity factor γ

FIGURE 7.21: The number of infeasible incumbents returned (out of 30) as a function of the soft constraint violation severity factor γ and the initial acceptance ratio χ_0 when solving the IEEE-RTS with minimisation of the probability of unit failure as scheduling criterion.

TABLE 7.18: Mean optimality gaps for all combinations of parameter values during of the second phase of the parameter optimisation experiment involving variation of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ for the IEEE-RTS with minimisation of the probability of unit failure as scheduling criterion.

		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	2.5833	2.6319	3.4465
		2	2.4360	2.6981	4.4769
		4	2.4820	3.8422	5.6140
	0.9	1	2.7257	2.7730	2.9354
		2	2.5274	2.6036	4.1051
		4	2.5990	2.9321	5.1915
	0.95	1	6.3869	5.6438	5.6974
		2	2.7137	2.7433	3.3881
		4	2.7821	2.5004	3.8992

TABLE 7.19: Mean computation times required for all combinations of parameter values during the second phase of the parameter optimisation experiment involving variation of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ for the IEEE-RTS with minimisation of the probability of unit failure as scheduling criterion.

		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	28800	28800	12125
		2	28800	24548	4378
		4	28800	10380	2024
	0.9	1	28800	28800	16820
		2	28800	28261	6536
		4	28800	23133	2610
	0.95	1	28800	28800	28800
		2	28800	28800	16150
		4	28800	28336	6121

0.95, no clear change is observed in the average optimality gap. This is observed in Table 7.18 — as one moves down the table, no clear change in the average optimality gaps are observed. This may also be seen in Figure 7.22(a), where it is observed that the mean optimality gaps remain relatively constant as a function of the cooling parameter. The mean optimality gaps corresponding to the cooling parameter values of 0.85, 0.90 and 0.95 are 3.357%, 3.182% and 3.989%, respectively, which represent a small variation. In Figure 7.22(a) it is observed that no clear correlation exists between the spreads of the optimality gaps as a function of the cooling parameter. It is, however, observed in Figure 7.22(a) that the cooling parameter value 0.90 results in the smallest spread of optimality gaps, but exhibits a number of outliers.

As for the 21-unit test system, the correlation between the reheating parameter and the mean optimality gap is much more prominent than that between the mean optimality gap and the cooling parameter. As the reheating parameter increases in Table 7.18 (*i.e.* moving from left to right in the table), an increase in the mean optimality gap is observed. This is also observed in the box plot in Figure 7.22(b), which compares the mean optimality gaps for the three different reheating parameter values of 0.55, 0.75 and 0.95. Here a clear increase in the mean optimality gap is observed as the reheating parameter increases. The mean optimality gaps corresponding to the reheating parameter values of 0.55, 0.75 and 0.95 are 3.039%, 3.175%

TABLE 7.20: Number of infeasible incumbents (out of 30) returned for all combinations of parameter values during the second phase of the parameter optimisation experiment involving variations of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ for the IEEE-RTS with minimisation of the probability of unit failure as scheduling criterion.

		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	4 (13.33%)	2 (6.67%)	1 (3.33%)
		2	3 (10.00%)	1 (3.33%)	5 (16.66%)
		4	1 (3.33%)	1 (3.33%)	9 (30.00%)
	0.9	1	2 (6.67%)	1 (3.33%)	2 (6.67%)
		2	3 (10.00%)	0 (0.00%)	1 (3.33%)
		4	2 (6.67%)	3 (6.67%)	5 (16.67%)
	0.95	1	1 (3.33%)	2 (6.67%)	2 (6.67%)
		2	0 (0.00%)	3 (10.00%)	1 (3.33%)
		4	2 (6.67%)	3 (10.00%)	3 (10.00%)

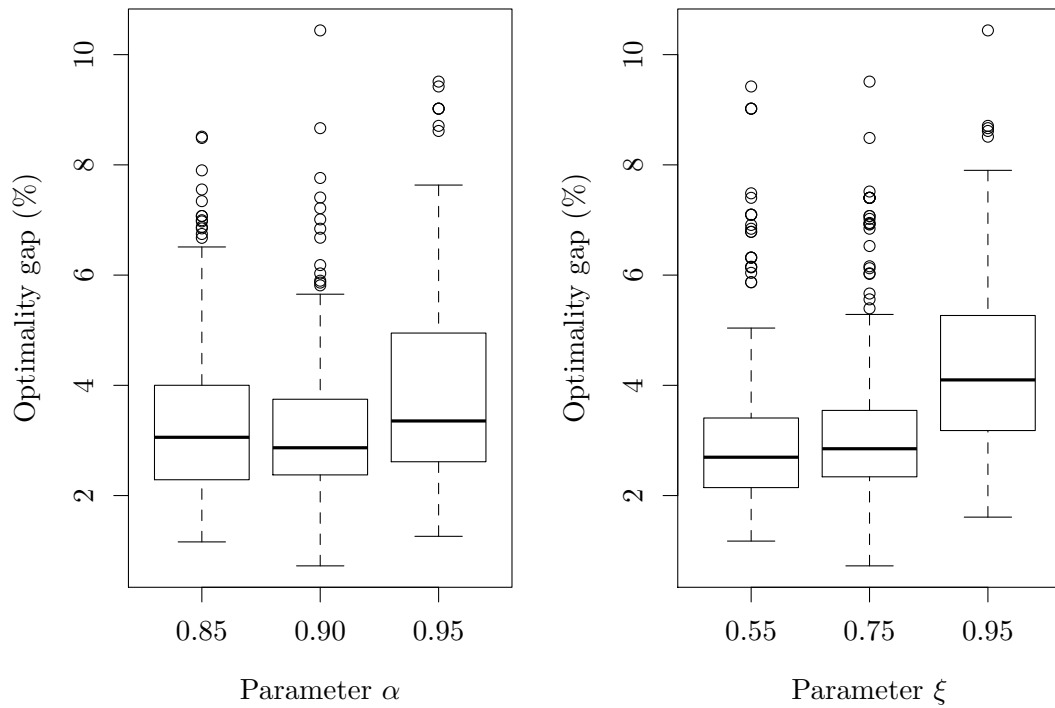
and 4.313%, respectively. It is also observed in Figure 7.22(b) that as the reheating parameter increases, an undesirable increase results in the spreads of the optimality gaps. Furthermore, the reheating parameter value 0.75 exhibits the largest number of outliers with respect to the optimality gap.

An analysis of the epoch parameter in terms of the mean optimality gap shows no real increase or decrease in the optimality gap as the epoch parameter increases. This is also visible in Figure 7.22(c). In this box plot, it may be seen that as the epoch parameter increases from 1 to 4, no significant change is observed in respect of the optimality gaps. The mean optimality gaps for epoch parameter values of 1, 2 and 4 are 3.885%, 3.085% and 3.557%, respectively. This is also the case for the spreads and the number of outliers of the optimality gaps. No clear response in these values is observed as the epoch parameter value increases.

The effects of these three parameter combinations on the required computation time are reported in Table 7.19. The mean computation time required to solve the IEEE-RTS for each combination of the three parameter values is presented in Figure 7.23. Here it may be observed that as cooling parameter increases, an increase in the required computation time results. The mean computation time required for the cooling parameter values of 0.85, 0.90 and 0.95 are 18 739 seconds, 21 395 seconds and 24 823 seconds, respectively. As explained previously, this increase in computation time may be the result of a higher cooling parameter providing a slower decay in the temperature of the SA algorithm which may, in turn, cause the algorithm to terminate after a longer time.

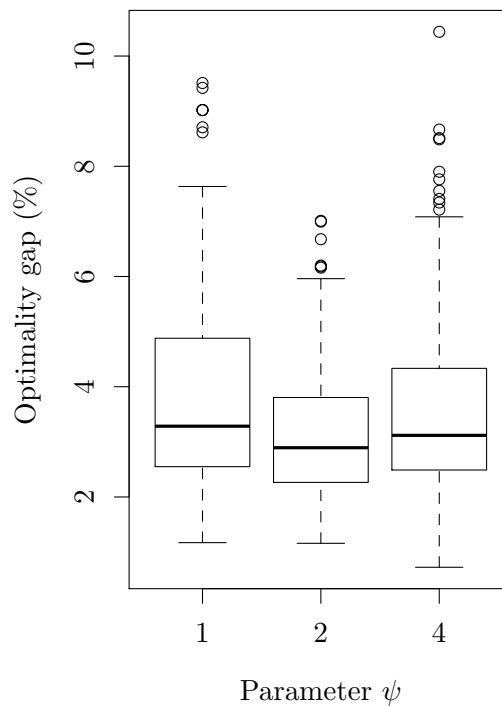
In Table 7.19 and Figure 7.23, the required computation time is observed to decrease as the reheating parameter increases. It is clear that as one moves from right to left in the table, the computation time decreases. In Figure 7.23, an almost exponential decrease is observed in the mean required computation time as the reheating parameter increases. The mean computation time required for evaluation of the reheating parameter values of 0.55, 0.75 and 0.95 are 28 800 seconds, 25 540 seconds and 10 618 seconds, respectively, which represents a very large variation. One reason for observing very short computation times at high reheating parameter values, as explained for the 21-unit test system, may be that the increase in temperature of the SA algorithm caused by a large reheating parameter is very small and so the algorithm terminates easier due to the three-consecutive-reheating termination criterion.

The effect that the epoch parameter has on the computation time required by the SA algorithm to solve the IEEE-RTS may also be observed in Table 7.19 and Figure 7.23. As the epoch



(a) Box plot comparison of the optimality gap as a function of the cooling parameter

(b) Box plot comparison of the optimality gap as a function of the reheating parameter



(c) Box plot comparison of the optimality gap as a function of the epoch parameter

FIGURE 7.22: Box plot comparison of the optimality gaps obtained during the second phase of the parameter optimisation experiment involving variation of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ for the IEEE-RTS with minimisation of the probability of unit failure as scheduling criterion.

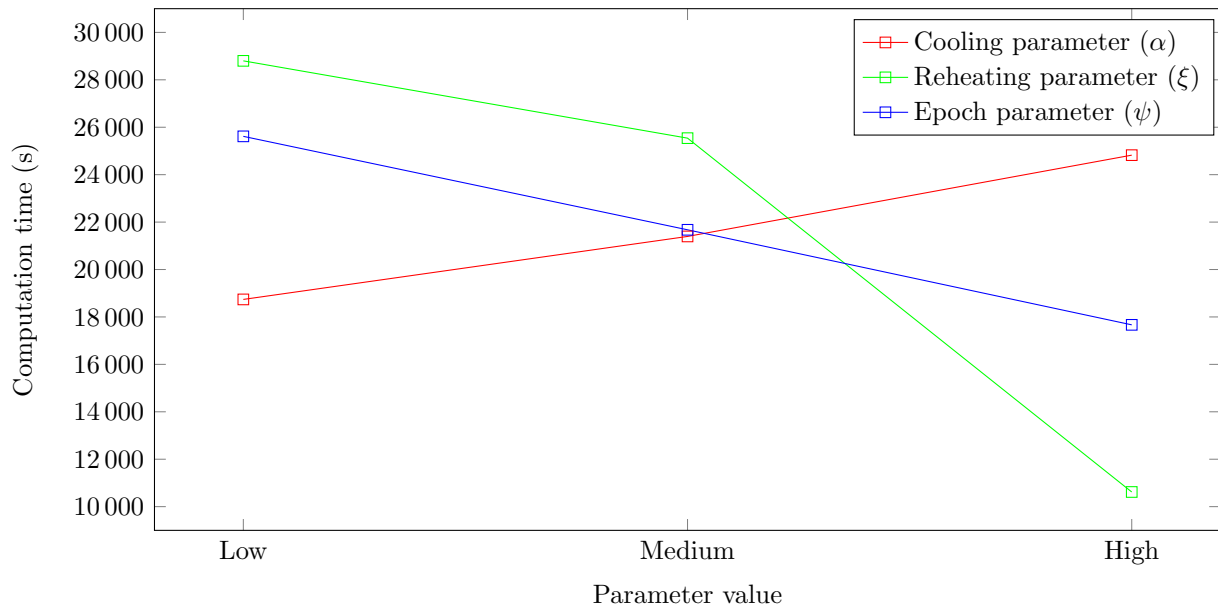


FIGURE 7.23: The computation time required to solve the IEEE-RTS as a function of the cooling parameter, the reheating parameter and the epoch parameter.

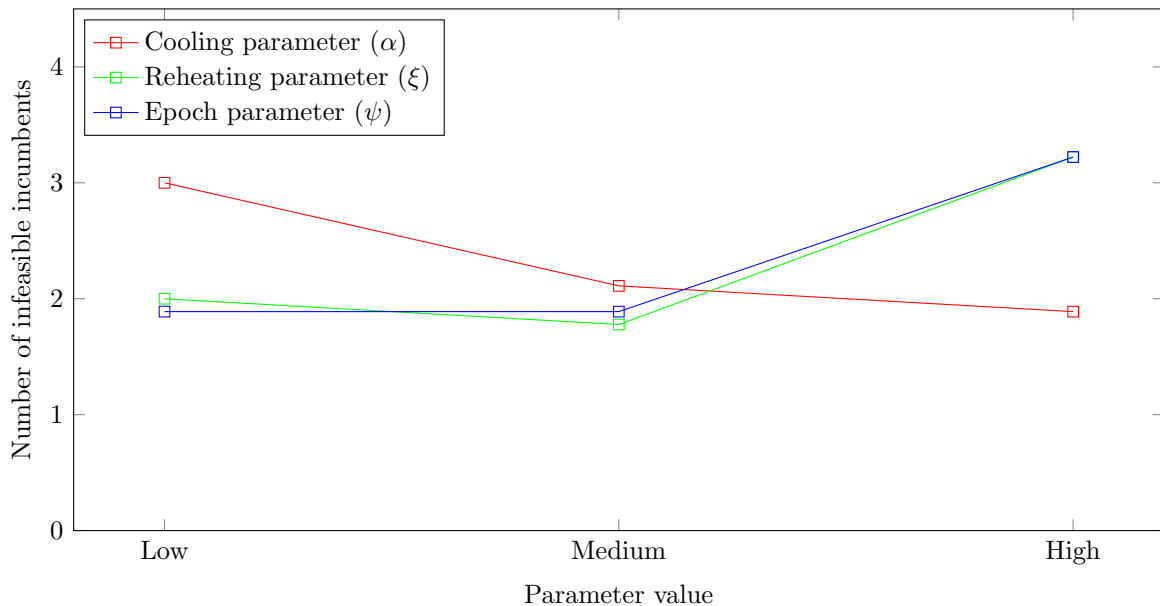


FIGURE 7.24: The mean number of infeasible incumbents (out of 30) as a function of the cooling parameter, the reheating parameter and the epoch parameter for the IEEE-RTS with minimisation of the probability of unit failure as scheduling criterion.

parameter value increases, the required computation time decreases. The mean computation time required for epoch parameter values of 1, 2 and 4 are 25 616 seconds, 21 675 seconds and 17 667 seconds, respectively, which again represents a large variation. Once again, as previously explained, a reason for this observation may be that a larger epoch parameter value causes a shorter epoch which, in turn, results in a decrease in the number of iterations required before cooling or reheating is performed. This may cause the entire SA algorithm to terminate faster, because the temperature will decrease very rapidly and three consecutive reheats will be achieved early during the algorithmic execution.

Finally, the number of infeasible incumbents returned (out of 30) during execution of the SA algorithm is reported in Table 7.20. The mean number of infeasible incumbents returned during each search run is presented in Figure 7.24. Here an almost exponential decrease in the number of infeasible incumbents returned is observed when increasing the cooling parameter. The mean number of infeasible incumbents returned (out of 30) for cooling parameter values of 0.85, 0.90 and 0.95 are 3.000, 2.111 and 1.889, respectively. In terms of the reheating parameter and epoch parameter, an almost exponential increase in the number of infeasible incumbents returned is observed as these parameter values increase. The mean number of infeasible incumbents returned (out of 30) for reheating parameter values of 0.55, 0.75 and 0.95 are 2, 1.778 and 3.222, respectively, whereas the mean number of infeasible incumbents returned (out of 30) for the epoch parameter values of 1, 2 and 4 are 1.889, 1.889 and 3.222, respectively. It has to be noted, however, that the change in the mean number of infeasible incumbents returned as the cooling parameter, the reheating parameter and the epoch parameter increase (shown in Figure 7.24) is very small, varying between a maximum mean number of infeasible incumbents returned of 3.222 to a minimum of 1.778.

The aim of the parameter optimisation experiment in this section was to obtain a suitable combination of parameters which may be used in the approximate solution approach (*i.e.* the method of SA) so as to obtain good GMS solutions for the IEEE-RTS when minimising probability of unit failure. In Figure 7.22(b), it may be observed that the large value for the reheating parameter achieves the largest mean optimality gap. This parameter value also produced the largest mean number of infeasible incumbents shown in Figure 7.24. The reheating parameter value of 0.95 was therefore eliminated from the parameter values considered as candidates for the final set of parameters in the context of the IEEE-RTS. It was furthermore observed that a reheating parameter of 0.55 results in the algorithm requiring the maximum allowed computation time (28 800 seconds) for all the cases. For this reason, this parameter value was also eliminated from the parameter values considered as candidates for the final set of parameters for the IEEE-RTS. The only reheating parameter value remaining is therefore 0.75, which achieves an acceptable mean optimality gap value (3.175%) as well as an acceptable mean computation time required by the SA algorithm (25 540 seconds). This parameter value also returns, on average, the smallest number of infeasible incumbents (1.778).

Furthermore, the cooling parameter value of 0.95 was eliminated from the candidates considered due to the parameter value exhibiting a large spread of optimality gaps as well as exhibiting the largest mean optimality gap, shown in Figure 7.22(a). From the remaining two cooling parameter values, 0.85 was selected as the best candidate for the IEEE-RTS. The reason for this is that the cooling parameter value of 0.90, although achieving a smaller mean optimality gap, exhibits the most outliers and was therefore eliminated as a candidate. Finally, in terms of the results obtained for the epoch parameter in combination with the value of 0.85 for the cooling parameter and 0.75 for the reheating parameter, it was observed that the larger value (*i.e.* 4) returned, on average, 3.222 infeasible incumbents out of thirty, which is the largest mean number of infeasible incumbents observed and was therefore eliminated as a possible candidate

for the SA algorithm. Of the remaining epoch parameter values, the value 2 was selected as the final epoch parameter as this value results in a smaller mean optimality gap as well as returning the smallest number of infeasible incumbents of 1.778 out of thirty. This epoch parameter value also returned the smallest mean optimality gap of all three epoch parameter values. The final set of parameters adopted in the SA algorithm for solving the IEEE-RTS is shown in Table 7.21.

TABLE 7.21: *The complete set of parameters selected for the IEEE-RTS, as well as the mean optimality gap, the required computation time and the number of infeasible incumbents associated with these parameter values during the second phase of the parameter optimisation experiment with minimisation of the probability of unit failure as scheduling criterion.*

χ_0	γ	α	ξ	ψ	Optimality gap (%)	Computation time (s)	Infeasibilities
0.5	0.75	0.85	0.75	2	2.6981	24548	1

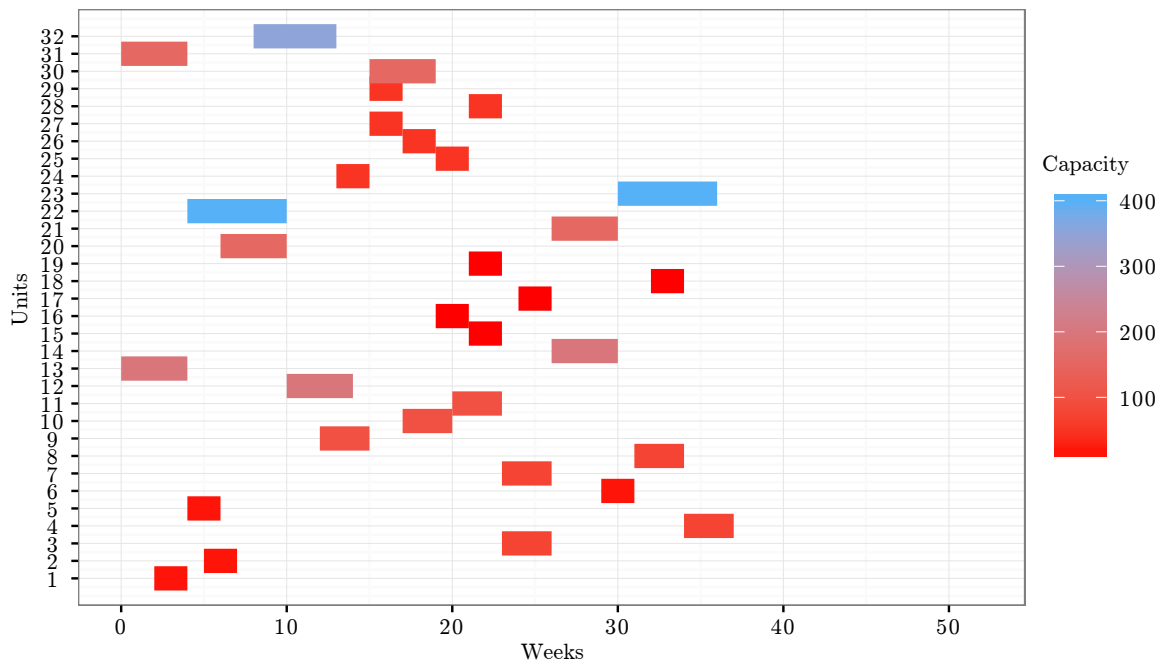
Numerical results

An approximate solution to the linear GMS model of §4.3 was obtained for the IEEE-RTS by the SA algorithm with the parameter combinations as specified in Table 7.21. The decision variable values of the incumbent are given in integer decision vector form by $\mathbf{x} = [5, 3, 18, 38, 6, 29, 24, 35, 21, 11, 15, 1, 7, 34, 25, 22, 40, 36, 41, 13, 31, 5, 27, 26, 23, 24, 19, 30, 21, 1, 17, 9]$, which corresponds to an objective function value of 58.557 (1.507% worse than that of the optimal solution for the IEEE-RTS, as reported in §7.1.2).

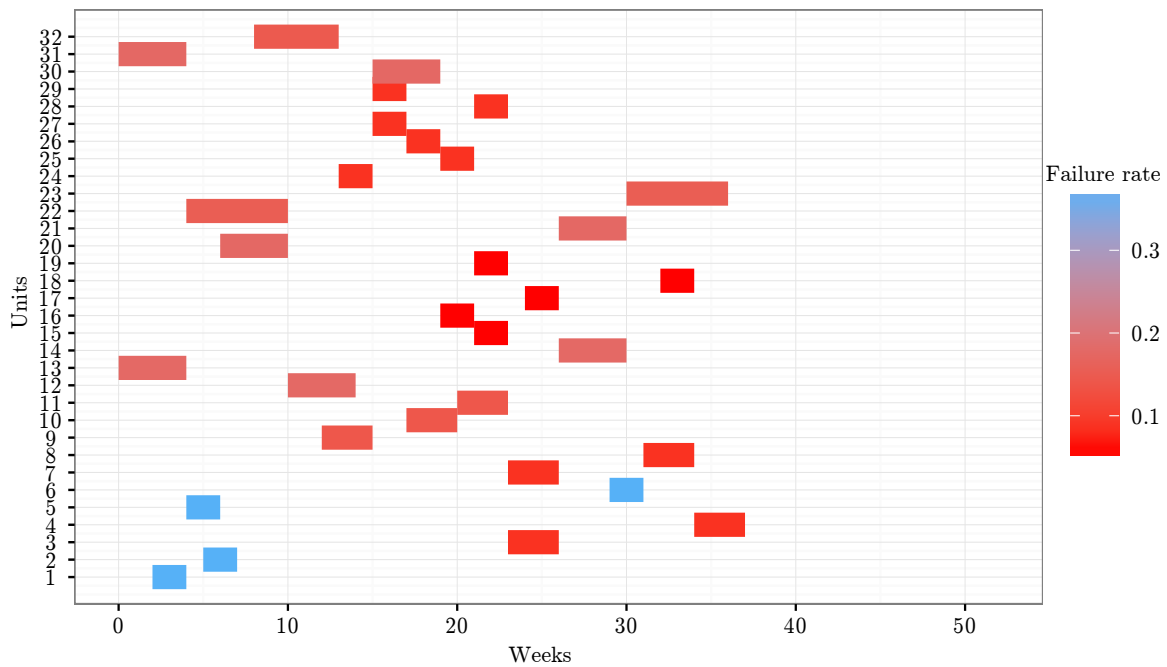
A graphical representation of this maintenance schedule is presented in Figure 7.25 with the colour scale in Figure 7.25(a) indicating the rated capacity (in MW) of each PGU and the colour scale in Figure 7.25(b) indicating the failure rate of each PGU. The manpower required over the duration of the scheduling window in order to implement the solution in Figure 7.25 is shown in Figure 7.26(a), while the available system capacity over the duration of the scheduling window associated with this solution is shown in Figure 7.26(b).

The solution may be analysed by comparing it with the optimal solution obtained in §7.1.2 for the IEEE-RTS. A graphical representation of the incumbent solution returned by the approximate solution methodology of §5.3 (*i.e.* the method of SA) and the optimal solution obtained by the exact solution approach described in §7.1.2 (*i.e.* by means of CPLEX) is shown in Figure 7.27 for the IEEE-RTS. The maintenance schedules of the two solutions are compared in the figure, with the colour scale indicating the rated capacity of the PGUs in the system. The effects of the two maintenance schedules on the manpower required and the available system capacity are shown in Figures 7.28(a) and 7.28(b), respectively.

Comparing these two maintenance schedules, it is found that the objective function values only differ by 1.507% and that there are many similarities between the two maintenance schedules. Comparing the available system capacity in Figure 7.28(b), it may be seen during the early stages of the scheduling window that the available system capacity is exactly the same. After week 7, however, a small difference is observed in the available system capacity, but a similar trend is followed for both maintenance schedules. It is only after week 33 that a significant difference is observed, where the maintenance schedule returned by the exact solution approach completes maintenance on all the PGUs during planning period 38, whereas the maintenance schedule returned by the approximate solution approach only completes the maintenance during week 43. Comparing the available manpower over the scheduling window in Figure 7.28(b), a noticeable difference is observed. The manpower required for implementation of the exact solution approach is very high during the early stages of the scheduling window and only starts to decrease during week 28. The manpower required to implement the approximate solution varies much more over the scheduling window than that of the exact solution approach. During many planning periods

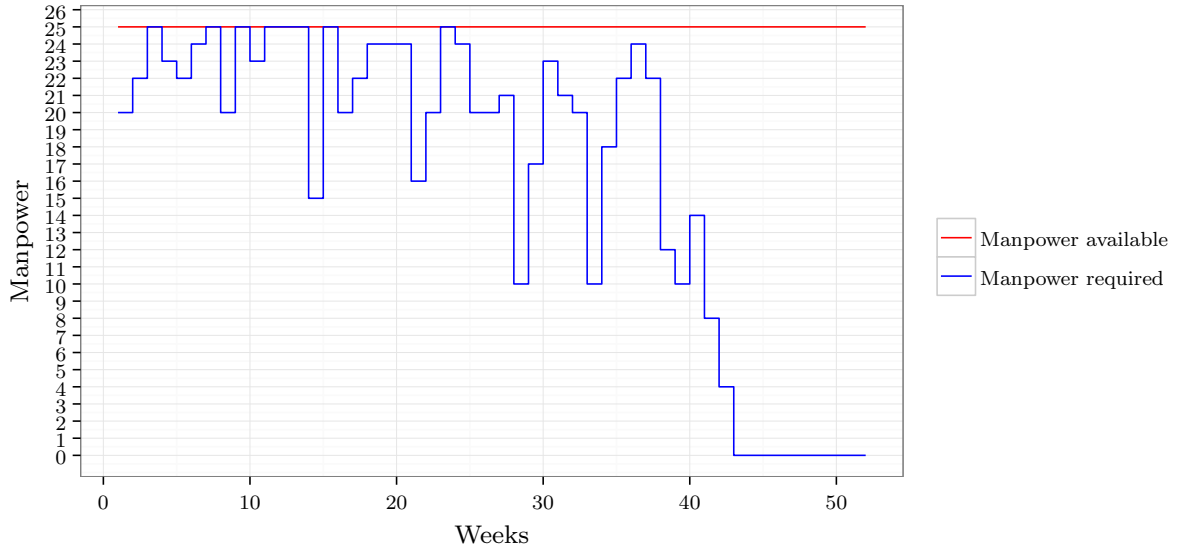


(a) The incumbent returned by the SA algorithm for the linear model of §4.3, with the colour scale indicating the capacity of the PGUs

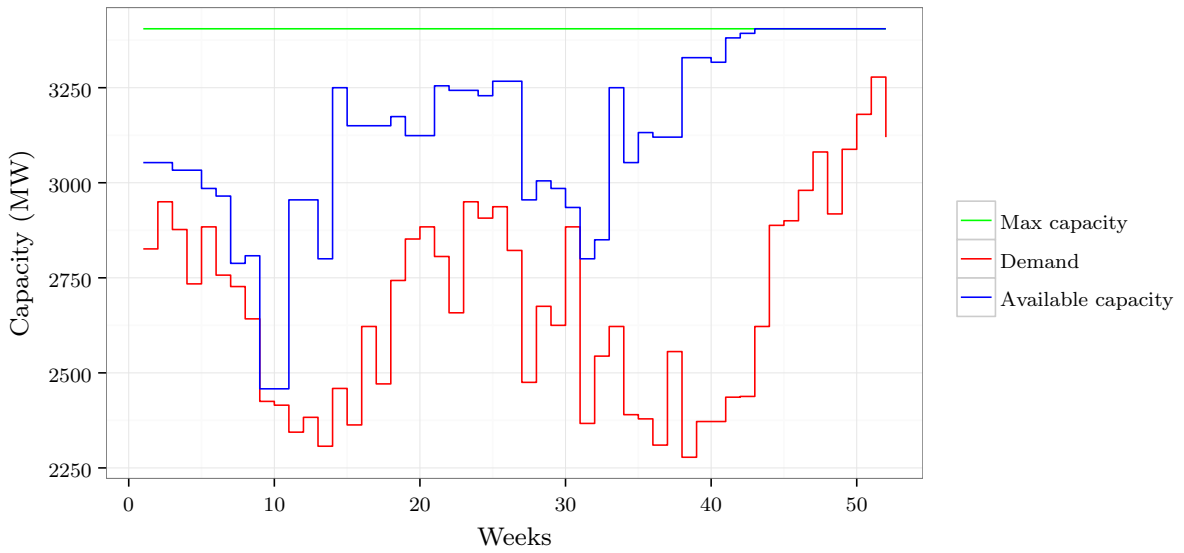


(b) The incumbent returned by the SA algorithm for the linear model of §4.3, with the colour scale indicating the failure rate of the PGUs

FIGURE 7.25: The incumbent returned by the SA algorithm for the linear model of §4.3, in the context of the IEEE-RTS for parameter values as indicated in Table 7.21 with minimisation of the probability of unit failure as scheduling criterion.



(a) The manpower required over the scheduling window corresponding to the maintenance schedule in Figure 7.25



(b) The system capacity over the scheduling window corresponding to the maintenance schedule in Figure 7.25

FIGURE 7.26: Evaluation of the manpower required and system capacity available over the duration of the scheduling window for the IEEE-RTS maintenance schedule in Figure 7.25.

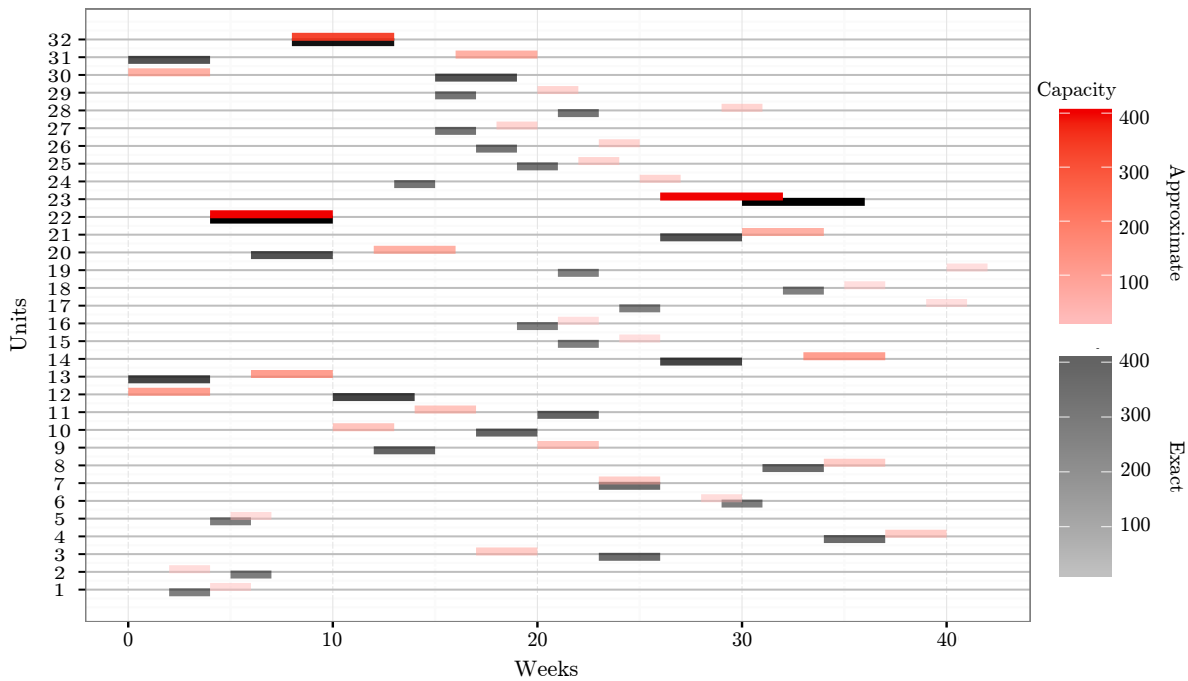


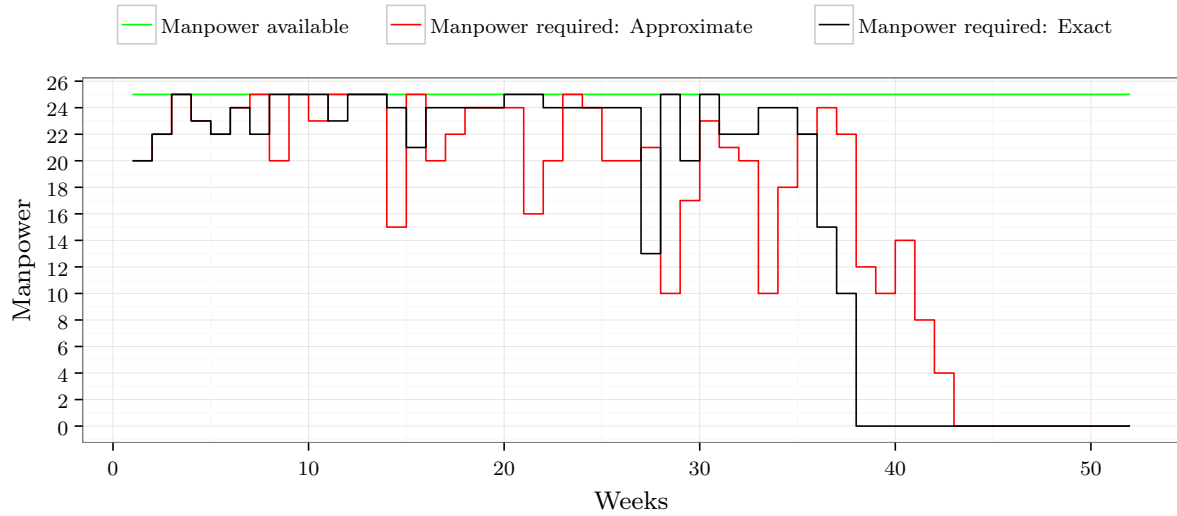
FIGURE 7.27: Two maintenance schedules for the IEEE-RTS returned by two different approaches towards solving the GMS problem of §4.3 with minimisation of the probability of unit failure as scheduling criterion.

of the scheduling window the manpower drops below 16 for the approximate solution approach. It is also observed that manpower is only utilised until week 43 to implement the maintenance schedule returned by the approximate solution, whereas the manpower is only utilised until week 38 for the exact solution approach.

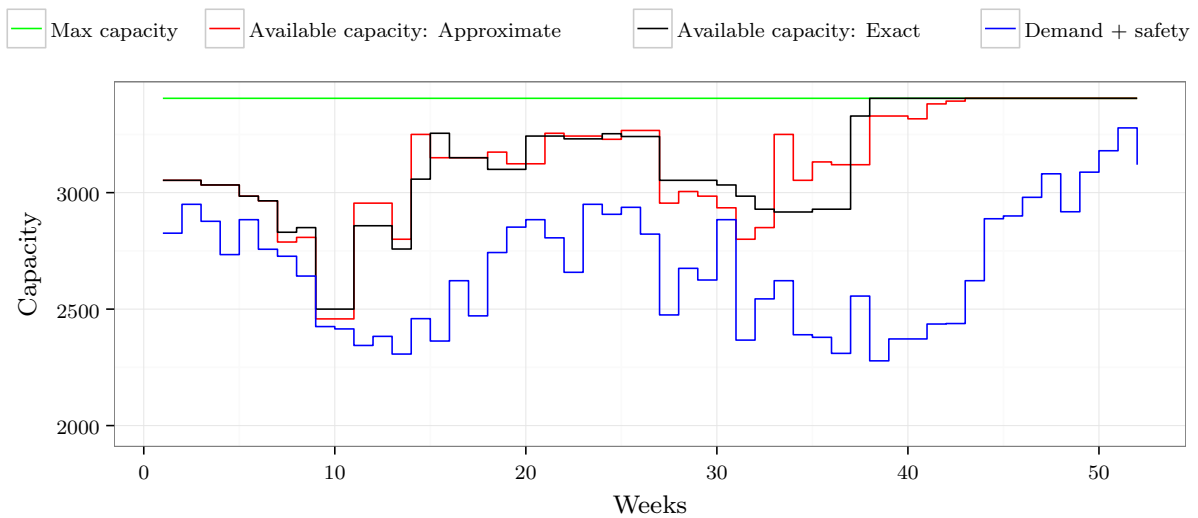
7.3 Chapter summary

In this chapter, the results obtained by employing an exact solution approach toward solving the linear GMS model of §4.3 were reported in the contexts of a 21-unit test system [54] and the 32-unit IEEE-RTS [188]. The results for the 21-unit test system were presented in §7.1, which included an optimal maintenance schedule for the problem instance, a comparison with a maintenance schedule according to another scheduling objective from the literature and an analysis of the practical feasibility of an exact solution approach for the 21-unit test system in the form of a sensitivity analysis in §7.1.1. The same study was performed for the IEEE-RTS, with the presentation of an optimal maintenance schedule for the problem instance, a comparison with a maintenance schedule according to another scheduling objective from the literature and an analysis of the practical feasibility of an exact solution approach for the 32-unit IEEE-RTS in the form of a sensitivity analysis in §7.1.2.

It was demonstrated that for small problem instances with tightly constrained maintenance window constraints, such as the original 21-unit test system and the original 32-unit IEEE-RTS, an exact solution approach is practically feasible for the linear GMS model of §4.3 (*i.e.* requires a computation time in the order of a few hundred seconds). It was also observed that an increase in the peak demand of the system does not affect the processing time significantly. Relaxation of the maintenance window constraints, however, has a dramatic influence on the computation



(a) The required manpower over the scheduling window corresponding to the maintenance schedules in Figure 7.27



(b) The system capacity over the scheduling window corresponding to the maintenance schedules in Figure 7.27

FIGURE 7.28: Comparison between the manpower required and the available system capacity associated with the schedules in Figure 7.27 for the IEEE-RTS.

time of an exact solution approach via CPLEX. It was also found that the computation time required to solve the original IEEE-RTS exactly was more than 20 times more than that of the 21-unit test system. It is therefore anticipated that adopting an exact solution approach toward solving the linear GMS model of §4.3 for a real-world problem instance, which may easily contain more than 100 PGUs, will not be practically feasible. An approximate solution approach was consequently explored in §7.2 in order to be able to accommodate larger power systems.

The results obtained by the approximate solution approach employed, the method of SA, was presented in §7.2 for the same the two academic benchmark systems. This was followed by

a presentation of the results obtained from a parameter optimisation experiment for the 21-unit test system in §7.2.1. In this section, the different combinations of parameter values were compared in order to obtain the best combination for the method of SA in the context of the 21-unit test system. This section also contained a description of the best incumbent returned by solving the 21-unit system upon utilisation of the best parameter combination values. A comparison of this solution with the optimal solution obtained by the exact solution was also carried out in §7.1.1. It was found that the objective function values of the solution obtained by the exact solution approach and the approximate solution approach differ by only 0.0689%. A similar approach was taken for the IEEE-RTS in §7.2.2 where the results of the parameter optimisation experiment, as well as a comparison of the difference in the combinations of the parameter values, were presented. Finally, the best incumbent returned by solving the IEEE-RTS upon utilisation of the best parameter value combination was presented and a comparison was performed with the optimal solution obtained by the exact solution approach. For the IEEE-RTS, it was found that the objective function values of the solution obtained by the exact solution approach and that obtained by the approximate solution approach differ by 1.507%.

CHAPTER 8

Maximising expected energy production

Contents

8.1 Piecewise linear approximation results	157
8.2 Metaheuristic approximate solution approach	177
8.3 Chapter summary	204

The results obtained for the two academic benchmark systems (presented in §6) when adopting the maximisation of expected energy production objective function (described in §4.3) are presented in this chapter. This presentation includes the results obtained by employing the piecewise linear approximation and the metaheuristic approximate solution approaches as described in §5.

8.1 Piecewise linear approximation results

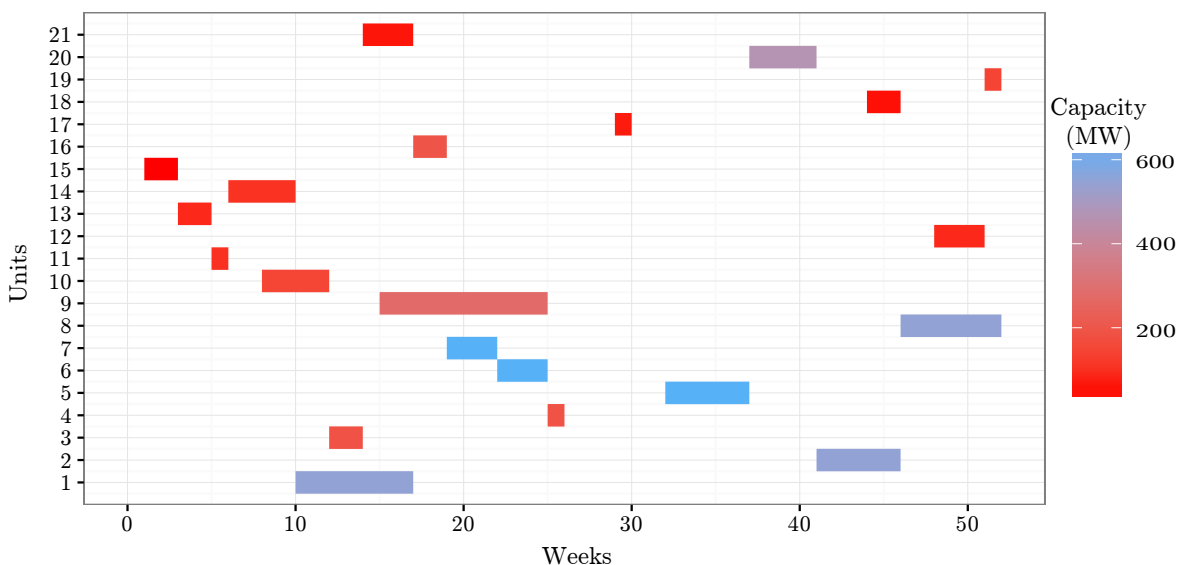
Piecewise linear approximation solutions to the nonlinear GMS model of §4.3 are presented in this section within the context of the two GMS test systems reviewed in §6. These piecewise linear approximation solutions are also contrasted with solutions from the literature for the same test systems, but in which another GMS objective within the class of reliability criteria was adopted, as well as with results obtained by the exact method when minimising the probability of unit failure objective function, as reported in §7. A personal computer, with an Intel CoreTM i7-4770 processor and 8 GB RAM running at 3.4 GHz within a MicrosoftTM Windows 7 64-bit operating system was used to perform all the computational evaluations reported in this chapter.

8.1.1 The 21-unit system

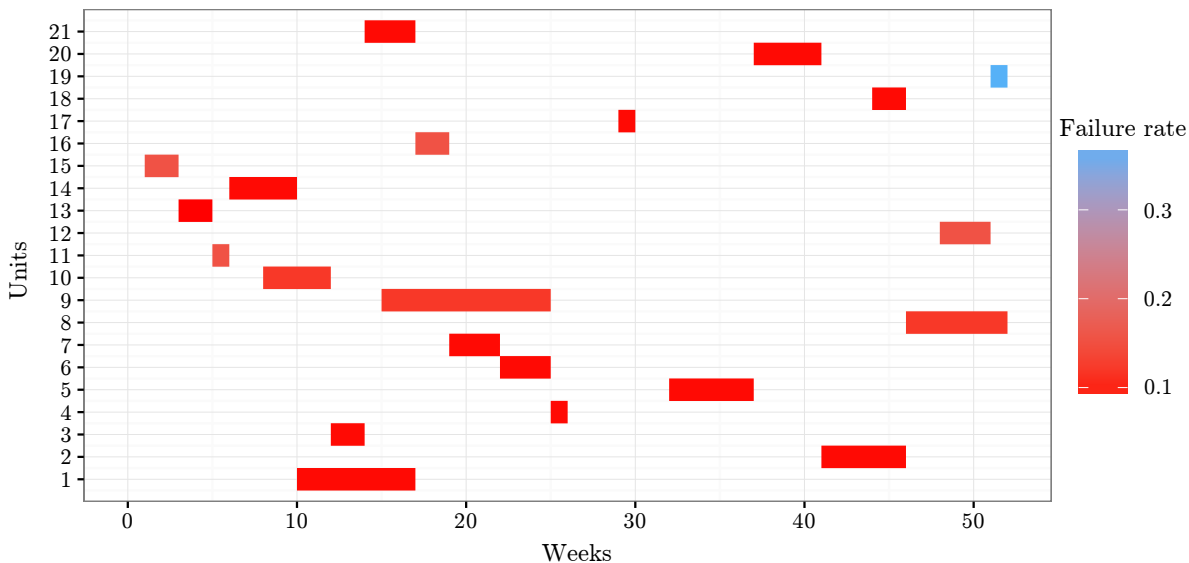
In this section, a piecewise linear approximation solution obtained by CPLEX for the nonlinear model of §4.3 as applied to the 21-unit test system [54] is presented. The solution is contrasted with a solution obtained in the literature upon adoption of another reliability scheduling criterion (minimisation of the SSR). The solution is also contrasted with the solution obtained in §7.1.1 when adopting the minimisation of the probability of unit failure objective function. The feasibility of a piecewise linear approximation solution approach for the nonlinear model in §4.3 within the context of the 21-unit test system is also analysed in the form of a sensitivity analysis involving various relaxations of the demand and maintenance scheduling window constraints.

Numerical results

A piecewise linear approximation solution to the nonlinear GMS model of §4.3 is obtained for the 21-unit test system of §6.1 by CPLEX within 58 seconds. The optimal decision variable values of this solution are given in integer decision vector form by $\mathbf{x} = [11, 42, 13, 26, 33, 23, 20, 47, 16, 9, 6, 49, 4, 7, 2, 18, 30, 45, 52, 38, 15]$. These values correspond to an optimal objective function value of 219 717 MW· week (36 912 456 MWh). The nonlinear objective function value for the optimal decision variable values is 219 996 MW· week (36 959 328 MWh). This corresponds to a 0.127% difference in objective function value between the piecewise linear approximation and the nonlinear function. A graphical representation of the optimal maintenance schedule is presented in



(a) An optimal maintenance schedule obtained by piecewise linear approximation of the nonlinear GMS model objective, with the colour scale indicating the capacities of the PGUs in the 21-unit test system

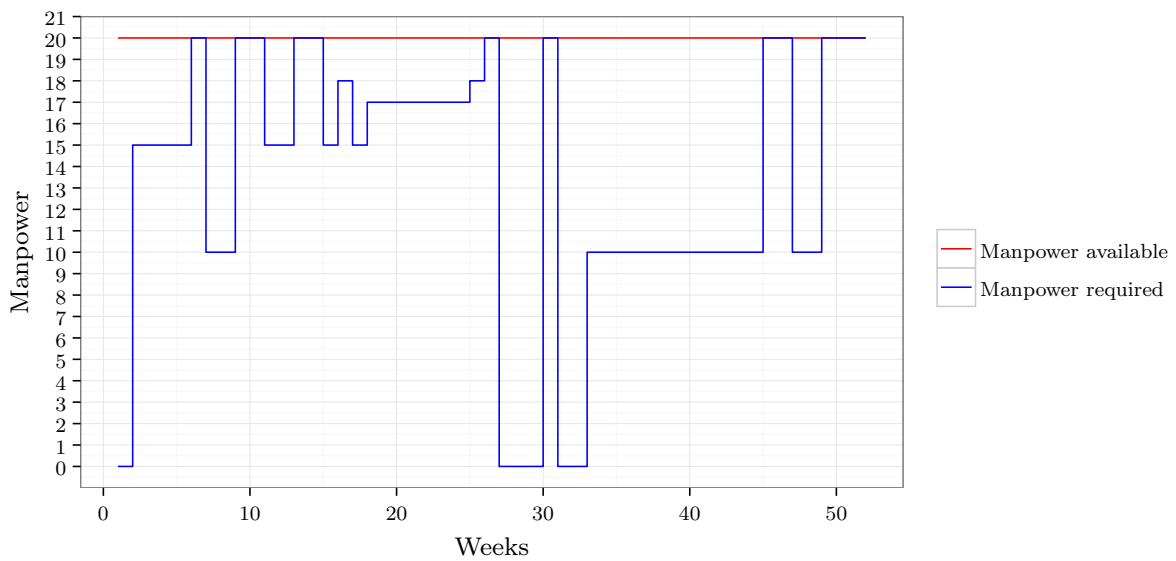


(b) An optimal maintenance schedule obtained by piecewise linear approximation of the nonlinear GMS model objective, with the colour scale indicating the failure rates of the PGUs in the 21-unit test system

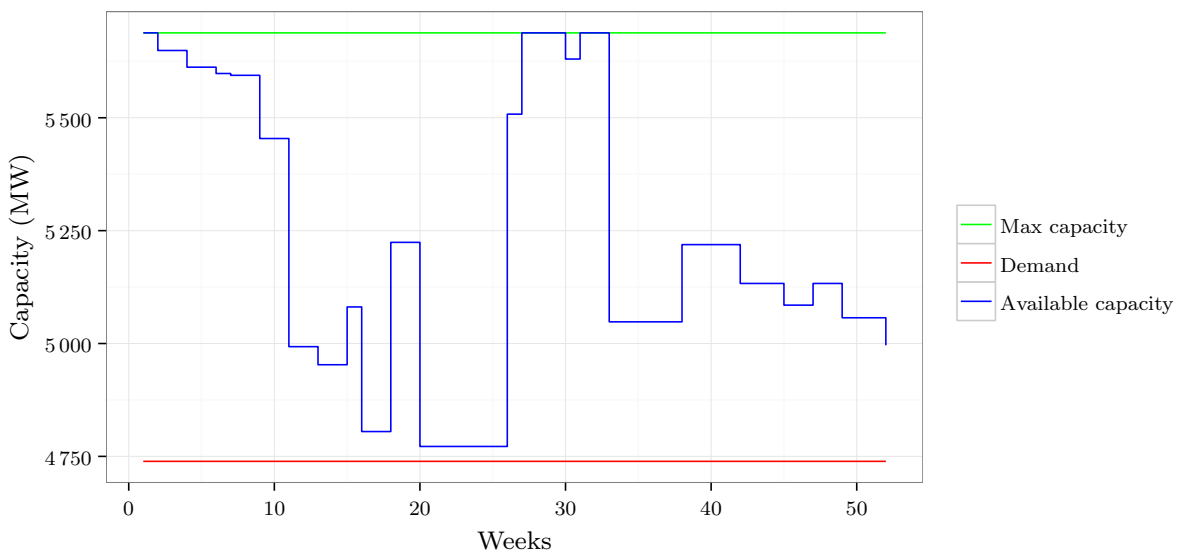
FIGURE 8.1: An optimal solution to a piecewise linearisation of the nonlinear model of §4.3 for the 21-unit test system.

Figures 8.1(a) and 8.1(b), with the colour scale indicating the capacity (in MW) and failure rate of each PGU, respectively. The manpower required over the duration of the scheduling window to implement the optimal solution in Figure 8.1 is shown in Figure 8.2(a). The available capacity over the duration of the scheduling window associated with the optimal solution in Figure 8.1 is shown in Figure 8.2(b). In Figure 8.3, the expected energy production for each of the PGUs is presented, with the maintenance starting at the dates indicated by means of black dots.

Figures 8.1, 8.2 and 8.3 may be used in conjunction to analyse the solution to the nonlinear GMS model of §4.3 obtained by the piecewise linear approximation for the 21-unit test system.



(a) The manpower required over the scheduling window corresponding to the optimal maintenance schedule in Figure 8.1



(b) The system capacity over the scheduling window corresponding to the optimal maintenance schedule in Figure 8.1

FIGURE 8.2: Evaluation of the manpower required and the system capacity available over the duration of the scheduling window for the 21-unit test system.

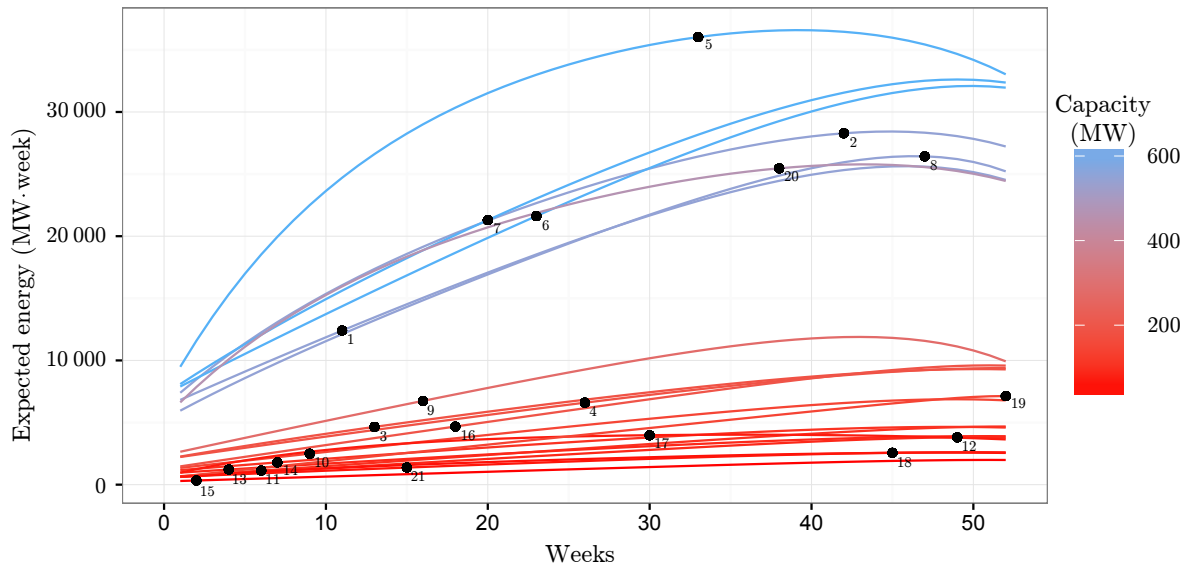


FIGURE 8.3: Graphical representation of the expected energy production for each PGU of the 21-unit test system with the starting time of planned maintenance indicated by the dot as per the optimal solution shown in Figure 8.1.

For instance, the influence that the rated capacity of a PGU has on its scheduled maintenance starting time is clear in Figures 8.1(b) and 8.3. PGUs with large capacities (PGUs 2, 5, 8 and 20) are scheduled for maintenance commencement as close as possible to the peaks of their expected energy curves. This is also the case for PGUs 1, 6 and 7, but as these PGUs have maintenance windows that end at planning period 27; they therefore cannot be scheduled for maintenance later than planning period 27. It is therefore clear that good solutions to the nonlinear GMS model of §4.3 (when maximising expected energy production) will seek to schedule PGUs with larger capacities for maintenance as close as possible to the dates at which these PGUs are expected to produce the most energy over the scheduling window (*e.g.* near the peaks of their expected energy production curves). In Figure 8.2(b), the available capacity is observed to be at a maximum near the beginning of the scheduling window, but generally diminishes towards planning period 27, which is typically when the second set of maintenance window constraints start. After planning period 27, the available capacity is again at a maximum whereafter it diminishes towards the end of the scheduling window.

Comparison with results from the literature

The effects of adopting a scheduling criterion which seeks to maximise the expected energy production over the scheduling window, as in the newly proposed objective function of §4.3.2, may better be analysed by comparing the results reported above with results found in the literature for the same test problem when adopting other reliability scheduling criteria, such as minimisation of the SSR. The results reported above are therefore contrasted with the results obtained by Schlünz and Van Vuuren [188] who adopted the SSR scheduling criterion in (2.8). The results obtained by Schlünz and Van Vuuren [188] (referred to here as Scenario E) are compared with the results reported above (referred to here as Scenario F). Both GMS objectives adopted in Scenarios E and F reside within the class of reliability scheduling criteria and the results of Schlünz and Van Vuuren [188] for the 21-unit test system (with the minimisation of SSR as objective) represent the best results available in the literature for this particular scheduling criterion and problem instance combination. A graphical representation of the two maintenance

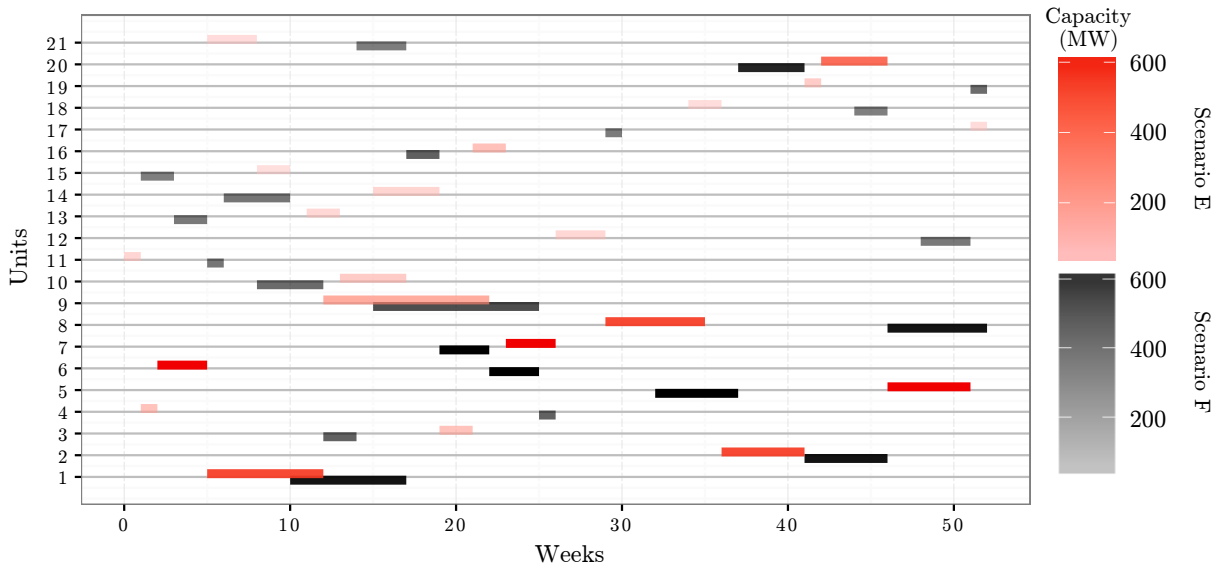
schedules and their effects on the manpower required, as well as the available capacity, are shown in Figure 8.4 for the 21-unit test system. In Figure 8.4(a), the maintenance schedules of the two scenarios are contrasted, with the colour scales indicating the rated capacity of the PGUs in the system. The effects of the two maintenance schedules on the manpower required and the available capacity are shown in Figures 8.4(b) and 8.4(c), respectively. A comparison between the results for the two scenarios in terms of both scheduling objectives is presented in Table 8.1. The optimal solution of Scenario F performs 18.66% worse than that of Scenario E in terms of the SSR scheduling objective. A similar observation is made when taking the best solution obtained for the maximisation of expected energy scheduling criterion. In this case, the solution obtained for Scenario F performs 10.29% better than that of Scenario E. The reason for this is that the two objectives conflict in terms of when maintenance should be scheduled for the PGUs. In Scenario E, the objective function aims to spread the maintenance of PGUs with large capacities out over the entire scheduling window in order to levelise the reserves of the system. The objective function of Scenario F, on the other hand, aims to schedule planned maintenance for the PGUs towards the end of the scheduling window close to the peaks of the PGUs' expected energy production curves.

TABLE 8.1: Comparison between the objective function values associated with the maintenance schedules in Figure 8.4(a) for Scenarios E and F in the context of the 21-unit test system. The percentage change values are computed for the solution of Scenario F relative to that of Scenario E.

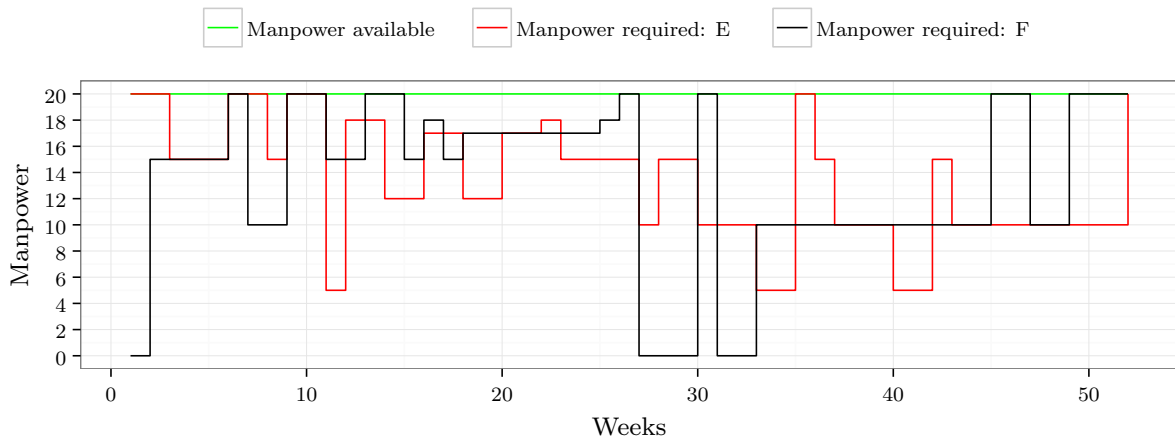
Scenario	SSR (MW ²)	Expected energy production (MW·week)
E	13 664 879	197 109
F	16 799 009	219 717
Percentage change	+18.66%	+10.29%

In order to minimise the SSR, maintenance schedules typically exhibit maintenance commencement dates that are spread out over the entire scheduling window, as mentioned above. This is the case in Scenario E where maintenance on PGUs with large capacities is performed throughout the scheduling window. The scheduling objective in Scenario F, however, aims to schedule PGUs with large rated capacities (PGUs that can produce more energy) close to the peaks of the expected energy curves of these PGUs (typically close to the end of the scheduling window). It may be seen in Figure 8.4(a) that PGUs with large rated capacities are typically scheduled later, but within their respective PGU maintenance windows. It may also be seen in Figure 8.4(a) that, compared to Scenario E, some PGUs with higher failure rates are scheduled for planned maintenance later during the scheduling window. PGU 19, for example, is scheduled for maintenance during the last week of the scheduling window due to exhibiting the highest failure rate of 0.3733, which is 10 weeks later than observed in Scenario E. A similar observation may be made for PGU 12, which is scheduled for maintenance 22 weeks later in Scenario F than in Scenario E due to its large failure rate of 0.1527.

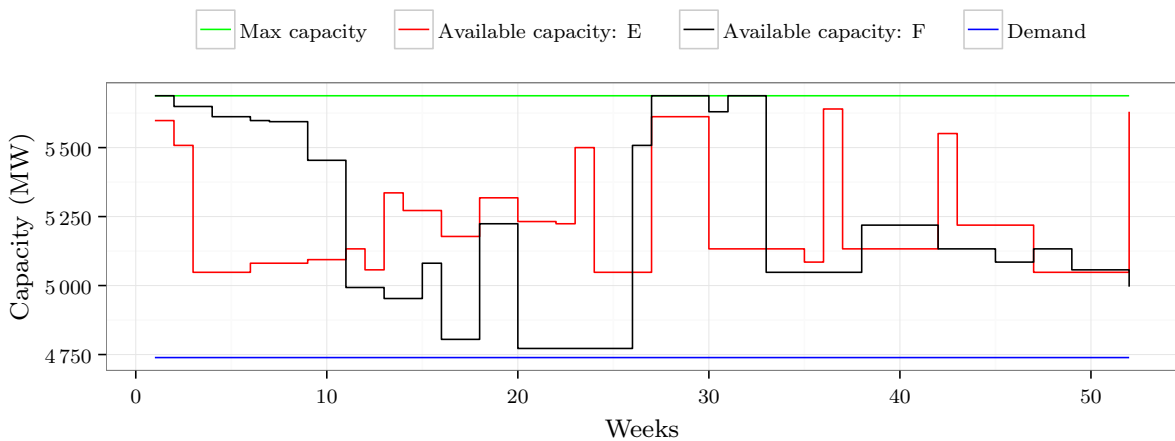
The different effects of the two scheduling objectives may also be observed in Figure 8.4(b). In Scenario E, the manpower required for planned maintenance is spread out over the entire scheduling window. During each planning period of the problem instance, manpower is required as there is no planning period within the scheduling window during which no PGUs are in maintenance. In Scenario F, on the other hand, it is observed that at the end of both halves of the scheduling window (*e.g.* weeks 1–27 and weeks 28–52) the manpower required is at the maximum available number as most of the PGUs exhibit peaks of their expected energy production curves towards the end of the scheduling window.



(a) Two maintenance schedules for the 21-unit test system corresponding to different GMS criteria



(b) The manpower required over the duration of the scheduling window for the two maintenance schedules in Figure 8.4(a)



(c) The available system capacity over the duration of the scheduling window for the two maintenance schedules in Figure 8.4(a)

FIGURE 8.4: Comparison between the maintenance schedules of Scenarios E and F for the 21-unit test system.

Similar observations may also be made in respect of Figure 8.4(c). In Scenario E, the available system capacity never falls below a 6% band above the demand. In Scenario F, on the other hand, the available system capacity drops down to 0.696% above the demand during the middle stages of the scheduling window as PGUs with large capacities (*e.g.* PGUs 6, 7, 9) are scheduled for maintenance towards the end of the first half of the scheduling window (due to their particular maintenance window constraints). The maximum system capacity is mainly available towards the middle of the graph for both Scenarios E and F, whereas fewer jumps are observed in the available capacity for Scenario E than for Scenario F.

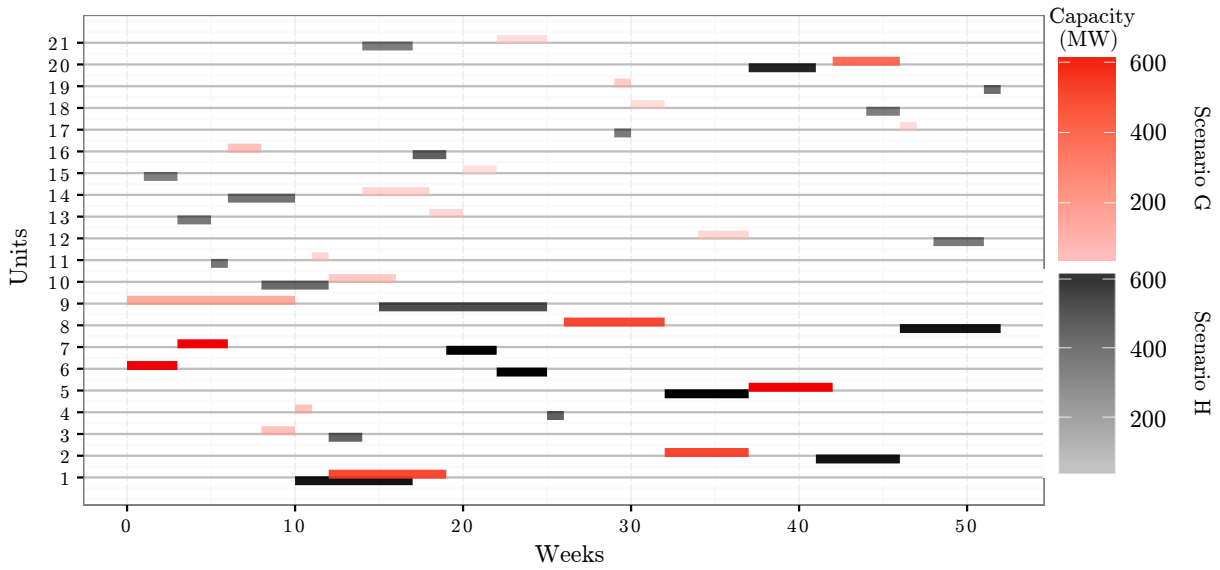
Comparison with results obtained by minimising probability of unit failure

The effects of the two novel scheduling criteria proposed in this dissertation (minimising the probability of unit failure and maximising the expected energy production) may better be contrasted with each other by comparing the results obtained for these scheduling criteria in the context of the 21-unit test system. The results obtained when minimising the probability of unit failure (referred to here as Scenario G), reported in earlier in this section, are compared with the results obtained by maximising the expected energy production (referred to here as Scenario H), reported in §8.1.1. Both GMS objectives adopted in Scenarios G and H reside within the class of the reliability scheduling criteria and therefore a comparison between these two reliability scheduling criteria is justified. A graphical representation of the two maintenance schedules of Scenarios G and H and their effects on the manpower required, as well as on the available capacity, is shown in Figure 8.5 for the 21-unit test system.

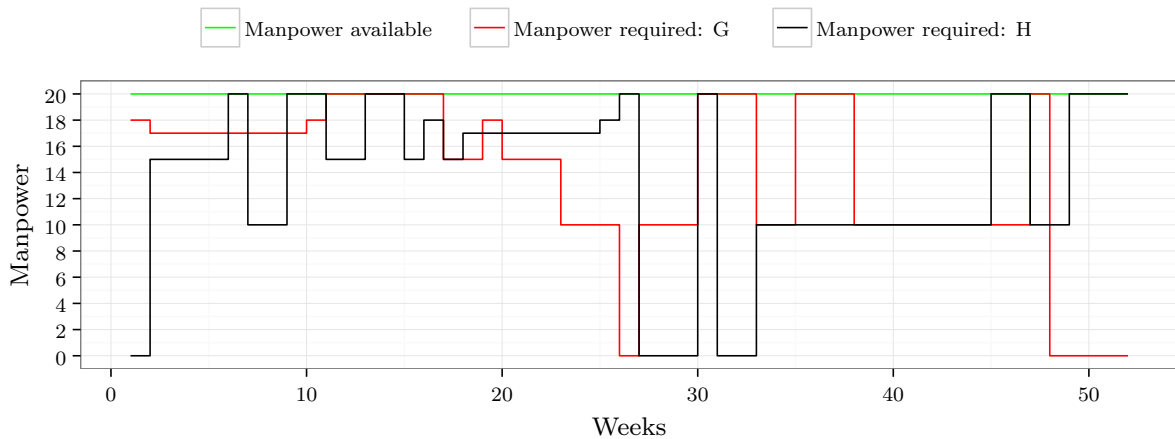
In Figure 8.5(a), the maintenance schedules of the two scenarios are compared, with the colour scales indicating the rated capacity of the PGUs in the system. The effects of the two maintenance schedules on the manpower required and the available capacity are shown in Figures 8.5(b) and 8.5(c), respectively. A comparison between the results for the two scenarios is furthermore presented in terms of both scheduling objectives in Table 8.2. The solution of Scenario H performs 15.70% worse than that of Scenario G in terms of minimising the probability of unit failure. A similar observation is made when considering the best solution obtained for the maximisation of expected energy scheduling criterion. In this case, the solution obtained for Scenario H performs 17.88% better than that of Scenario G. The reason for this is that the two objectives conflict with each other in terms of when maintenance should be scheduled for the PGUs. In Scenario G, the objective function aims to schedule planned maintenance as early as possible during the scheduling window so as to decrease the probability of PGU failure. The objective function of Scenario H, on the other hand, aims to schedule planned maintenance for the PGUs towards the end of the scheduling window close to the peaks of the PGUs' expected energy production curves.

TABLE 8.2: Comparison between the objective function values associated with the maintenance schedules in Figure 8.5(a) for Scenarios G and H in the context of the 21-unit test system. The asterisk indicates an optimal solution. The percentage change values are computed for the solution of Scenario H relative to that of Scenario G.

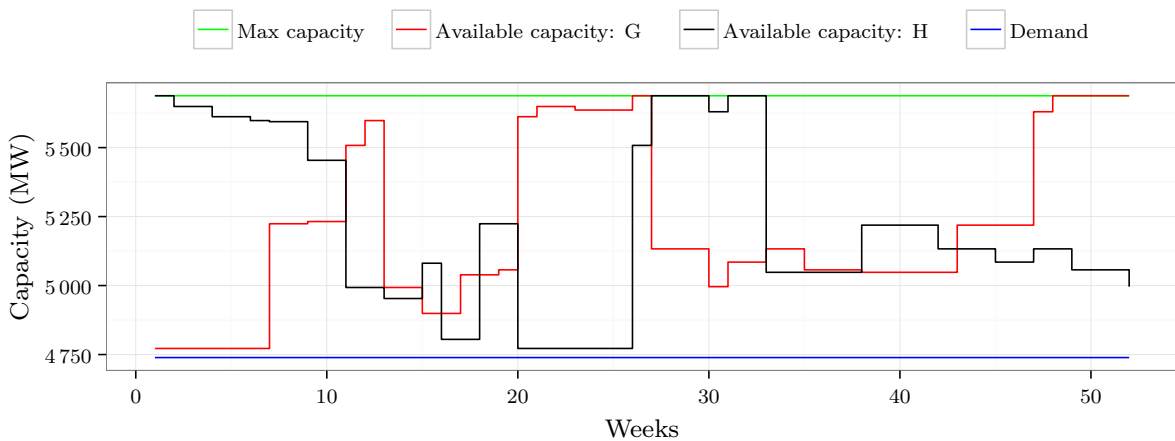
Scenario	Measure of probability of failure	Expected energy production (MW·week)
G	43.542*	180 423
H	51.650	219 717
Percentage change	+15.70%	+17.88%



(a) Two maintenance schedules for the 21-unit test system corresponding to different GMS criteria



(b) The manpower required over the duration of the scheduling window for the two maintenance schedules in Figure 8.5(a)



(c) The available system capacity over the duration of the scheduling window for the two maintenance schedules in Figure 8.5(a)

FIGURE 8.5: Comparison between the maintenance schedules of Scenarios G and H for the 21-unit test system.

As previously mentioned, in order to minimise the probability of unit failure, maintenance schedules typically exhibit maintenance commencement dates that are early during the scheduling window. This is the case in Scenario G where maintenance on PGUs with large capacities is performed early in the scheduling window. This objective function also seeks to schedule PGUs with large failure rates as early as possible so as to decrease the probability of PGU failure. The objective in Scenario H, however, aims to schedule PGUs with large capacities (PGUs that can produce more energy) close to the peaks of the expected energy curves of these PGUs, which are typically close to the end of the scheduling window. It may be seen in Figure 8.5(a) that PGUs with large capacities are typically scheduled later, but still within their respective PGU maintenance windows. It may also be seen in Figure 8.5(a) that, compared to Scenario G, some PGUs with higher failure rates are scheduled for planned maintenance later during the scheduling window in Scenario H. PGU 19 is, for example, scheduled for maintenance during the last week of the scheduling window due to having the highest failure rate of 0.3733 — this is 22 weeks later than in Scenario G. A similar observation may be made for PGU 12, which is scheduled for maintenance 14 weeks later in Scenario H than in Scenario G due to its large failure rate of 0.1527.

The different effects of the two scheduling objectives may also be observed in Figure 8.5(b). In Scenario G, the maximum amount of manpower is often required during the early stages of the scheduling window. It is also observed that towards the end of both halves of the scheduling window (*e.g.* weeks 1–27 and weeks 28–52) no manpower is required as no maintenance is scheduled during these times. In Scenario H, on the other hand, it is observed that towards the end of both halves of the scheduling window (*e.g.* weeks 1–27 and weeks 28–52) the manpower required is at the maximum available number as most of the PGUs exhibit peaks of their expected energy production curves towards the end of the scheduling window.

Similar observations may also be made in respect of Figure 8.5(c). In both Scenarios G and H, the available system capacity drops down to 0.696% above the demand. For Scenario G, this drop is observed during the early stages of the scheduling window whereas for Scenario H the drop is observed during the middle stages of the scheduling window, towards the end of the first half of the scheduling window, due to the maintenance window constraints. The maximum capacity is mainly available just before the end of the first and second halves of the scheduling window in Scenario G. In Scenario H, on the other hand, the maximum available capacity is available during the beginning of the first and second halves of the scheduling window.

Sensitivity analysis

A piecewise linear approximation approach towards solving the nonlinear GMS model of §4.3 for large power systems or very unconstrained systems (in terms of maintenance window constraints) is not expected to be feasible. The feasibility of an exact solution approach by CPLEX is also influenced by the nature of the objective function (*e.g.* linear or nonlinear). It was demonstrated above that employing such a piecewise linear approximation model solution approach in the context of the 21-unit test system is feasible.

In order to analyse the effects of alterations in the system specifications on the feasibility of the piecewise linear approximation model solution approach, six cases are again analysed in this section in terms of the computation times required by CPLEX to solve the nonlinear model of §4.3. These cases are the same as those considered in §7.1 and involve combinations of increasing the peak demand of the system by a certain margin and relaxing the maintenance window constraints to have an earliest starting time of 1 and latest starting time of 53 less the duration of maintenance of each PGU. The first case is the original 21-unit test system which

TABLE 8.3: Various statistics pertaining to optimal solutions to a piecewise linearisation of the nonlinear model of §4.3 in a sensitivity analysis in respect of demand and PGU maintenance windows for the 21-unit test system. The last column contains optimality gap values with respect to provable upper bounds on the maximum objective function value. An asterisk denotes that a time-out budget of 4 hours of computation time was reached by CPLEX.

Cases	Demand (%)	Maintenance window	Objective function value (MW·week)	Time (s)	Gap (%)
1	100	Original	219 717	36	0
2	103	Original	217 862	13	0
3	106.5	Original	215 048	27	0
4	100	Relaxed	260 095*	14 400	0.83
5	103	Relaxed	255 114	5 963	0
6	106.5	Relaxed	243 566*	14 400	6.14

is considered as a reference case for the other five cases. The second case involves a 3% increase in the peak demand, but adheres to the original test system's maintenance window constraints. The third case involves a 6.5% increase in the peak demand, but also adheres to the original maintenance window constraints. In the fourth case, the peak demand is kept as specified for the original 21-unit test system, but the maintenance window constraints are relaxed as described above. The fifth case involves a 3% increase in the peak demand and relaxed maintenance window constraints. Finally, the sixth case involves an increase in the peak demand of 6.5% and relaxed maintenance window constraints. Various statistics pertaining to optimal solutions to a piecewise linearisation of the nonlinear model in §4.3 are shown for these six cases in Table 8.3.

Case 1 requires 36 seconds of computing time by CPLEX to obtain an optimal solution to the piecewise linear approximation model. In Case 2, the demand is increased by 3%, which requires 13 seconds of computing time and yields a 0.844% worsening of the objective function value. It is observed that there is a decrease in computing time and that an optimal solution is still obtainable within a reasonable timeframe. Furthermore, when increasing the demand by 6.5%, in Case 3, a computation time of 27 seconds is required to obtain the optimal solution, which results in a 2.125% worsening of the objective function value compared to that in Case 1. It is observed that increasing the demand has a small impact on the computation time required to solve the nonlinear model of §4.3 (approximately) for the 21-unit test system. This computation time decreases by only 9 seconds over the course of a 6.5% increase in demand. The reason for the decrease in computation time as the demand is increased may be attributed to the smaller solution space through which the algorithm has to search in order to obtain an optimal solution.

When the maintenance window constraints are relaxed, however, the number of feasible solutions to the nonlinear model of §4.3 increases drastically, and so does the required computation time. Case 4, in which the demand is kept as specified for the original 21-unit test system and the maintenance window constraints are relaxed, results in a large increase in computation time. In fact, no optimal solution can be obtained within the time-out budget of 14 400 seconds of processing time. An optimality gap of 0.83% is obtained between the best objective function value (of 260 095) and the smallest provable upper bound on the objective function value within the allowed processing time. A decrease in computation time is observed in Case 5, where the demand is increased by 3% and the maintenance window constraints are relaxed. The first optimal solution is obtained within 5 963 seconds of computation time and yields an objective function value that is 16.11% better than that of Case 1. Finally, in Case 6, it is found that no optimal solution can be obtained within the time-out budget of 14 400 seconds of processing time. A gap of 6.14% is obtained between the best objective function value (of 243 566) and

the smallest provable upper bound on the objective function within four hours of computation time. It is therefore observed that relaxing the maintenance window constraints has a significant impact on the computation time required to solve the nonlinear model of §4.3 (approximately) for the 21-unit test system.

8.1.2 The IEEE-RTS system

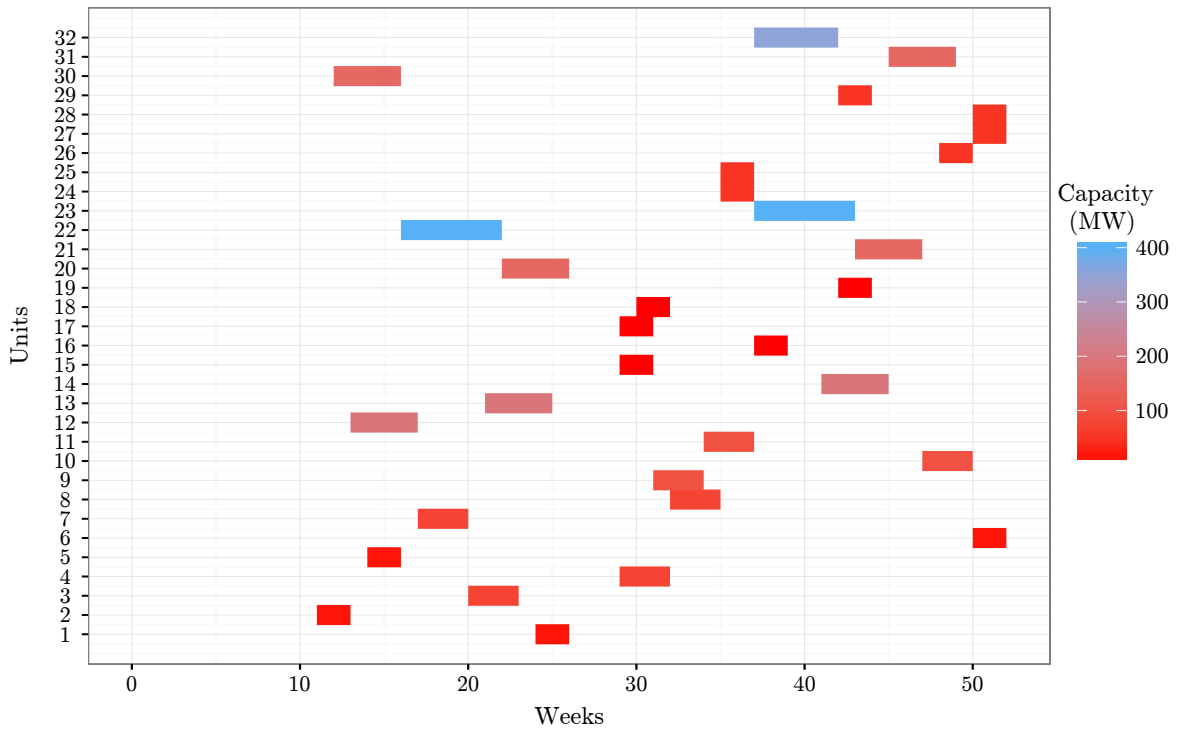
This section contains a presentation of the piecewise linear approximation solution obtained by CPLEX for the nonlinear model of §4.3 as applied to the IEEE-RTS [188]. The solution is contrasted with a solution obtained in the literature upon adoption of another reliability-related scheduling criterion (minimisation of the SSR). The solution is also compared with the solution reported in §7.1.2 when adopting the minimisation of the probability of unit failure objective function. The feasibility of a piecewise linear approximation solution approach for the nonlinear model of §4.3 within the context of the IEEE-RTS is finally also assessed in the form of a sensitivity analysis involving various relaxations of the demand and maintenance scheduling window constraints under an allowable computation time budget of 24 400s (8 hours).

Numerical results

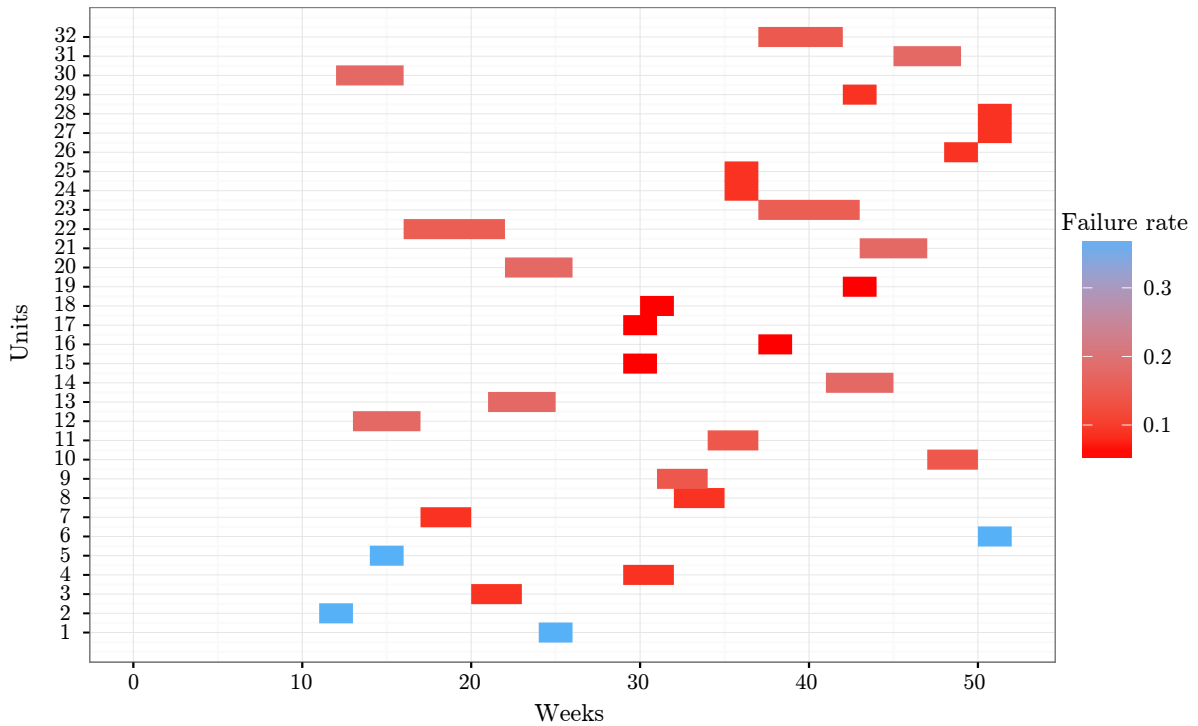
A piecewise linear approximation solution to the nonlinear GMS model of §4.3 is obtained for the IEEE-RTS by CPLEX within 93 538 seconds (25.983 hours). The optimal decision variable values of this solution are given in integer decision vector form by $\mathbf{x} = [25, 12, 21, 30, 15, 51, 18, 33, 32, 48, 35, 14, 22, 42, 30, 38, 30, 31, 43, 23, 44, 17, 38, 36, 36, 49, 51, 51, 43, 13, 46, 38]$, which corresponds to an objective function value of 130 630 MW-week (21 945 840 MWh). A graphical representation of the maintenance schedule is presented in Figure 8.6 with the colour scale in Figure 8.6(a) indicating the rated capacity (in MW) of each PGU and the colour scale in Figure 8.6(b) indicating the failure rate of each PGU. The manpower required over the duration of the scheduling window in order to implement the optimal solution in Figure 8.6 is shown in Figure 8.7(a). The available system capacity over the duration of the scheduling window associated with the optimal solution in Figure 8.6 is shown in Figure 8.7(b).

Figures 8.6 and 8.7 may be used to analyse the optimal solution obtained to the nonlinear GMS model of §4.3 obtained by the piecewise linear approximation solution approach for the IEEE-RTS. The influence that the rated capacity of a PGU has on its scheduled starting time is clear in Figure 8.6(b). PGUs with large rated capacities (such as PGUs 23 and 32) are scheduled to start as close as possible to the peaks of their expected energy curves. This is also the case for PGU 22, but as this PGU has a maintenance window that ends at planning period 27, it therefore must be scheduled for maintenance starting no later than planning period 27. From these examples it is concluded that good solutions to the nonlinear GMS model of §4.3 will typically aim to schedule maintenance on PGUs with large rated capacities as close as possible to the dates at which they are expected to produce the most energy (*e.g.* the peaks of their expected energy production curves). In terms of the failure rate of each PGU in the IEEE-RTS, it is observed that PGUs with large failure rates are either scheduled for maintenance early or late, as may be seen in Figure 8.6(b). The reason for this might be that PGUs with large failure rates have nearly flat expected energy curves. Therefore, the effect on the expected energy produced of moving a PGU with a large failure rate from one maintenance commencement date to another is small.

The manpower requirement for the maintenance schedule presented in Figure 8.6 is shown in Figure 8.7(a). It is observed that during the first 12 weeks of the maintenance schedule no

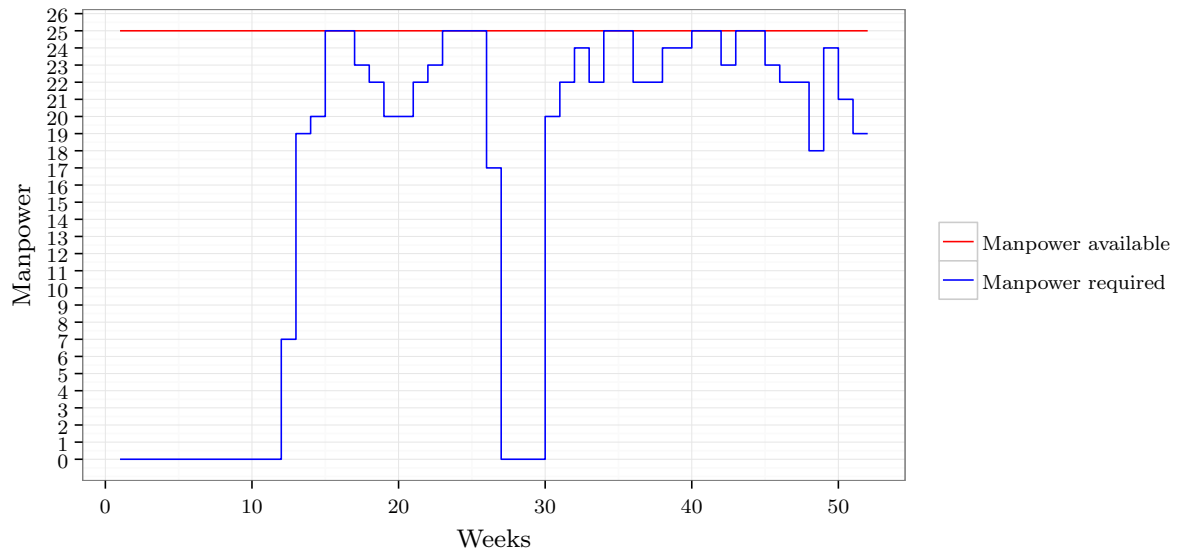


(a) An optimal maintenance schedule obtained by piecewise linear approximation of the nonlinear GMS model objective, with the colour scale indicating the capacities of the PGUs in the IEEE-RTS

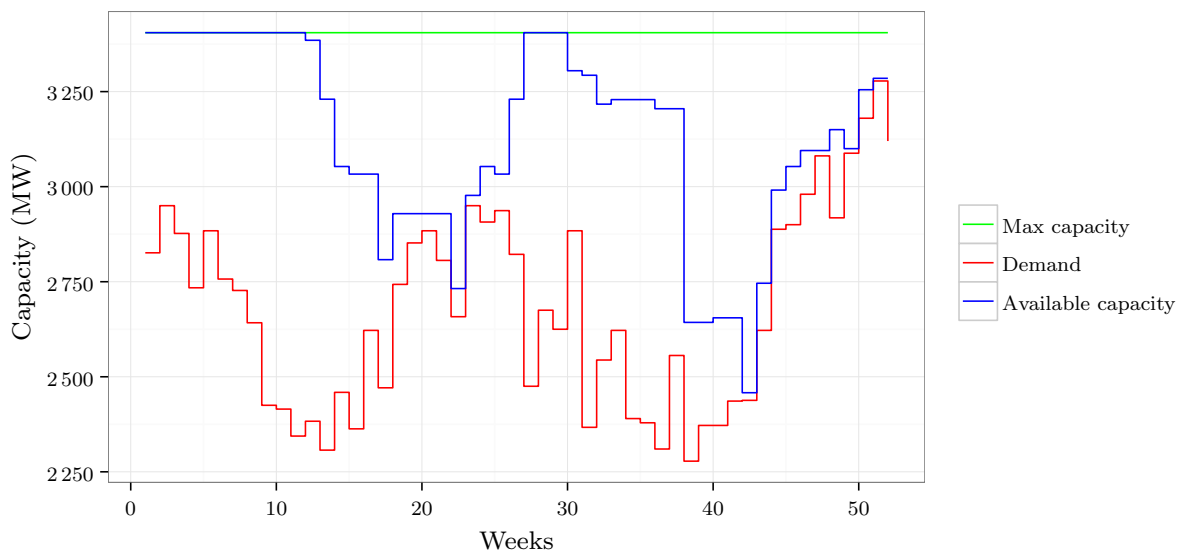


(b) An optimal maintenance schedule obtained by piecewise linear approximation of the nonlinear GMS model objective, with the colour scale indicating the failure rates of the PGUs in the IEEE-RTS

FIGURE 8.6: An optimal solution to a piecewise linearisation of the nonlinear model of §4.3 for the IEEE-RTS.



(a) The manpower required over the scheduling window corresponding to the optimal maintenance schedule in Figure 8.6



(b) The system capacity over the scheduling window corresponding to the optimal maintenance schedule in Figure 8.6

FIGURE 8.7: Evaluation of the manpower required and the system capacity available over the duration of the scheduling window for the IEEE-RTS.

manpower is required as no maintenance is scheduled during this time period. Thereafter, the manpower required increases sharply and is at a maximum for a number of time periods towards the end of the first half of the scheduling window. It is then observed that for 3 weeks after planning period 27, no manpower is again required, whereafter the manpower once again increases and is at a maximum during a number of planning periods for the remainder of the scheduling window. The reason for the maximum manpower required during the latter stages of the scheduling window is that most PGUs exhibit peaks in their expected energy curves towards the end of the scheduling window. Maintenance is therefore scheduled as close as possible to these peaks.

In Figure 8.7(b), the available capacity is observed to be at a maximum at the beginning of the scheduling window, but diminishes towards planning period 27, which is typically when the second set of maintenance window constraints starts. After planning period 27, the available capacity is again at a maximum, whereafter it diminishes towards the end of the scheduling window. Towards the end of the scheduling window, it is also observed that the difference between the energy demand and available system capacity is very small.

Comparison with results from the literature

The effects of adopting a scheduling criterion which seeks to maximise the expected energy production, as in the newly proposed objective function of §4.3.2, may be better analysed by comparing the numerical results reported above with results found in the literature when adopting other reliability-related scheduling criteria, such as minimisation of the SSR. The numerical results reported above are therefore compared in this section with the results obtained by Schlünz and van Vuuren [188], who adopted the SSR scheduling criterion in (2.8). The results obtained by Schlünz and Van Vuuren [188] (referred to here as Scenario I) are compared with the results reported above (referred to here as Scenario J). Both GMS objectives adopted in Scenarios I and J reside within the class of the reliability scheduling criteria and the results of Schlünz and Van Vuuren [188] for the IEEE-RTS (with the minimisation of SSR as objective) represent the best results available in the literature for this particular scheduling criterion and problem instance combination. A graphical representation of the two maintenance schedules are shown in Figure 8.9 and their corresponding effects on the manpower required and the available system capacity for the IEEE-RTS are shown in Figures 8.9(a) and 8.9(b), respectively. In Figure 8.8, the maintenance schedules of the two scenarios are compared, with the colour scale indicating the rated capacity of the PGUs in the system.

A comparison between the results for the two scenarios in terms of both scheduling objectives

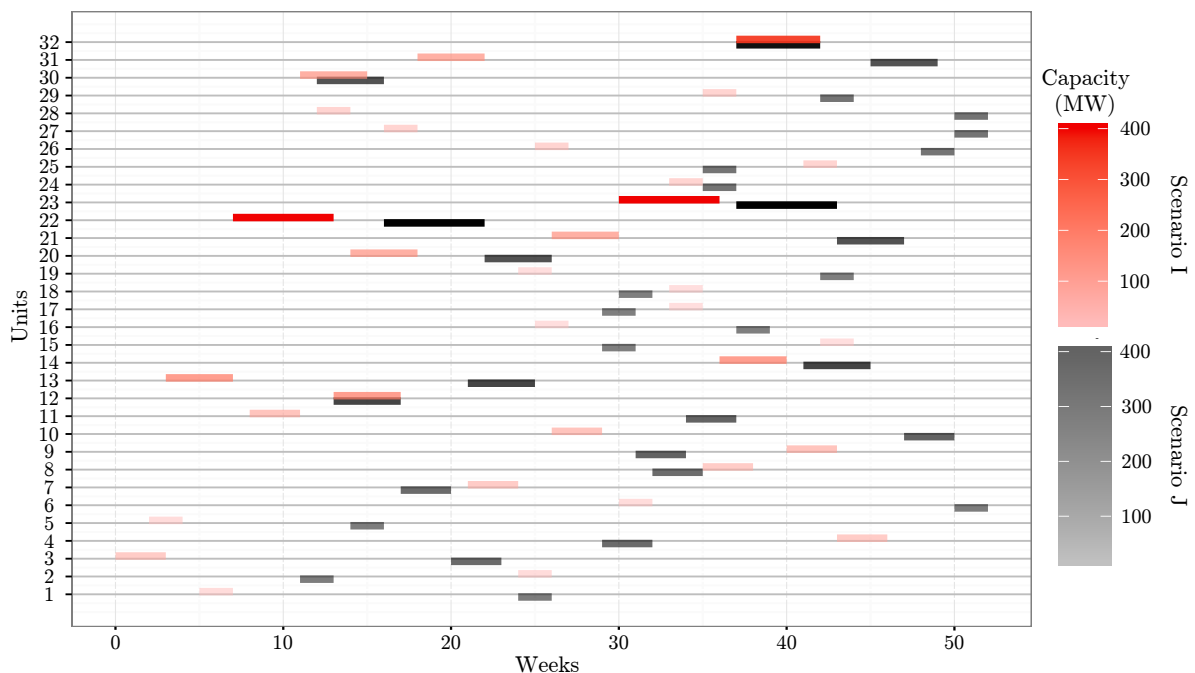
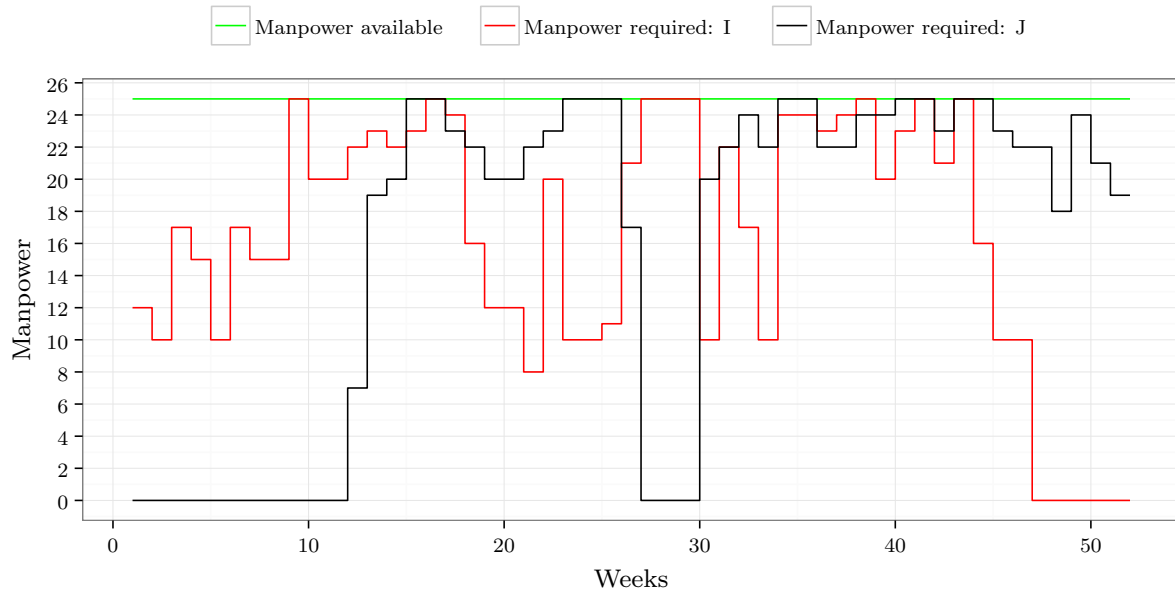
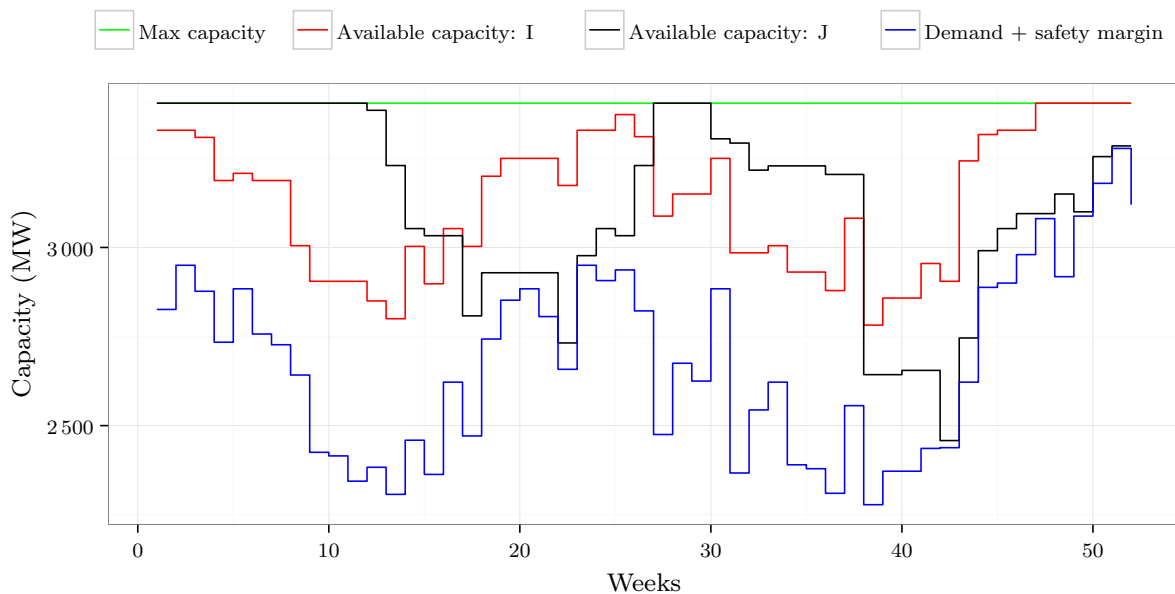


FIGURE 8.8: Comparison between the maintenance schedules of Scenario I and J for the IEEE-RTS.



(a) The manpower required over the duration of the scheduling window for the two maintenance schedules in Figure 8.8



(b) The available system capacity over the duration of the scheduling window for the two maintenance schedules in Figure 8.8

FIGURE 8.9: Comparison between the manpower required and the available system capacity of Scenarios I and J for the IEEE-RTS.

is presented in Table 8.4. The optimal solution of Scenario J performs 19.67% worse than that of Scenario I in terms of the SSR scheduling objective. A similar observation is made when considering the best solution obtained for the maximisation of the expected energy scheduling criterion. In this case, the optimal solution obtained for the objective of Scenario J performs 12.31% better than that of Scenario I. The reason for this is again, of course, that the two objectives conflict in terms of when maintenance should be scheduled for the PGUs. In Scenario I, the objective function aims to spread the maintenance of PGUs with large capacities out over

TABLE 8.4: Comparison between the objective function values associated with the maintenance schedules in Figure 8.8 for Scenarios I and J in the context of the IEEE-RTS. The percentage change values are computed for the solution of Scenario J relative to that of Scenario I.

Scenario	SSR (MW) ²	Expected energy production (MW·week)
I	33 627 292	104 938
J	38 348 738	130 630
Percentage change	+12.31%	+19.67%

the entire scheduling window in order to levelise the reserves of the system, as mentioned. The objective function of Scenario J, on the other hand, aims to schedule planned maintenance for the PGUs towards the end of the scheduling window as close as possible to the peaks of the PGUs' expected energy production curves.

As previously mentioned, in order to minimise the SSR, maintenance schedules typically exhibit maintenance commencement dates that are spread out over the entire scheduling window. This is indeed the case in Scenario I where maintenance is performed on PGUs with large capacities throughout the scheduling window. The objective in Scenario J, however, aims to schedule PGUs with large capacities close to the peak of the expected energy curves of these PGUs, which is typically close to the end of the scheduling window. It may be seen in Figure 8.8 that PGUs with large capacities are typically scheduled later, but still within their respective PGU maintenance windows. It may also be seen in Figure 8.8 that, compared to Scenario I, some PGUs with larger failure rates are either scheduled for planned maintenance early or late during the scheduling window in Scenario J. PGU 2, for example, is scheduled for maintenance during the early stages of the scheduling window due to having the largest failure rate of 0.3733, which is 13 weeks earlier than in Scenario I. On the other hand, it is observed for PGU 6, which also has a failure rate of 0.3733, that maintenance is scheduled 20 weeks later in Scenario J than in Scenario I. Maintenance is either scheduled during the early stages of the scheduling window or during the later stages of this window for PGUs with high failure rates.

The different effects of the two scheduling objectives on manpower required and available system capacity may also be observed in Figure 8.9(a). In Scenario I, the manpower required for planned maintenance is spread out over the entire scheduling window. During each planning period of the problem instance, manpower is required as there is no planning period within the scheduling window during which no PGUs are in maintenance until week 47. In Scenario J, on the other hand, it is observed that at the end of both halves of the scheduling window (*e.g.* weeks 1–27 and weeks 28–52) the manpower required is at the maximum available number as most of the PGUs exhibit peaks in their expected energy production curves towards the end of the scheduling window.

Similar observations may also be made in respect of Figure 8.9(b). In Scenario I, the available system capacity never falls below a 6% band above the demand. In Scenario J, on the other hand, the available system capacity drops down to 0.214% above the demand during week 51 of the scheduling window as many PGUs are scheduled for maintenance towards the end of the scheduling window due to the expected energy curves peaking towards the end of the scheduling window. The maximum available system capacity mainly occurs towards the middle of the graph for both Scenario I and J, whereas fewer jumps are observed in the available capacity for Scenario I than for Scenario J.

Comparison with results obtained by minimising probability of unit failure

The effects of the two novel scheduling criteria proposed in this dissertation (minimising the probability of unit failure and maximisation of the expected energy production) may again better be analysed by comparing the results obtained for these scheduling criteria with one another. The results obtained when minimising the probability of unit failure (referred to here as Scenario K), reported in §7.1.2, are compared with the results obtained when maximising the expected energy production (referred to here as Scenario L), reported above. A graphical representation of the two maintenance schedules is shown in Figure 8.10 with their effects on the manpower required and the available system capacity shown in Figures 8.11(a) and 8.11(b), respectively.

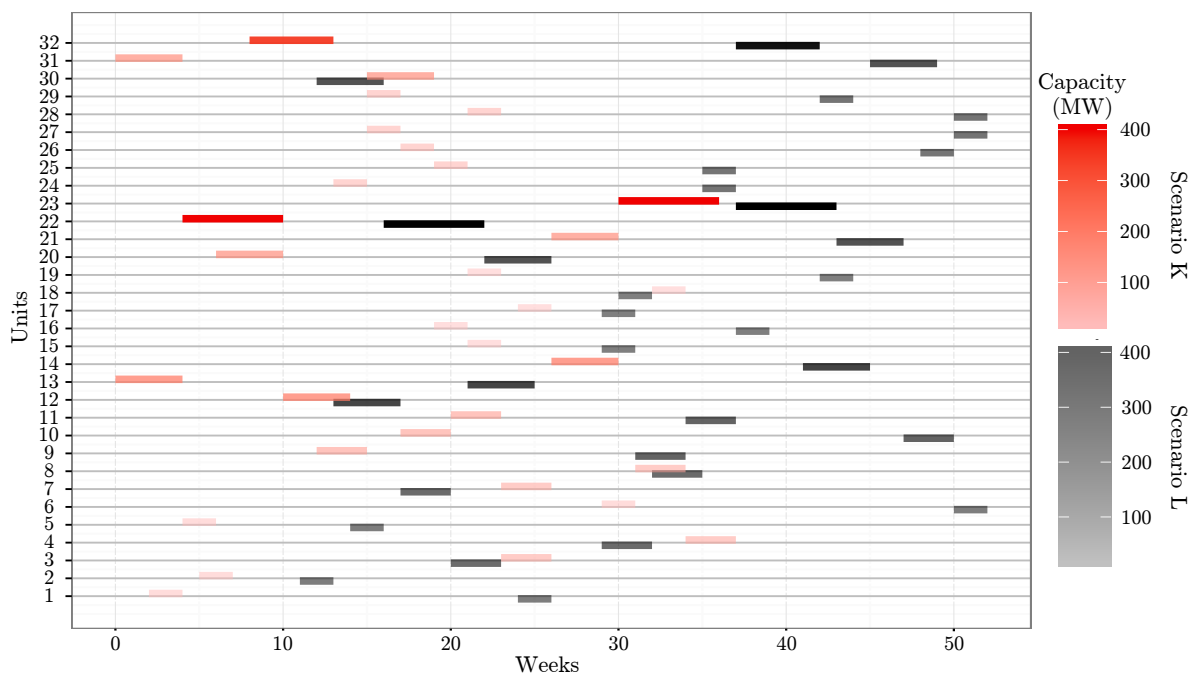
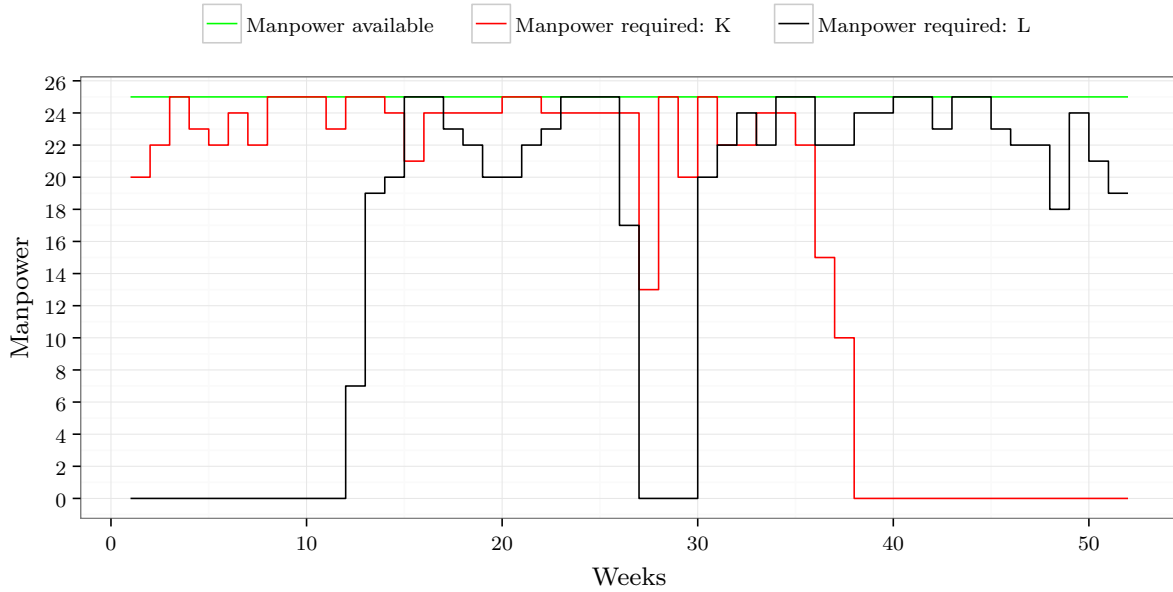


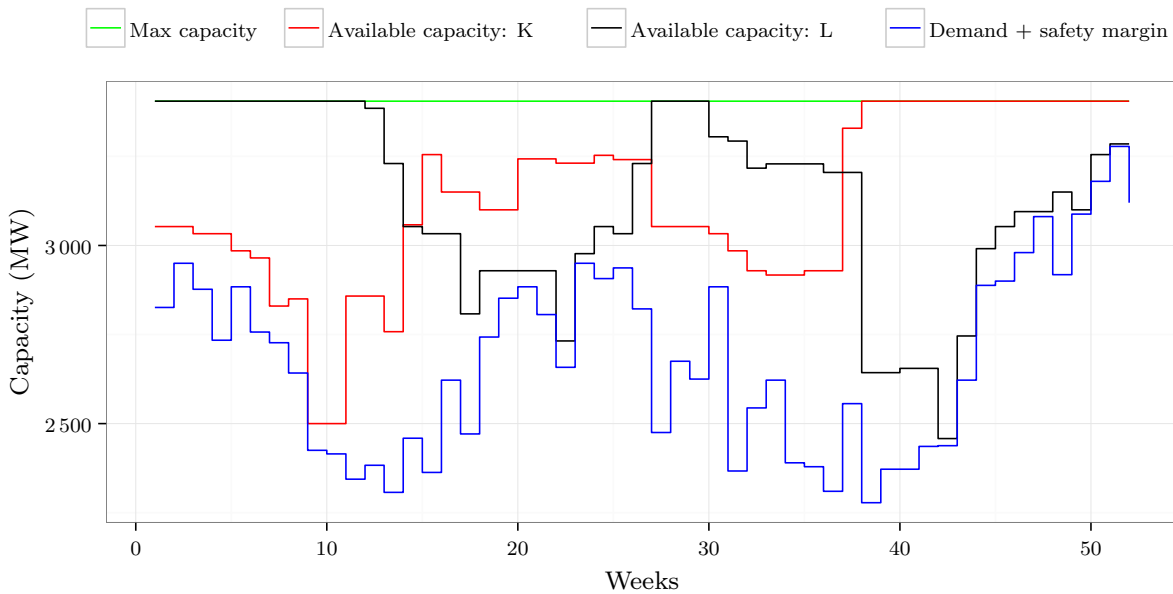
FIGURE 8.10: Comparison between the maintenance schedules of Scenarios K and L for the IEEE-RTS.

A comparison between the results for the two scenarios in terms of both scheduling objectives is presented in Table 8.5. The optimal solution of Scenario L performs 25.67% worse than that of Scenario K in terms of the minimisation of the probability of unit failure scheduling objective. A similar observation is made when considering the best solution obtained for the maximisation of the expected energy scheduling criterion. In this case, the solution obtained for the objective of Scenario L performs 34.75% better than that of Scenario K. This difference is quite substantial.

As previously mentioned, in order to minimise the probability of unit failure, maintenance schedules typically exhibit maintenance commencement dates that are early during the scheduling window. This is certainly the case in Scenario K where maintenance on PGUs with large capacities is performed early during the scheduling window. This objective function also seeks to schedule PGUs with large failure rates as early as possible so as to decrease their probability of PGU failure. The objective in Scenario L, however, aims to schedule PGUs with large rated capacities close to the peaks of their expected energy curves, which are typically close to the end of the scheduling window. It may be seen in Figure 8.10 that PGUs with large rated capacities are typically scheduled later in Scenario L than in Scenario K, but still within their respective PGU maintenance windows. It may also be seen in Figure 8.10 that, compared to Scenario K,



(a) The manpower required over the duration of the scheduling window for the two maintenance schedules in Figure 8.10



(b) The available system capacity over the duration of the scheduling window for the two maintenance schedules in Figure 8.10

FIGURE 8.11: Comparison between the manpower required and the available system capacity of Scenarios K and L for the IEEE-RTS.

some PGUs with larger failure rates are scheduled for planned maintenance later during the scheduling window in Scenario L. PGU 6, for example, is scheduled for maintenance during the second last week of the scheduling window due to exhibiting the largest failure rate of 0.3733, which is 21 weeks later than in Scenario K. A similar observation may be made for PGU 31 mainly due to its 0.1750 failure rate (this PGU is scheduled for maintenance 45 weeks later, which is also the largest difference between the maintenance dates of a single PGU of the two scenarios).

TABLE 8.5: Comparison between the objective function values associated with the maintenance schedules in Figure 8.10 for Scenarios K and L in the context of the IEEE-RTS. The asterisk indicates an optimal solution. The percentage change values are computed for the solution of Scenario L relative to that of Scenario K.

Scenario	Measure of probability of failure	Expected energy production (MW·week)
K	57.886*	85 241
L	77.872	130 630
Percentage change	+25.67%	+34.75%

The different effects of the two scheduling objectives may also be observed in Figure 8.11(a). In Scenario K, the maximum amount of manpower is often required during the early stages of the scheduling window. It is observed that the required manpower decreases gradually, on average, as the end of the scheduling window approaches, until no manpower is required after week 38. In Scenario L, on the other hand, it is observed that at the end of both halves of the scheduling window (*e.g.* weeks 1–27 and weeks 28–52) the manpower required is at the maximum available number as most of the PGUs exhibit peaks in their expected energy production curves towards the end of the scheduling window. It is observed that the effects that these two scheduling criteria have on the manpower required are almost the exact opposites of one another.

Similar observations may also be made in respect of Figure 8.11(b). In both Scenarios K and L, major drops are observed in the available system capacity. In Scenario K, this drop is observed during the early stages of the scheduling window whereas in Scenario L the drop is observed towards the end of the scheduling window due to the peaks of the expected energy production curves of PGUs being towards the end of the scheduling window. The maximum system capacity is available from week 38 onwards as no maintenance is scheduled after this planning period in Scenario K. In Scenario L, on the other hand, the maximum available capacity occurs during the beginning of the first and second halves of the scheduling window as no maintenance is scheduled during these planning periods. It is observed, as was the case for the manpower required, that the effects of these two scheduling criteria on the available capacity are almost the exact opposites of one another.

Sensitivity analysis

A piecewise linear approximation approach towards solving the nonlinear GMS model of §4.3 for large power systems or very unconstrained systems is again not expected to be feasible. It was demonstrated above that employing such a piecewise linear approximation model solution approach in the context of the IEEE-RTS is not feasible within 8 hours of computation time as CPLEX required 25.983 hours to solve the model.

In order to analyse the effects of alterations in the system specifications on the piecewise linear approximation model solution approach, four cases are analysed in this section in terms of the computation time required by CPLEX to solve the nonlinear model of §4.3 in the context of the IEEE-RTS. These cases involve combinations of increasing the peak demand of the system by a certain margin and relaxing the maintenance window constraints to have an earliest starting time of 1 and latest starting time of 53 less the duration of maintenance of each PGU. The first case is the original IEEE-RTS, which is considered as a reference case for the other three cases. The second case involves a 3% increase in the peak demand, but adheres to the original test system's maintenance window constraints. In the third case, the peak demand is kept as specified for the

original IEEE-RTS, but the maintenance window constraints are relaxed as described above. Finally, the fourth case involves a 3% increase in the peak demand and relaxed maintenance window constraints. Various statistics pertaining to solutions of the nonlinear model in §4.3 are shown for these four cases in Table 8.6.

TABLE 8.6: Various statistics pertaining to optimal solutions to a piecewise linearisation of the nonlinear model of §4.3 in a sensitivity analysis in respect of demand and PGU maintenance windows for the IEEE-RTS. The last column contains gap values with respect to provable lower bounds on the minimum objective function value. An asterisk denotes that a time-out budget of 8 hours of computation time was reached by CPLEX.

Cases	Demand (%)	Maintenance window	Objective function value (MW·week)	Time (s)	Gap (%)
1	100	Original	130 625	28 800	0.30
2	103	Original	128 590	28 800	0.45
3	100	Relaxed	142 169	28 800	3.16
4	103	Relaxed	138 633	28 800	3.20

In Case 1 it is found that no optimal solution can be obtained within the time-out budget of 28 800 seconds of computation time for the piecewise linearised model. The optimality gap achieved within the allowed budget of computation time for Case 1 was 0.3%. In Case 2, the demand is increased by 3%, which once again does not return an optimal solution as the maximum allowed computation time was reached. The best solution obtained yields a 1.558% worsening of the objective function value at a 0.45% optimality gap. Even with a smaller solution space to evaluate in Case 2, where the demand is increased by 3%, it is observed that the maximum computation time is again reached and the optimality gap also increases.

When the maintenance window constraints are relaxed, however, the number of optimal solutions to the piecewise linearisation of nonlinear model of §4.3 increases drastically, and so does the required computation time. Case 3, in which the demand is kept as specified for the original IEEE-RTS and the maintenance window constraints are relaxed, results in a large increase in the achievable optimality gap. Once again, no optimal solution can be obtained within the time-out budget of 28 800 seconds of computation time. An optimality gap of 3.16% is obtained between the best objective function value and the smallest provable upper bound on the objective function value within the allowed processing time. This results in a 90.51% increase in the optimality gap relative to Case 1. Finally, as was observed for the previous three cases, no optimal solution is achievable within the time-out budget of 28 800 seconds of computation time. In Case 4, the demand is increased by 3% and the maintenance window constraints are also relaxed. The best solution obtained within the allowed computation time exhibits an objective function value of 138 633 with an optimality gap of 3.20% between this objective function value and the smallest provable upper bound on the objective function value within the allowed processing time.

It is observed in all of the four cases compared above, that an exact solution approach employing a piecewise model linearisation approach is not feasible in the allowed computation time of 28 800 seconds. In all of the cases this time-out budget was reached and no provably optimal solution was obtained. Large optimality gaps were also observed in the cases where the maintenance window constraints were relaxed.

8.2 Metaheuristic approximate solution approach

In this section, parameter optimisation experiments in the form of an experimental design are carried out in order to establish suitable parameter values for the SA algorithm in the context of the nonlinear model of §4.3 as applied to the 21-unit test system [54] and the 32-unit IEEE-RTS [188] described in §6. Thereafter, a presentation follows of the results obtained by the SA algorithm for these two test systems when adopting the best parameter value combination uncovered during the parameter optimisation experiments. The approximate solutions thus obtained are finally compared with the results obtained by the exact solution approach, as described in §8.1. The SA solution approach described in §5.3 was implemented in the software package R [175] in combination with RStudio [184] as the *integrated development environment* (IDE) for R in order to compute approximate solutions to the nonlinear model of §4.3.

8.2.1 The 21-unit system

This section is devoted to an application of the parameter optimisation experiment described in §5.3.3 within the context of the 21-unit test system. The best algorithmic parameter combination thus uncovered is then employed to solve the test instance approximately in terms of the maximisation of the energy production scheduling criterion.

Parameter optimisation experiment

In this section, a parameter optimisation experiment is performed for the 21-unit test system in two separate phases according to the design described in §5.3.3. The first phase of the experiment involves variation of the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ . During the second phase of the experiment, the parameters varied are the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ .

Phase 1: Initial acceptance ratio & soft constraint violation severity factor

The mean optimality gaps (measured as percentages relative to the optimal objective function values obtained by CPLEX for the piecewise linear approximation in §8.1.1) associated with the feasible incumbents returned during the first phase of the parameter optimisation experiment are shown in Table 8.7. The mean computation times involved in evaluating combinations of these parameter values are shown in Table 8.8 (which includes computation times expended during runs that returned infeasible incumbents). The numbers of times (out of 30) that an infeasible incumbent was returned during an SA search run are finally shown in Table 8.9.

In these results, a correlation is observed between the mean optimality gap and the soft constraint violation severity factor. As the severity factor increases, it is observed that the mean optimality gap also increases. This may also be seen in Figure 8.12(a) where a box plot comparison is presented of the mean optimality gap as a function of the soft constraint violation severity factor. As mentioned in §7.2.1, the reason for this correlation may be that the search space contains many disjoint pockets of feasible regions. Therefore, in order to move from one such pocket of feasibility to a different pocket of feasibility, infeasible solutions have to be accepted during the transition in view of the fact that the SA algorithm is a trajectory-based metaheuristic. This type of transition is, however, less probable when adopting a larger soft constraint violation severity factor. It is therefore observed that the smallest value of the soft constraint violation

TABLE 8.7: Mean optimality gaps for all parameter combinations of the first phase of the parameter optimisation experiment (involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ) for the 21-unit test system with maximisation of the expected energy production as scheduling criterion.

χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	4.4224	6.1328	6.1065	6.6905	7.3375
0.5	3.7789	4.7433	6.4027	6.6187	6.5350
0.6	4.6215	4.7668	5.7669	6.3012	7.2557
0.7	3.5656	5.8016	6.4235	6.5505	5.6886
0.8	3.5254	5.4698	6.2172	5.8874	6.7177

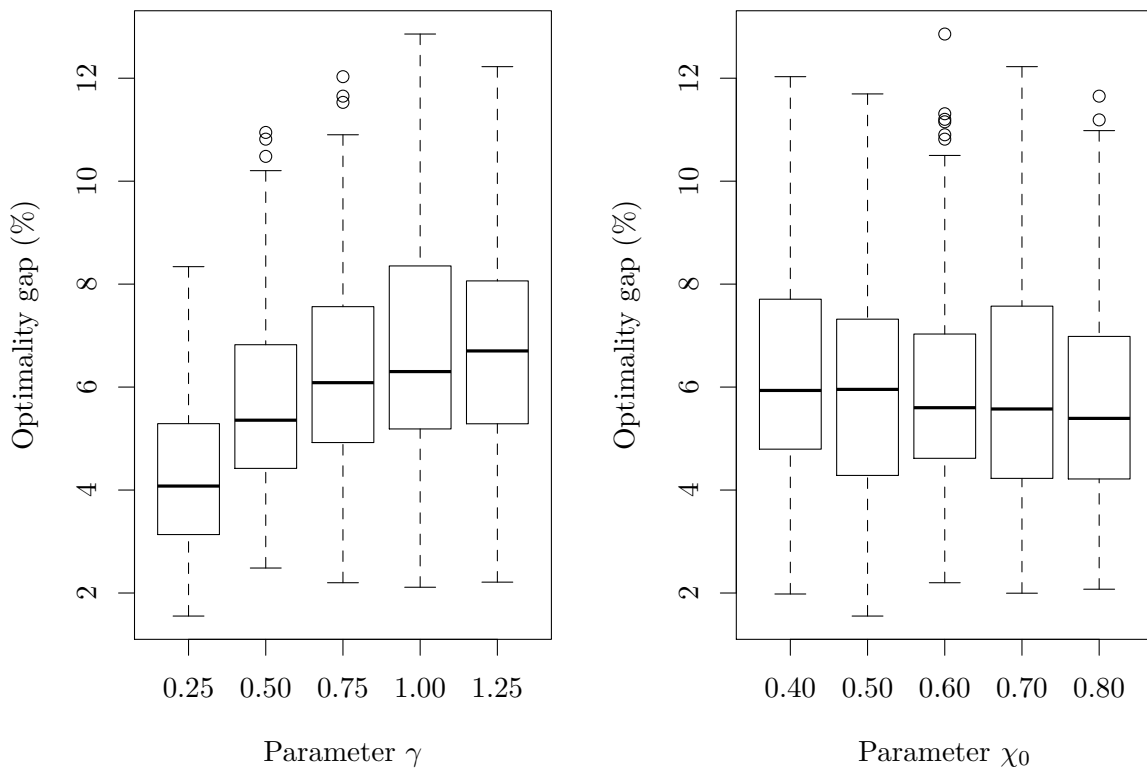
TABLE 8.8: Mean computation times required for all parameter combinations of the first phase of the parameter optimisation experiment (involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ) for the 21-unit test system with maximisation of expected energy production as scheduling criterion.

χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	3 616	2 074	2 765	2 564	2 531
0.5	2 814	2 812	3 681	2 188	3 065
0.6	1 979	1 855	1 914	2 601	1 660
0.7	2 258	1 874	2 225	3 746	2 876
0.8	2 846	2 422	2 085	1 910	2 311

severity factor leads to the discovery of feasible incumbents during the SA search that are closer to optimal. It is, however, also observed that this parameter combination results in the smallest spread in values. The effect of the initial acceptance ratio on the mean optimality gap seems to be less prominent. Almost no increase or decrease is observed in the mean optimality gap as the initial acceptance ratio is increased. This is clearly visible in the box plot comparison presented in Figure 8.12(b). It is observed that the spreads of the mean optimality gaps for all five of the initial acceptance ratio values are fairly similar, with the initial acceptance ratio value of 0.6 exhibiting the most outliers. The results obtained in respect of the initial acceptance ratio are similar. This similarity in mean optimality gaps may be due to the fact that the SA algorithm typically terminates as a result of the three-consecutive-epoch termination criterion and not the criterion associated with the maximum allowable computation time. The algorithm therefore has enough time to reach an acceptably good incumbent solution, even when the initial temperature is higher in some cases (as is the case for a higher initial acceptance ratio).

TABLE 8.9: Number of infeasible incumbents (out of 30) returned for all parameter combinations of the first phase of the parameter optimisation experiment (involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ) for the 21-unit test system with maximisation of expected energy production as scheduling criterion.

χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	21 (70.00%)	12 (40.00%)	10 (33.33%)	9 (30.00%)	3 (10.00%)
0.5	24 (80.00%)	9 (30.00%)	5 (16.67%)	6 (20.00%)	3 (10.00%)
0.6	24 (80.00%)	12 (40.00%)	5 (16.67%)	9 (30.00%)	6 (20.00%)
0.7	24 (80.00%)	11 (36.67%)	9 (30.00%)	6 (20.00%)	5 (16.67%)
0.8	22 (73.33%)	10 (33.33%)	7 (23.33%)	4 (13.33%)	4 (13.33%)



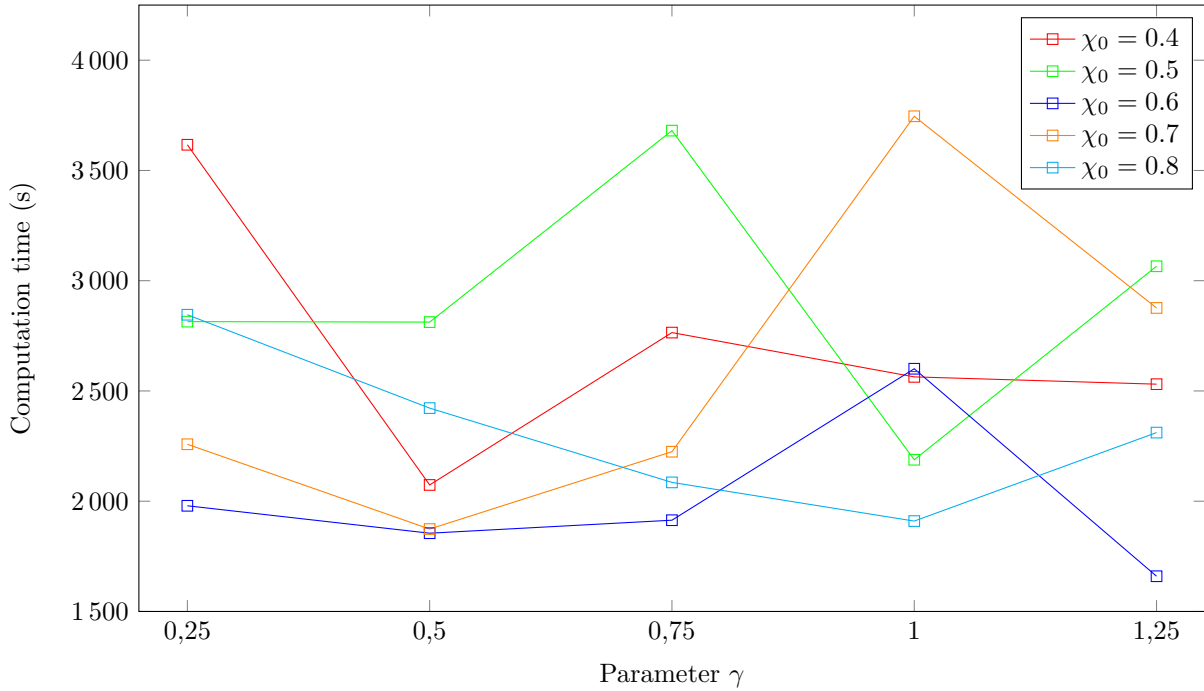
(a) Box plot comparison of the optimality gap as a function of the soft constraint violation severity factor

(b) Box plot comparison of the optimality gap as a function of the initial acceptance ratio

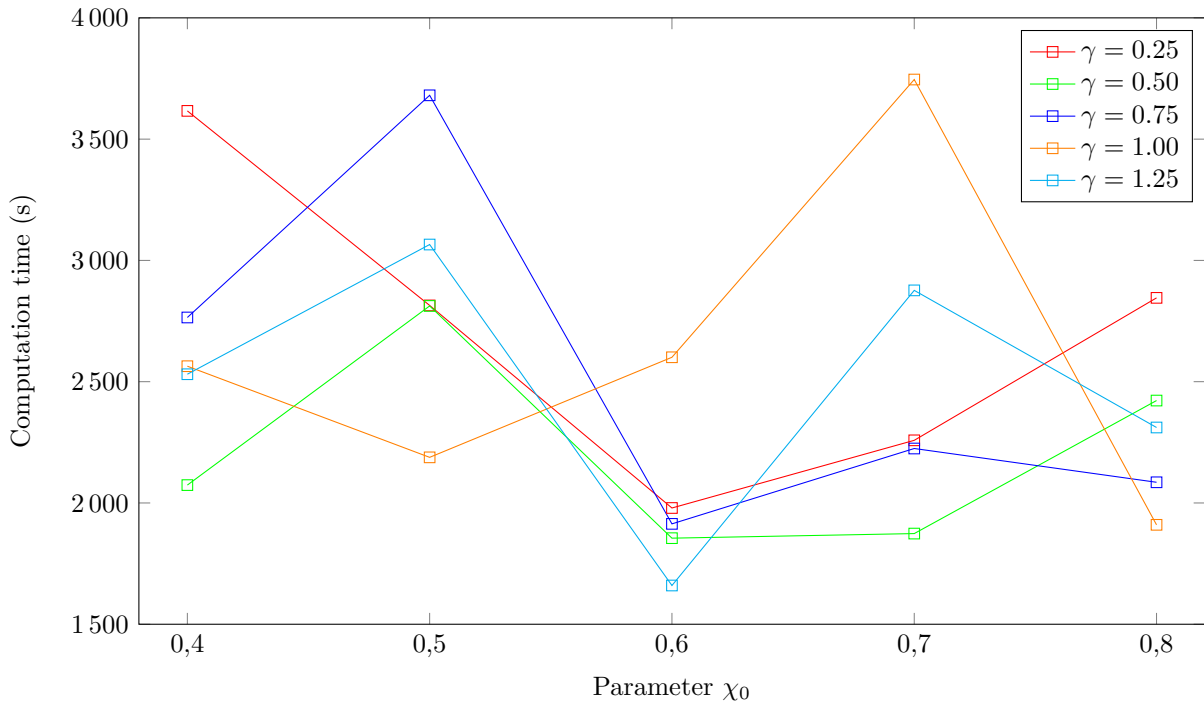
FIGURE 8.12: Box plot comparison of the optimality gaps for the first phase of the parameter optimisation experiment (involving the soft constraint violation severity factor γ and the initial acceptance ratio χ_0) for the 21-unit test system with maximisation of expected energy production as scheduling criterion.

In terms of the computation time expended to evaluate each of the parameter combinations presented in Table 8.8, no clear correlation is observed between this computation time and the soft constraint violation factor. That is, as the soft constraint violation severity factor increases, no clear increase or decrease in the average computation time is observed. This is visible in Figure 8.13(a), which contains a plot of the computation time as a function of the soft constraint violation severity factor for the different initial acceptance ratios. Similarly, for the initial acceptance ratio, no clear pattern is observed in the mean computation time when this parameter is varied, as may be seen in Figure 8.13(b).

It is observed in Table 8.9 that as the soft constraint violation severity factor increases, the number of infeasible incumbents decreases, as expected. A very large proportion of infeasible incumbents are observed for the small soft constraint violation severity factor value of 0.25. It is, however, observed that as the soft constraint violation severity factor increases, the number of infeasible incumbents returned (out of the 30) decreases at a rate that seems to be exponential. This trend is also visible in Figure 8.14(a), where the number of infeasible incumbents returned is presented as a function of the soft constraint violation severity factor for different initial acceptance ratios. For the initial acceptance ratio, however, it is observed, as previously also noted, that no significant change results in the number of infeasible incumbents returned as the value of the initial acceptance ratio varies. This may also be observed in Figure 8.14(b), where the number of infeasible incumbents is presented as a function of the initial acceptance ratio for different soft constraint violation severity factor values.

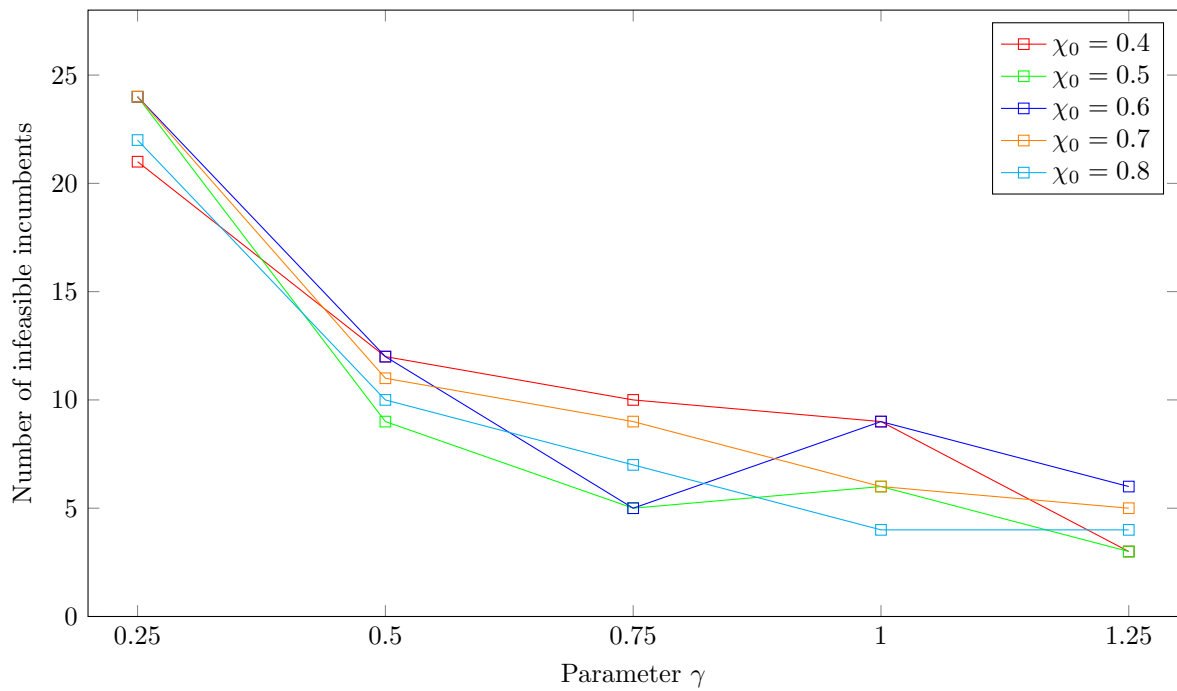


(a) The required computation time as a function of the soft constraint violation severity factor γ for different values of the initial acceptance ratio χ_0

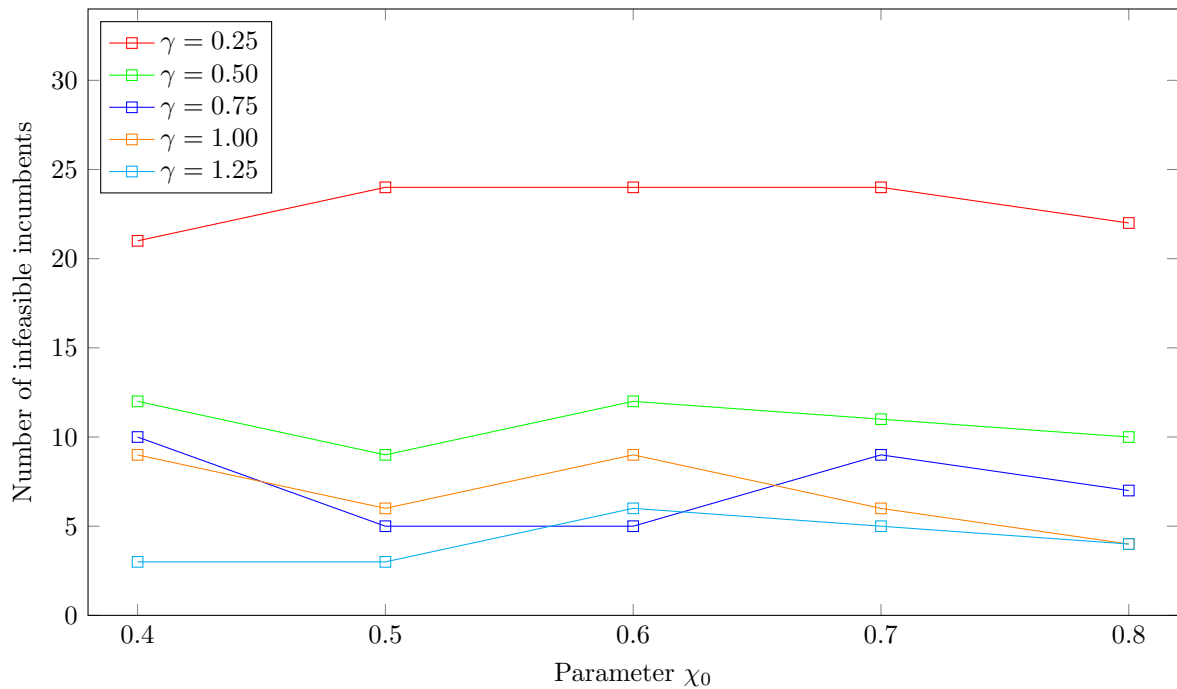


(b) The required computation time as a function of the initial acceptance ratio χ_0 for different values of the soft constraint violation severity factor γ

FIGURE 8.13: The required computation time as functions of the soft constraint violation severity factor γ and the initial acceptance ratio χ_0 when solving the 21-unit test system with maximisation of expected energy production as scheduling criterion.



(a) The number of infeasible incumbents returned (out of 30) as a function of the soft constraint violation severity factor γ for different values of the initial acceptance ratio χ_0



(b) The number of infeasible incumbents returned (out of 30) as a function of the initial acceptance ratio χ_0 for different values of the soft constraint violation severity factor γ

FIGURE 8.14: The number of infeasible incumbents returned (out of 30) as functions of the soft constraint violation severity factor γ and the initial acceptance ratio χ_0 when solving the 21-unit test system with maximisation of expected energy production as scheduling criterion.

From the results obtained during the first phase of the parameter optimisation experiment, which included variation of the soft constraint violation severity factor and the initial acceptance ratio, a value of 0.75 is selected for the soft constraint violation severity factor and a value of 0.6 is selected for the initial acceptance ratio, as shown in Table 8.10. For this combination of parameter values, an acceptable mean optimality gap of 5.7669% is obtained. As no trend was observed in the computation time as the parameters were varied, the computation time did not play a role in the final decision of the parameter combination. A mean computation time of 1914 seconds required to solve the 21-unit test system is, however, also acceptable as it is lower than the time-out budget allowed for the SA search, namely 14400 seconds. A small proportion of infeasible incumbents is also returned (out of the 30 runs) for this parameter value combination. The combination is therefore adopted as the final parameter values in the context of solving the 21-unit test system with maximisation of expected energy production as scheduling criterion. These values for the soft constraint violation severity factor and the initial acceptance ratio are also adopted during the second phase of the parameter optimisation experiment involving variation of the cooling parameter, the reheating parameter and the epoch parameter values.

TABLE 8.10: Selected parameter combination for the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ , as obtained from the first phase of the parameter optimisation experiment for the 21-unit test system with maximisation of expected energy production as scheduling criterion.

χ_0	γ	Optimality gap (%)	Computation time (s)	Infeasibilities
0.6	0.75	5.767	1914	5 (out of 30)

Phase 2: Cooling parameter, reheating parameter, epoch parameter

The mean optimality gaps (again measured as a percentage relative to the optimal objective function value obtained by CPLEX) associated with the piecewise linear approximation in §8.1.1 for the feasible incumbents returned during the second phase of the parameter optimisation experiment, are shown in Table 8.11, while the mean computation times required for the evaluation of the combinations of these parameter values are shown in Table 8.12. These times again include computation times expended during runs that returned infeasible incumbents. The numbers of times (out of 30) that an infeasible incumbent was returned during an SA search run are shown in Table 8.13. Furthermore, box plot comparisons of the mean optimality gaps are presented as functions of the cooling parameter, the reheating parameter and the epoch parameter in Figures 8.15(a), 8.15(b) and 8.15(c), respectively.

From these results, it may be observed that there is a correlation between the cooling parameter and average optimality gap. As the cooling parameter increases from 0.85 to 0.95, the average optimality gap decreases. This is observed in Table 8.11 — as one moves down in the table, slight differences in the average optimality gaps are observed. This may also be seen in Figure 8.15(a) where it is observed that the average optimality gaps decrease as a function of the cooling parameter. The mean optimality gaps corresponding to the cooling parameter values of 0.85, 0.90 and 0.95 are 7.057%, 6.568% and 5.939%, respectively. From Figure 8.15(a) it is also observed that as the cooling parameter increases, the spreads of the optimality gaps decrease slightly.

The correlation between the reheating parameter and the mean optimality gap is much more prominent than that between the mean optimality gap and the cooling parameter. As the reheating parameter increases in Table 8.11 (*i.e.* moving from left to right in the table), an increase in the mean optimality gap is observed. This is also observed in the box plot in Figure 8.15(b), which contains a comparison of the mean optimality gaps for the three different reheating parameter values of 0.55, 0.75 and 0.95. Here a clear increase in the mean optimality

TABLE 8.11: Mean optimality gaps for all the combinations of parameter values during of the second phase of the parameter optimisation experiment (involving variation of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ) for the 21-unit test system with maximisation of expected energy production as scheduling criterion.

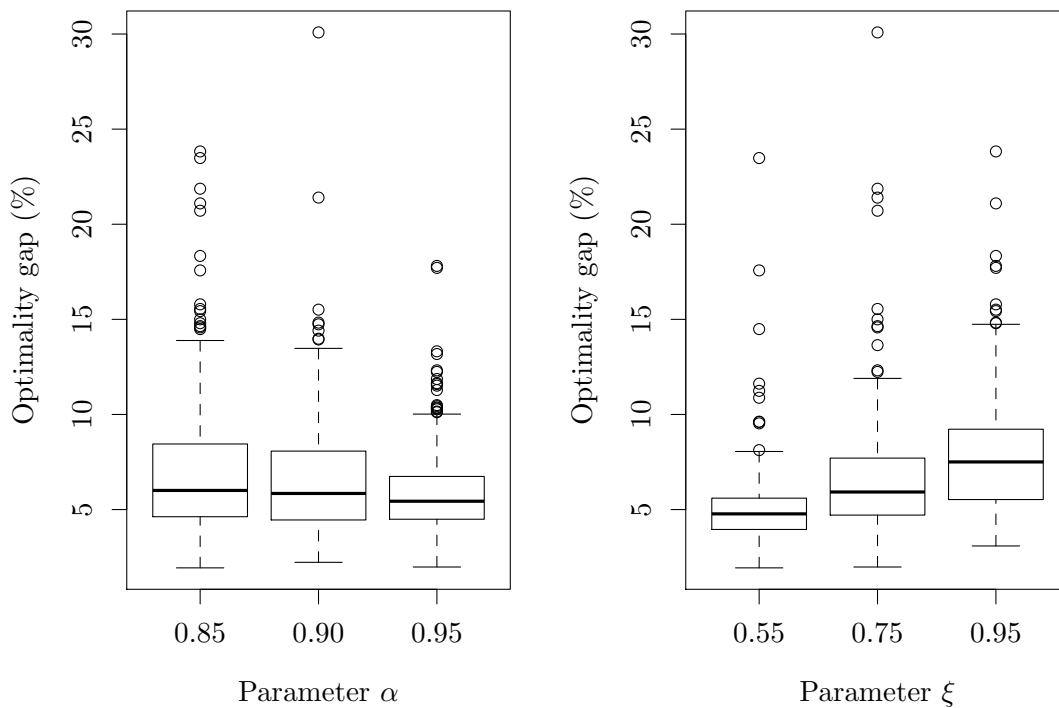
		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	4.0475	5.3627	6.4522
		2	5.1460	5.7293	7.2512
		4	5.7603	7.7781	8.9063
	0.9	1	4.4646	4.8157	5.9681
		2	4.2544	6.0540	6.8553
		4	4.9394	7.0242	8.0746
	0.95	1	4.6918	4.8278	5.4780
		2	4.7040	5.3554	6.6066
		4	4.9465	5.2437	6.6878

TABLE 8.12: Mean computation times required for all the combinations of parameter values during the second phase of the parameter optimisation experiment (involving variation of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ) for the 21-unit test system with maximisation of expected energy production as scheduling criterion.

		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	14 243	4 184	1 462
		2	7 888	1 137	496
		4	3 196	577	176
	0.9	1	14 067	8 659	2 125
		2	14 127	4 700	765
		4	10 781	750	290
	0.95	1	14 400	13 922	6 623
		2	14 400	10 384	2 175
		4	13 986	4 735	468

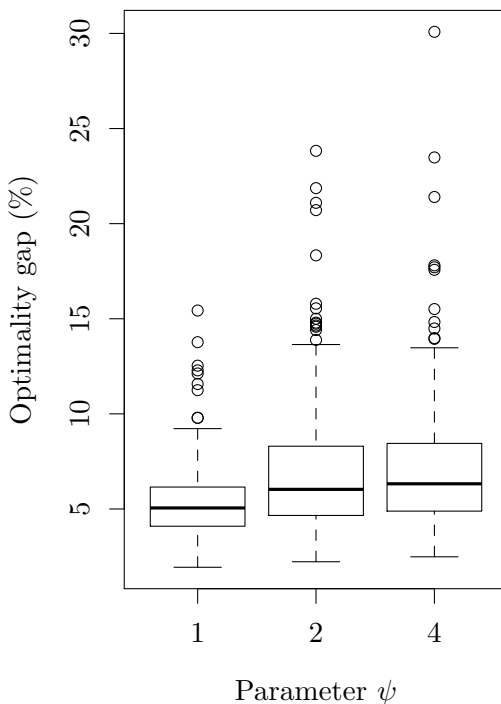
TABLE 8.13: Number of infeasible incumbents (out of 30) returned for all the combinations of parameter values during the second phase of the parameter optimisation experiment (involving variations of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ) for the 21-unit test system with maximisation of expected energy production as scheduling criterion.

		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	6 (20.00%)	9 (30.00%)	12 (40.00%)
		2	6 (20.00%)	9 (30.00%)	10 (33.33%)
		4	10 (33.33%)	15 (50.00%)	19 (63.33%)
	0.9	1	4 (13.33%)	5 (16.67%)	7 (23.33%)
		2	5 (16.67%)	6 (20.00%)	12 (40.00%)
		4	9 (30.00%)	16 (53.33%)	21 (70.00%)
	0.95	1	4 (13.33%)	5 (16.67%)	10 (33.33%)
		2	6 (20.00%)	10 (33.33%)	13 (43.33%)
		4	9 (30.00%)	10 (33.33%)	17 (56.67%)



(a) Box plot comparison of the optimality gap as a function of the cooling parameter

(b) Box plot comparison of the optimality gap as a function of the reheating parameter



(c) Box plot comparison of the optimality gap as a function of the epoch parameter

FIGURE 8.15: Box plot comparison of the optimality gaps obtained during the second phase of the parameter optimisation experiment (involving variation of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ) for the 21-unit test system with maximisation of expected energy production as scheduling criterion.

gap is observed as the reheating parameter increases. The mean optimality gaps corresponding to the reheating parameter values of 0.55, 0.75 and 0.95 are 5.053%, 6.567% and 7.944%, respectively. It is also observed in Figure 8.15(b) that as the reheating parameter increases, an undesirable increase results in the spreads and the number of outliers of the optimality gaps.

An analysis of the epoch parameter in terms of the mean optimality gap shows a gradual increase in the optimality gap as the epoch parameter increases. This is not so easily observed in Table 8.11, but it is clear in Figure 8.15(c). In this box plot, it may be seen that as the epoch parameter increases from 1 to 4, the optimality gap increases gradually. The mean optimality gaps for epoch parameter values of 1, 2 and 4 are 5.481%, 6.952% and 7.130%, respectively. It is also important to note that as the epoch parameter increases, so does the spread and the number of outliers of the optimality gaps.

The effects of these three parameter combinations on the required computation time are reported in Table 8.12. The mean computation time required to solve the 21-unit test system for each combination of the three parameter values, is presented in Table 8.12 and Figure 8.16. In the figure it may be observed that as the cooling parameter increases, an increase in the required computation time results. The mean computation time required for the cooling parameter values of 0.85, 0.90 and 0.95 are 3 707 seconds, 6 252 seconds and 9 010 seconds, respectively. This increase in computation time may be the result of a higher cooling parameter providing a slower decay in the temperature of the SA algorithm which may, in turn, cause the algorithm to terminate after a longer time.

In Table 8.12 and Figure 8.16, the required computation time is observed to decrease as the reheating parameter increases. It is clear that as one moves from right to left in the table, the computation time decreases. The average computation time required for evaluation of the reheating parameter values of 0.55, 0.75 and 0.95 are 11 899 seconds, 5 450 seconds and 1 620 seconds, respectively, which represents a very large variation. One reason for observing very short computation times for large reheating parameter values may be that the increase in temperature of the SA algorithm caused by a large reheating parameter is very small and so the algorithm may terminate easier due to the three-consecutive-reheating termination criterion.

The effect that the epoch parameter has on the computation time required by the SA algorithm to solve the 21-unit test system may also be observed in Table 8.12 and Figure 8.16. As the epoch parameter value increases, the required computation time decreases. The mean computation time required for epoch parameter values of 1, 2 and 4 are 8 854 seconds, 6 230 seconds and 3 884 seconds, respectively, which again represents a large variation. A reason for this observation may be that a larger epoch parameter value causes a shorter epoch which, in turn, results in a decrease in the number of iterations required before cooling or reheating is performed. This may cause the entire SA algorithm to terminate faster, because the temperature will decrease very rapidly and three consecutive reheats may be achieved early on during the algorithmic execution.

Finally, the number of infeasible incumbents returned (out of 30) during execution of the SA algorithm is reported in Table 8.13. The mean number of infeasible incumbents returned during each search run is presented in Figure 8.17. Here no significant change in the number of infeasible incumbents returned is observed when varying the cooling parameter. The mean number of infeasible incumbents returned (out of 30) for cooling parameter values of 0.85, 0.90 and 0.95 are 10.667, 9.444 and 9.333, respectively. In terms of the reheating parameter and epoch parameter, an almost exponential increase in the number of infeasible incumbents returned is observed as these parameter values increase. The average number of infeasible incumbents returned (out of 30) for reheating parameter values of 0.55, 0.75 and 0.95 are 6.556, 9.444 and 13.444, respectively, whereas the average number of infeasible incumbents returned (out of 30) for the epoch parameter values of 1, 2 and 4 are 6.889, 5.556 and 14.000, respectively.

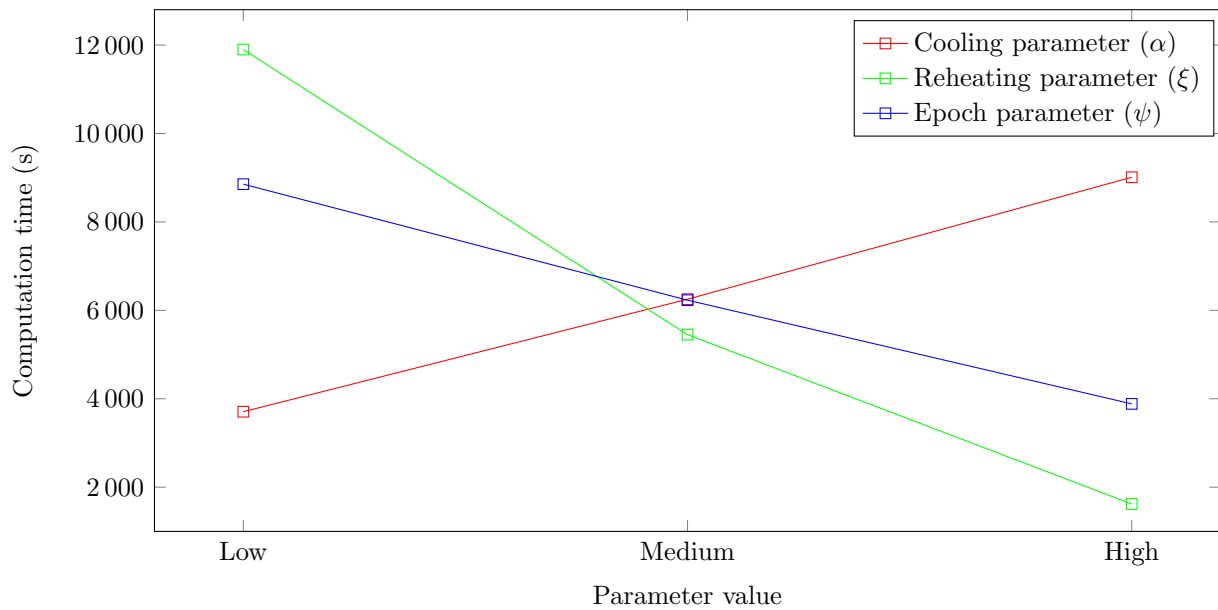


FIGURE 8.16: The computation time required to solve the 21-unit test system as a function of the cooling parameter, the reheating parameter and the epoch parameter with maximisation of expected energy production as scheduling criterion.

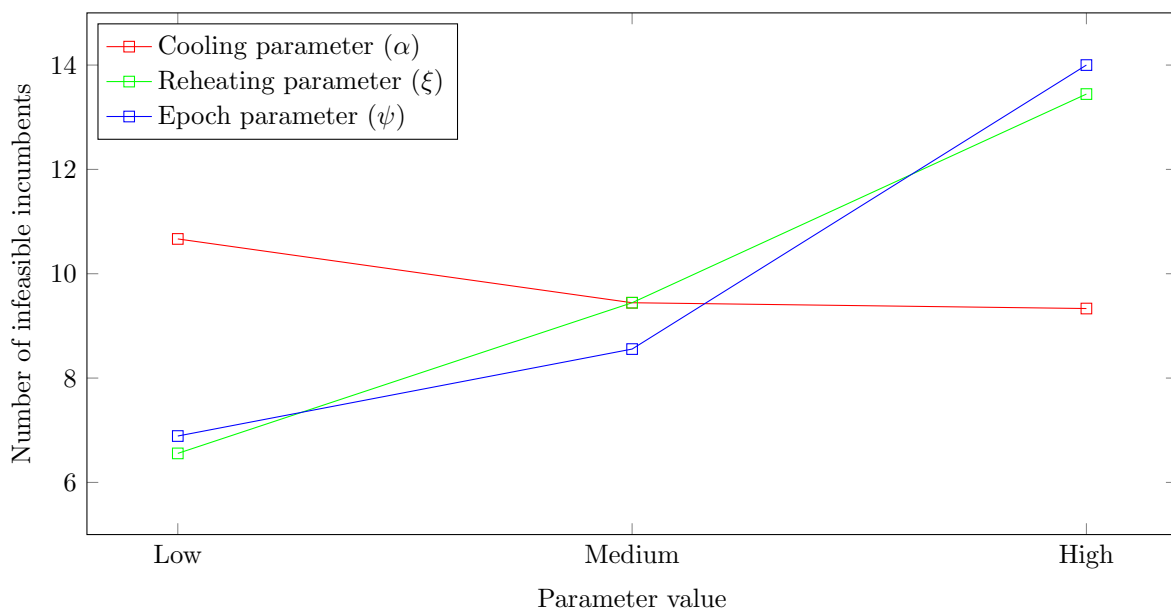


FIGURE 8.17: The mean number of infeasible incumbents (out of 30) as a function of the cooling parameter, the reheating parameter and the epoch parameter for the 21-unit test system with maximisation of expected energy production as scheduling criterion.

The aim of the parameter optimisation experiment in this section was to obtain a suitable combination of parameters which may be used in the approximate solution approach (*i.e.* the method of SA) so as to obtain good GMS solutions for the 21-unit test system with maximisation of expected energy production as scheduling criterion. In Figure 8.15(b), it may be observed that the large value for the reheating parameter achieves the largest mean optimality gap and that many outliers are observed in this case. This parameter value also produced a large mean number of infeasible incumbents. The reheating parameter value of 0.95 was therefore eliminated from the parameter values considered as candidates for the final set of parameters in the context of the 21-unit test system. It was furthermore observed that a reheating parameter of 0.55 results in the longest required computation time (an average of 11 899 seconds). For this reason, this parameter value was also eliminated from the parameter values considered as candidates for the final set of parameters for the 21-unit test system. The only reheating parameter value left, is therefore 0.75, which achieves an acceptable mean optimality gap value as well as an acceptable mean computation time. This parameter value also returns, on average, fewer than ten infeasible incumbents out of thirty.

Furthermore, the cooling parameter value of 0.95 was also eliminated from the candidates considered due to the parameter value returning, on average, a more than six percent optimality gap. Hence a cooling parameter value of 0.95 was therefore eliminated from the parameter values considered as candidates for the final set of parameters in the context of the 21-unit test system. The cooling parameter value of 0.85 was also eliminated from the parameter values considered as candidates for the final set of parameters due to the parameter value returning, on average, the largest computation time of the three cooling parameter values (an average of 8 854 seconds). Hence the cooling parameter value of 0.9 was selected as the best candidate for the 21-unit test system. Finally, in terms of the results obtained for the epoch parameter it was observed that the larger value (*i.e.* 4) returned 14 infeasible incumbents out of thirty, which is an undesirably large proportion. Of the remaining epoch parameter values, the value 1 was selected as the final epoch parameter as this value results in a smaller mean optimality gap as well as returning a smaller number of infeasible incumbents of seven out of thirty. The final set of parameters adopted in the SA algorithm for solving the 21-unit test system is shown in Table 8.14.

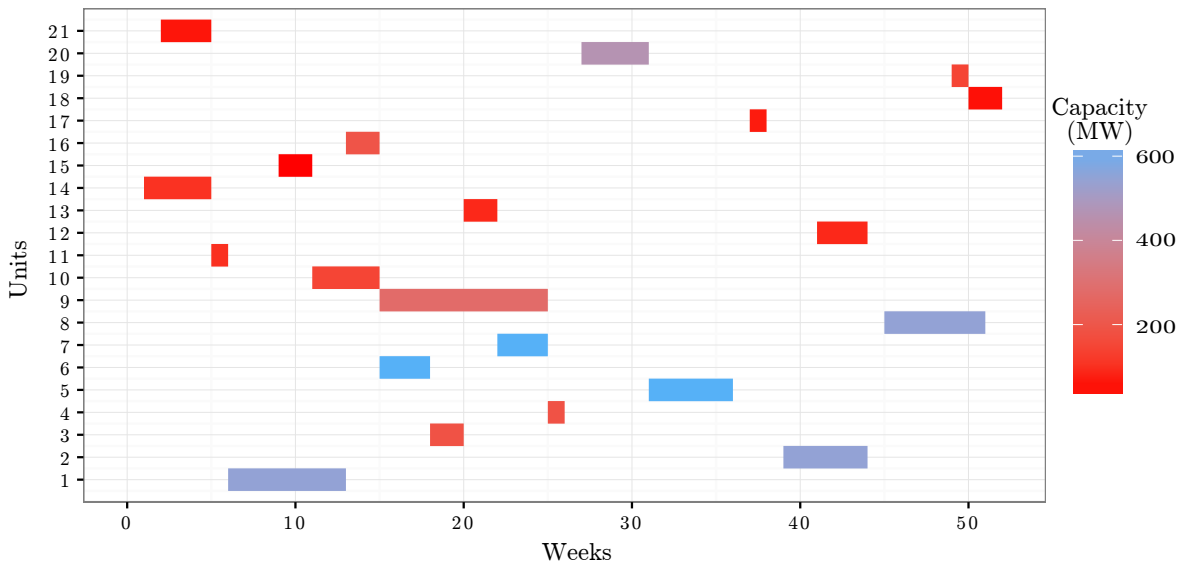
TABLE 8.14: *The complete set of parameters selected for the 21-unit test system with maximisation of expected energy production as scheduling criterion, as well as the mean optimality gap, the mean required computation time and the number of infeasible incumbents associated with these parameter values during the second phase of the parameter optimisation experiment.*

χ_0	γ	α	ξ	ψ	Optimality gap (%)	Computation time (s)	Infeasibilities
0.6	0.75	0.9	0.75	1	4.816	8 659	5 (out of 30)

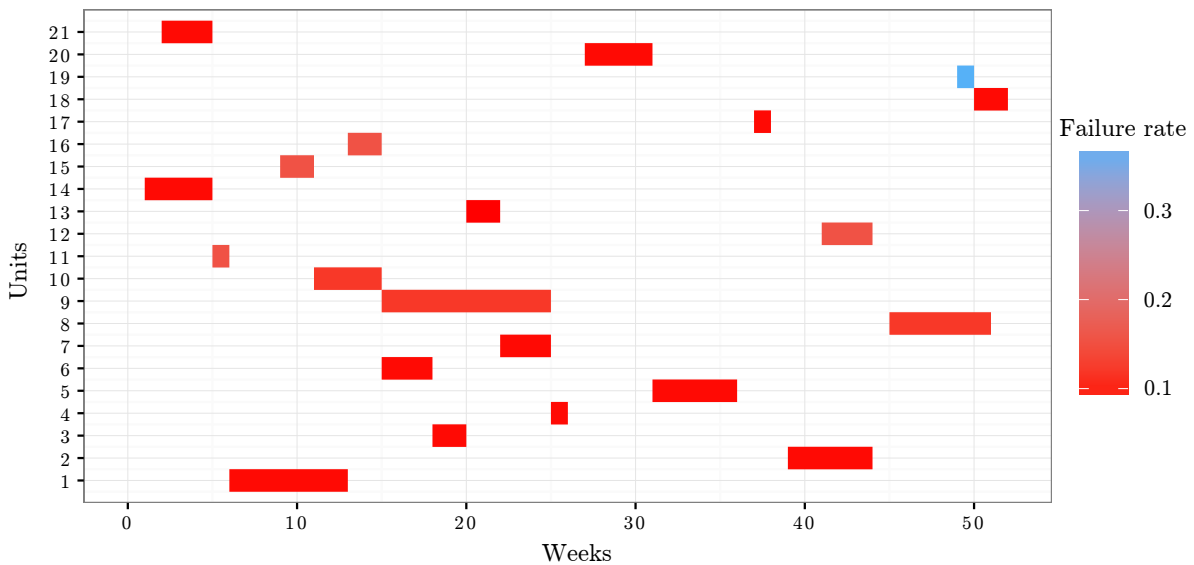
Numerical results

An approximate solution to the nonlinear GMS model of §4.3 was obtained for the 21-unit test system by the SA algorithm with the parameter combinations as specified in Table 8.14. The decision variable values of the incumbent are given in integer decision vector form by $\mathbf{x} = [7, 40, 19, 26, 32, 16, 23, 46, 16, 12, 6, 42, 21, 2, 10, 14, 38, 51, 50, 28, 3]$, which corresponds to an objective function value of 213 853 MW·week (2.669% worse than that of the optimal piecewise linear approximation solution for the 21-unit test system, as reported in §8.1.1).

A graphical representation of this maintenance schedule is presented in Figure 8.18 with the colour scale in Figure 8.18(a) indicating the capacity (in MW) of each PGU and the colour scale in Figure 8.18(b) indicating the failure rate of each PGU. The manpower required over the



(a) The incumbent returned by the SA algorithm for the nonlinear model of §4.3, with the colour scale indicating the rated capacity of the PGUs



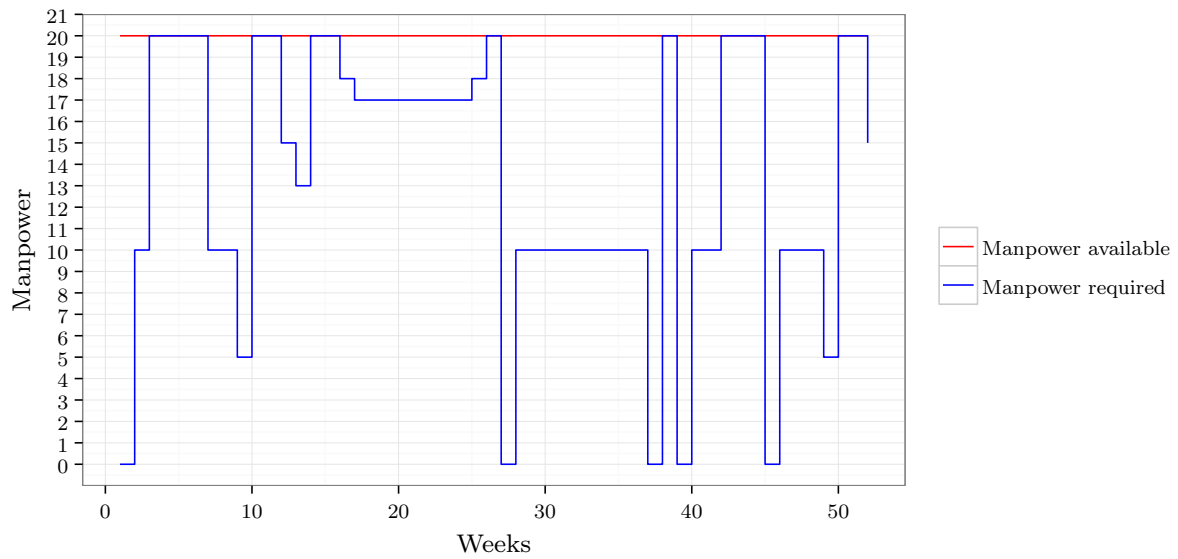
(b) The incumbent returned by the SA algorithm for the nonlinear model of §4.3, with the colour scale indicating the failure rate of the PGUs

FIGURE 8.18: The incumbent returned by the SA algorithm for the nonlinear model of §4.3, in the context of the 21-unit test system with maximisation of expected energy production as scheduling criterion and for parameter values as indicated in Table 8.14.

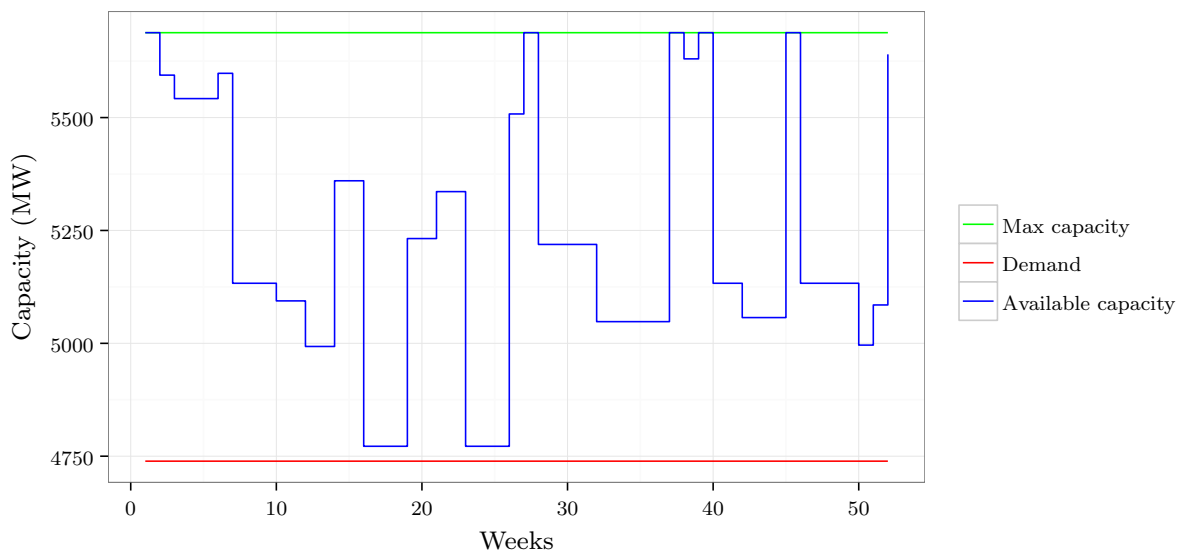
duration of the scheduling window in order to implement the solution in Figure 8.18 is shown in Figure 8.19(a), while the available system capacity over the duration of the scheduling window associated with this solution is shown in Figure 8.19(b).

Comparison with results obtained by piecewise linear approximation

The solution reported above may be analysed by comparing it with the optimal piecewise linear approximation solution obtained in §8.1.1 for the 21-unit test system with maximisation of expected energy production as scheduling criterion. A graphical representation of the in-



(a) The manpower required over the scheduling window corresponding to the maintenance schedule in Figure 8.18



(b) The system capacity over the scheduling window corresponding to the maintenance schedule in Figure 8.18

FIGURE 8.19: Evaluation of the manpower required and system capacity available over the duration of the scheduling window for the 21-unit test system maintenance schedule in Figure 8.18.

cumbent solution returned by the approximate solution methodology of §5.3 (*i.e.* the method of SA) and the optimal piecewise linear approximation solution obtained by the exact solution approach described in §8.1.1 (*i.e.* by means of CPLEX) is shown in Figure 8.20 for the 21-unit test system. The maintenance schedules of the two solutions are compared in the figure, with the colour scale indicating the rated capacity of the PGUs in the system. The effects of the two maintenance schedules on the manpower required and the available system capacity are shown in Figures 8.21(a) and 8.21(b), respectively.

Comparing these two maintenance schedules, a reasonable number of differences are observed in the scheduled commencement times of the PGUs, which cause the objective function values to

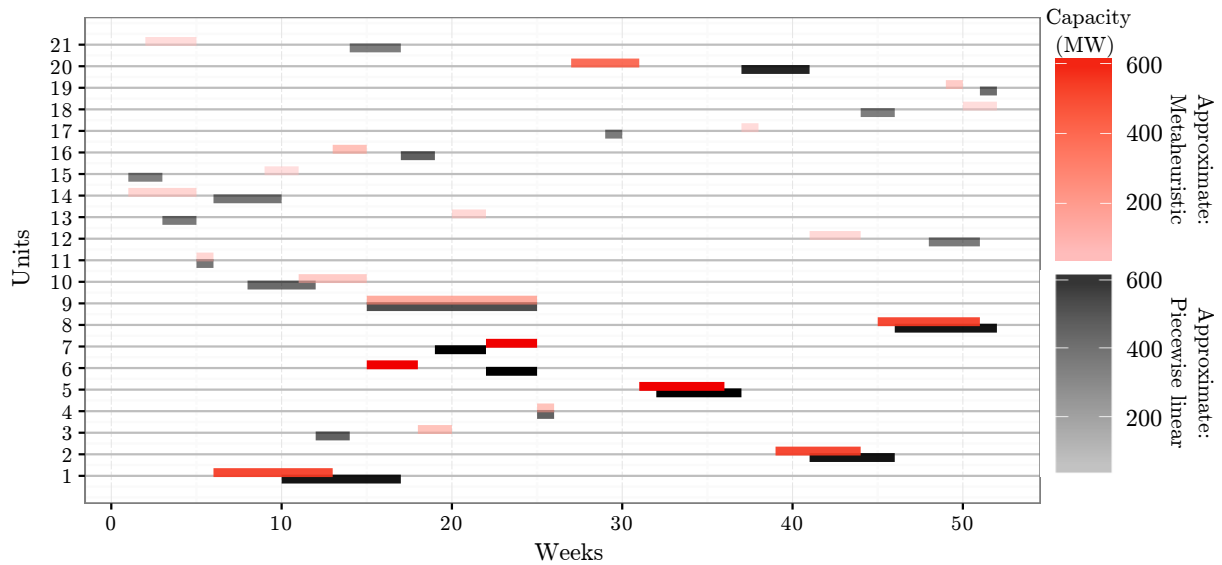


FIGURE 8.20: Two maintenance schedules for the 21-unit test system returned by two different approaches towards solving the nonlinear GMS model of §4.3 with maximisation of expected energy production as scheduling criterion.

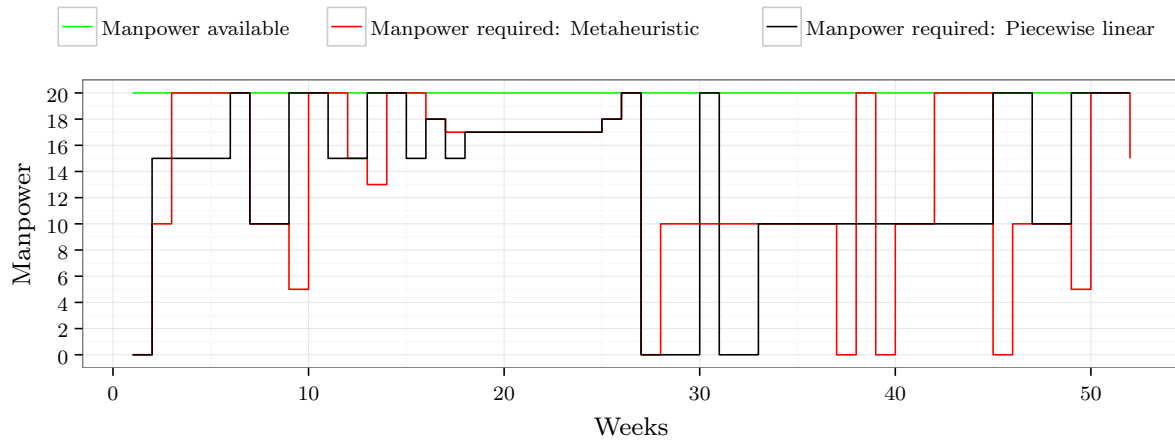
differ by 2.669%. Comparing the available manpower in Figure 8.21(a), it is seen for the solution obtained by the piecewise linear approximation solution approach that no manpower is required during the early stages of the scheduling window after which it increases within a short period to the maximum available manpower. This is due to the proposed objective function scheduling PGUs with large rated capacities in the scheduling window close to the peak of their expected energy production curves, resulting in these PGUs being scheduled for maintenance late during the scheduling window and the available manpower being low during the early stages of the scheduling window and larger towards the end. A similar result is observed in the case of the metaheuristic solution. It is observed that the available manpower is low early in the scheduling window and then gradually increases toward the end of the first half of the maintenance window. The major difference between these two solutions may be observed in the second half of the scheduling window where many jumps are observed for the result obtained by the metaheuristic solution. Figure 8.21(b) reveals that there are some similarities for these two solutions in terms of their available system capacities over the scheduling window. During the early stages of the scheduling window the system capacities are similar, but a noticeable difference is observed towards the end of the scheduling window where some jumps in the system capacity are observed for the metaheuristic solution. The piecewise linear approximation solution, on the other hand, exhibits a gradual decrease in the available system capacity towards the end of the scheduling window.

8.2.2 The IEEE-RTS

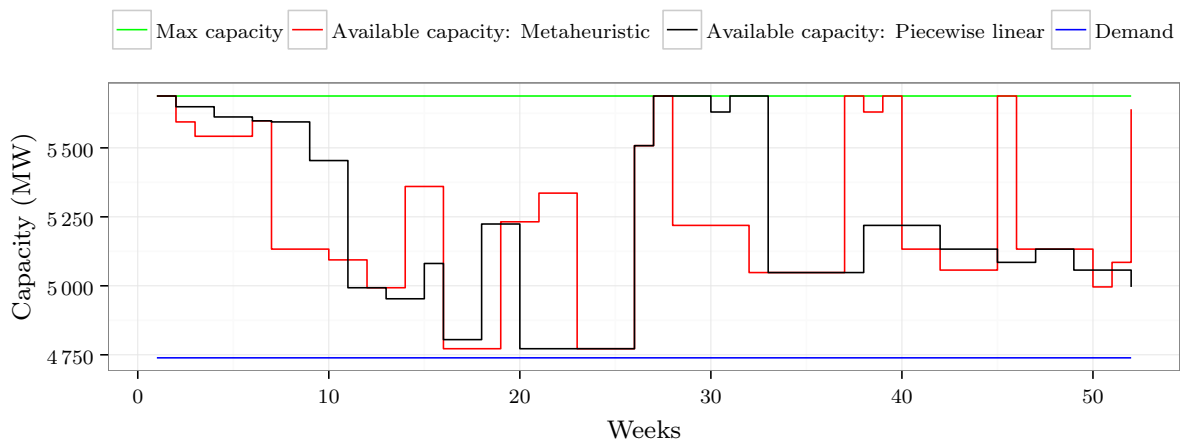
This section is devoted to an application of the parameter optimisation experiment described in §5.3.3 within the context of the IEEE-RTS. The best algorithmic parameter combination thus uncovered is then employed to solve the test instance approximately in terms of the maximisation of the energy production scheduling criterion.

Parameter optimisation experiment

In this section, a parameter optimisation experiment is performed for the IEEE-RTS in two separate phases according to the design described in §5.3.3. The first phase of the experiment



(a) The manpower required over the scheduling window corresponding to the maintenance schedules in Figure 8.20



(b) The system capacity over the scheduling window corresponding to the maintenance schedules in Figure 8.20

FIGURE 8.21: Comparison between the manpower required and the available system capacity associated with the schedules in Figure 8.20 for the 21-unit test system with maximisation of expected energy production as scheduling criterion.

involves variation of the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ . During the second phase of the experiment, the parameters varied are the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ .

Phase 1: Initial acceptance ratio & soft constraint violation severity factor

The mean optimality gaps (measured as percentages relative to the optimal objective function values obtained by CPLEX for the piecewise linear approximation in §8.1.2) associated with the feasible incumbents returned during the first phase of the parameter optimisation experiment are shown in Table 8.15. The mean computation times involved in evaluating combinations of these parameter values are shown in Table 8.16 (which includes computation times expended during runs that returned infeasible incumbents). The numbers of times (out of 30) that an infeasible incumbent was returned during an SA search run are finally shown in Table 8.17.

TABLE 8.15: Mean optimality gaps for all the combinations of the first phase of the parameter optimisation experiment (involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ) for the IEEE-RTS with maximisation of the expected energy production as scheduling criterion.

χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	2.101	2.190	2.602	2.060	2.136
0.5	1.827	2.373	2.172	2.456	2.191
0.6	2.015	1.988	2.212	2.454	2.121
0.7	2.122	2.177	2.524	2.344	2.379
0.8	1.872	1.998	2.033	2.359	2.394

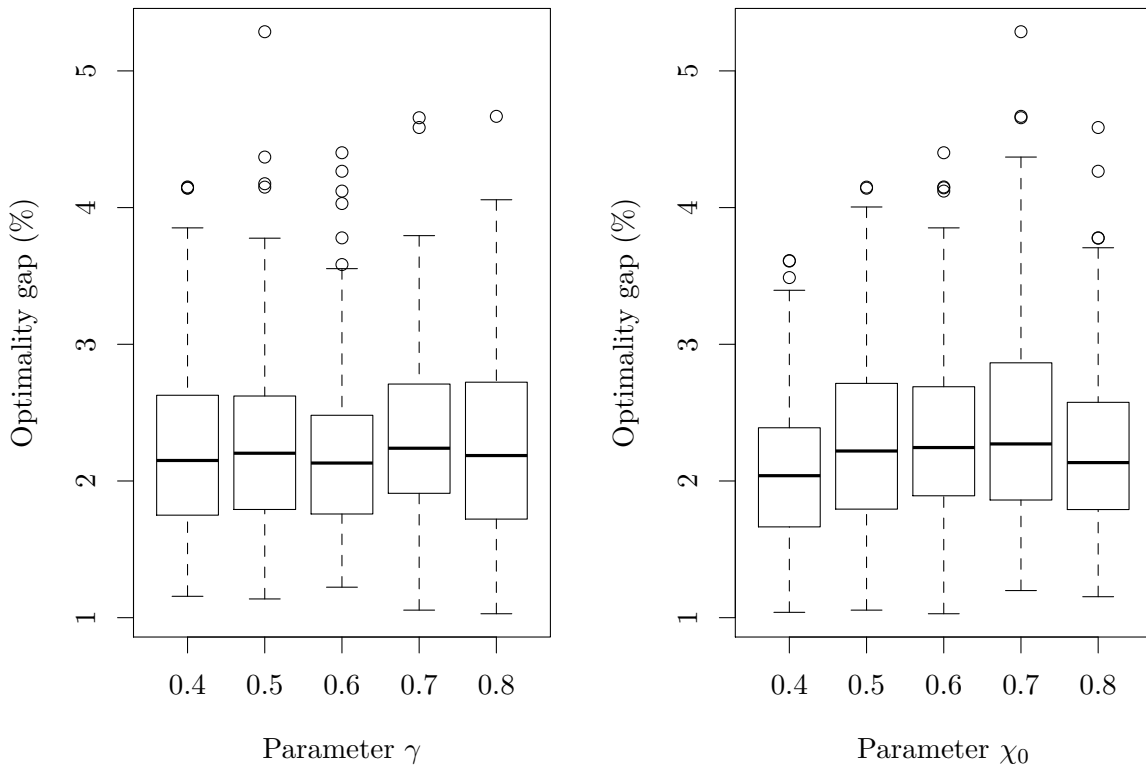
TABLE 8.16: Mean computation times required for all the combinations of the first phase of the parameter optimisation experiment (involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ) for the IEEE-RTS with maximisation of the expected energy production as scheduling criterion.

χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	18280	17613	15721	19649	19068
0.5	16059	14402	17647	18774	22381
0.6	16047	18279	16691	16587	19701
0.7	17265	17432	18244	19231	20699
0.8	14110	17489	19373	16461	18022

In these results, no clear correlation is observed between the mean optimality gap and the soft constraint violation severity factor. As the severity factor increases, no significant change in the mean optimality gap is observed. This may also be seen in Figure 8.22(a) where a box plot comparison is presented of the mean optimality gap as a function of the soft constraint violation severity factor. It is also observed that there is no significant change in the spread of the optimality gaps as the severity factor increases. The severity factor value with the largest number of outliers was observed to be 0.6. The effect of the initial acceptance ratio on the mean optimality gap seems slightly more prominent. It is observed that the mean optimality gap increases as the initial acceptance ratio is increased. A decrease is, however, observed in the mean optimality gap for the largest initial acceptance ratio value of 0.8. This is clearly visible in the box plot comparison presented in Figure 8.22(b). It is observed that the spreads and number of outliers of the mean optimality gaps for all five of the initial acceptance ratio values are fairly similar. This similarity in mean optimality gaps may be due to the fact that the SA

TABLE 8.17: Number of infeasible incumbents (out of 30) returned for all the combinations of the first phase of the parameter optimisation experiment (involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ) for the IEEE-RTS with maximisation of the expected energy production as scheduling criterion.

χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	26 (86.67%)	17 (56.67%)	7 (23.33%)	7 (23.33%)	3 (10.00%)
0.5	27 (90.00%)	16 (53.33%)	7 (23.33%)	4 (13.33%)	0 (0.00%)
0.6	26 (86.67%)	13 (43.33%)	7 (23.33%)	1 (3.33%)	4 (13.33%)
0.7	28 (93.33%)	17 (56.67%)	8 (26.67%)	6 (20.00%)	5 (16.67%)
0.8	27 (90.00%)	12 (40.00%)	7 (23.33%)	11 (36.67%)	2 (6.67%)



(a) Box plot comparison of the optimality gap as a function of the soft constraint violation severity factor

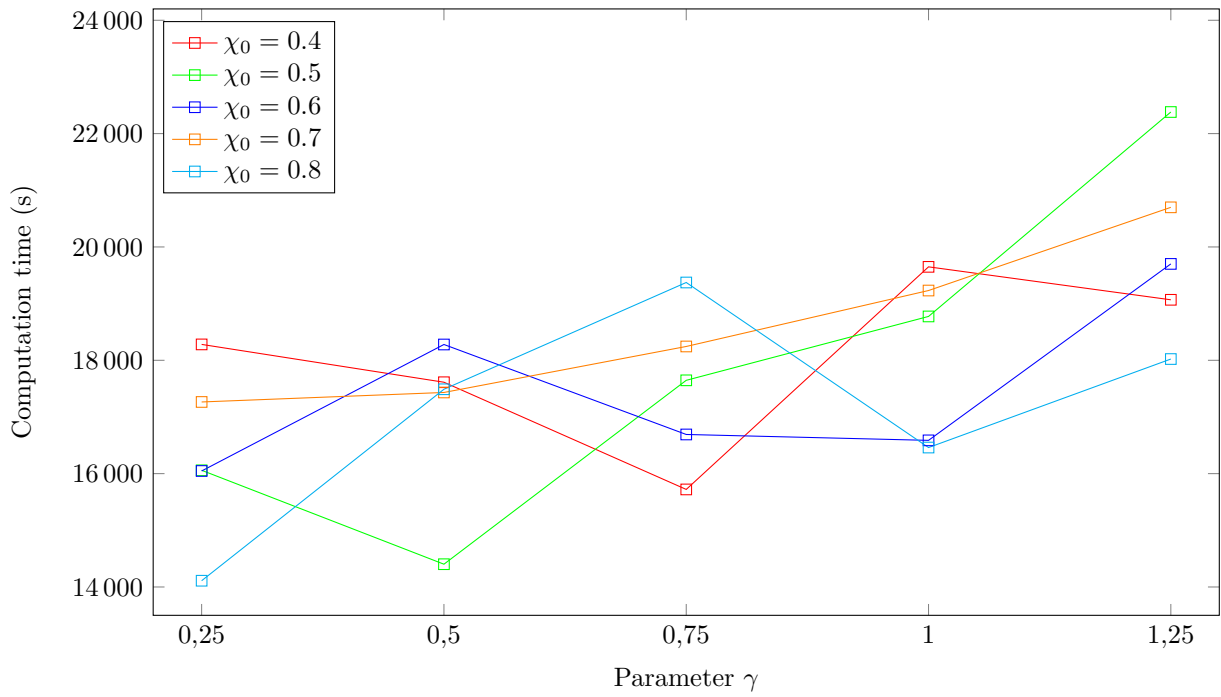
(b) Box plot comparison of the optimality gap as a function of the initial acceptance ratio

FIGURE 8.22: Box plot comparison of the optimality gaps for the first phase of the parameter optimisation experiment involving the soft constraint violation severity factor γ and the initial acceptance ratio χ_0 for the IEEE-RTS with maximisation of expected energy production as scheduling criterion.

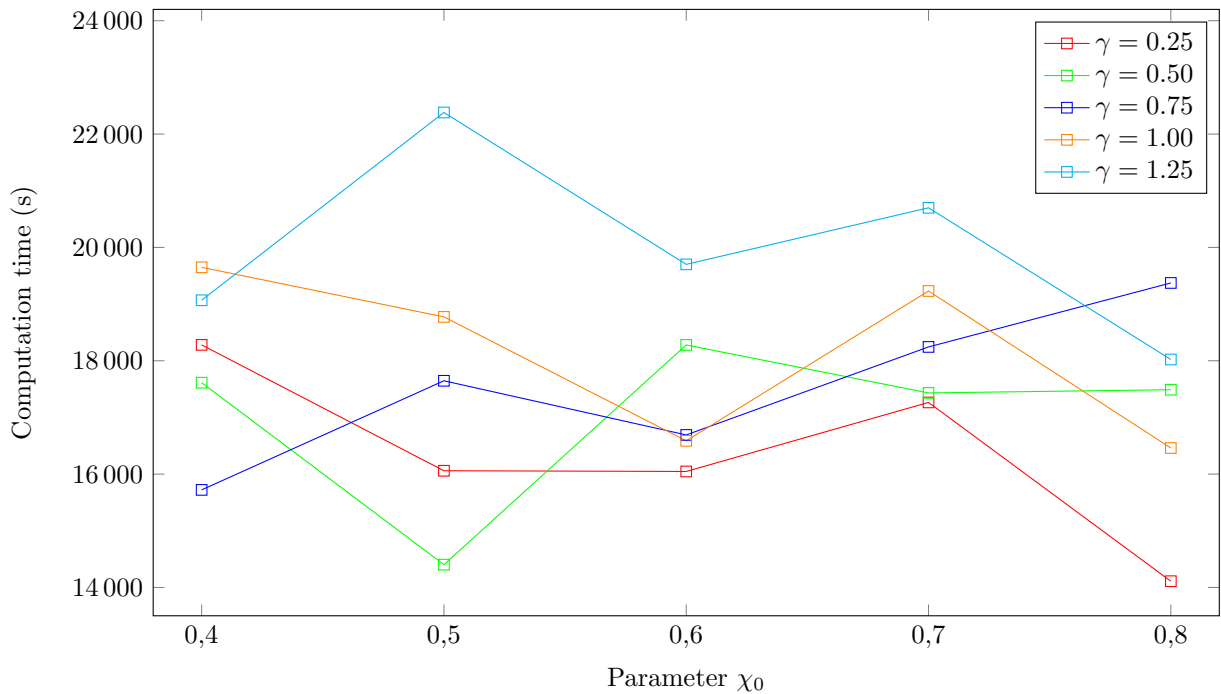
algorithm typically terminates as a result of the three-consecutive-epoch termination criterion and not as a result of the maximum allowable computation time. The algorithm therefore has enough time to reach an acceptably good incumbent solution, even when the initial temperature is higher in some cases (as is the case of a higher initial acceptance ratio).

In terms of the computation time expended to evaluate each of the parameter combinations presented in Table 8.16, a slight increase in computation time is observed as the soft constraint violation factor increases. This is visible in Figure 8.23(a), which contains a plot of the computation time as a function of the soft constraint violation severity factor for the different initial acceptance ratios. For the initial acceptance ratio, no increase or decrease in computation time is observed in the mean computation time when this parameter is increased, as may be seen in Figure 8.23(b).

The numbers of infeasible incumbents returned (out of the 30 test runs) for each of the 25 combinations of the soft constraint violation severity factor and the initial acceptance ratio are shown in Table 8.17. It is observed in this table that as the soft constraint violation severity factor increases, the number of infeasible incumbents decreases. A very large proportion of infeasible incumbents are observed for the small value of the soft constraint violation severity factor value of 0.25. It is, however, observed that as the soft constraint violation severity factor increases, the number of infeasible incumbents returned (out of the 30) decreases exponentially.

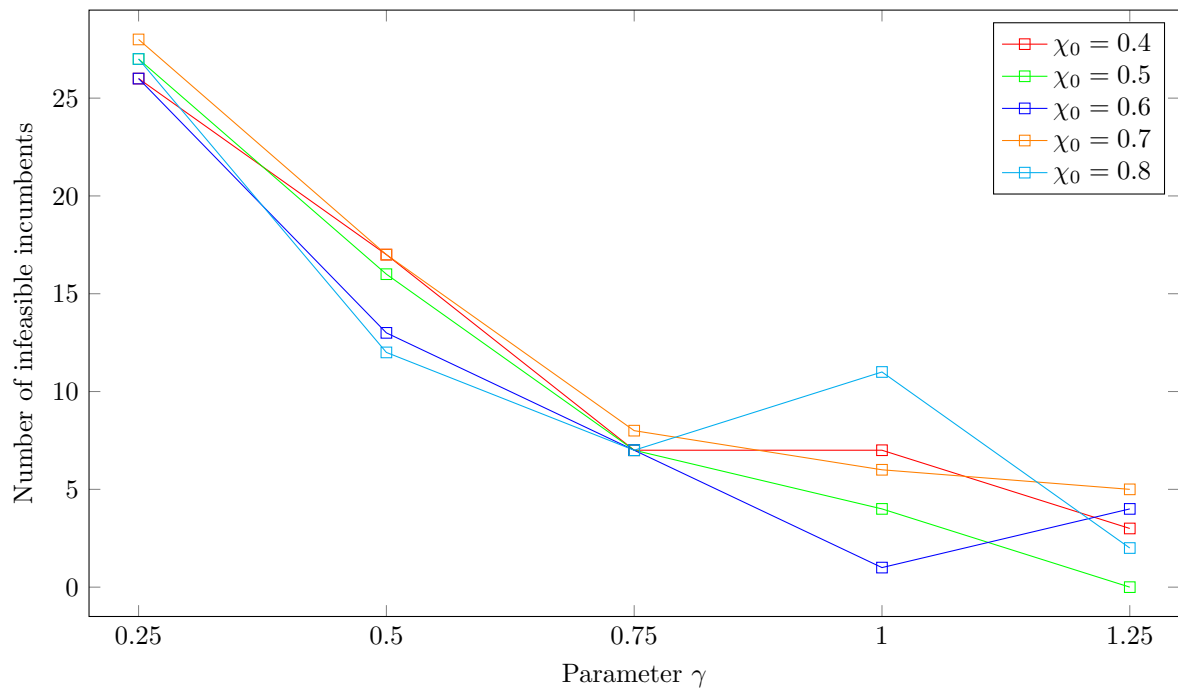


(a) The computation time required as a function of the soft constraint violation severity factor γ for different values of the initial acceptance ratio χ_0

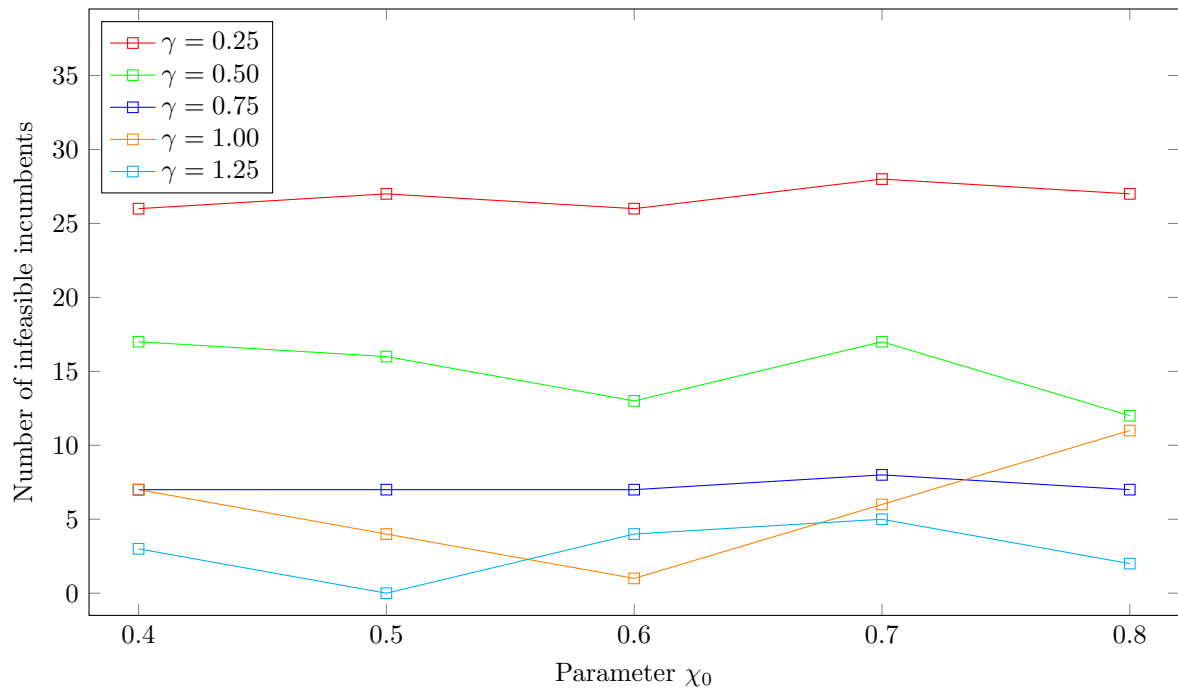


(b) The computation time required as a function of the initial acceptance ratio χ_0 for different values of the soft constraint violation severity factor γ

FIGURE 8.23: The computation time required as functions of the soft constraint violation severity factor γ and the initial acceptance ratio χ_0 when solving the IEEE-RTS with maximisation of expected energy production as scheduling criterion.



(a) The number of infeasible incumbents returned (out of 30) as a function of the soft constraint violation severity factor γ for different values of the initial acceptance ratio χ_0



(b) The number of infeasible incumbents returned (out of 30) as a function of the initial acceptance ratio χ_0 for different values of the soft constraint violation severity factor γ

FIGURE 8.24: The number of infeasible incumbents returned (out of 30) as functions of the soft constraint violation severity factor γ and the initial acceptance ratio χ_0 when solving the IEEE-RTS with maximisation of expected energy production as scheduling criterion.

This trend is also visible in Figure 8.24(a), where the number of infeasible incumbents returned is presented as a function of the soft constraint violation severity factor for different initial acceptance ratios. An almost exponential decay in the number of infeasible incumbents is observed as the soft constraint violation severity factor is decreased, with the highest severity factor value of 1.25 returning no infeasible incumbent solutions in one instance. For the initial acceptance ratio, however, it is observed, as previously also noted, that no significant change results in the number of infeasible incumbents returned as the value of the initial acceptance ratio varies. This may also be observed in Figure 8.24(b), where the number of infeasible incumbents is presented as a function of the initial acceptance ratio for different soft constraint violation severity factor values.

From the results obtained during the first phase of the parameter optimisation experiment, which included variation of the soft constraint violation severity factor and the initial acceptance ratio, a value of 1.25 is selected for the soft constraint violation severity factor and a value of 0.6 is selected for the initial acceptance ratio, as shown in Table 8.18. For this combination of parameter values, an acceptable mean optimality gap of 2.121% is obtained. As no clear trend is observed in the computation time as the parameters are varied, the computation time did not play a role in the final decision of the parameter combination. A mean computation time, however, of 19 701 seconds required to solve the IEEE-RTS is also acceptable as it is lower than the maximum amount of computation time allowed for the SA search, namely 28 800 seconds. A small proportion of infeasible incumbents is also returned (out of the 30 runs) by this parameter value combination, namely four. This combination is therefore adopted as the final parameter values in the context of solving the IEEE-RTS with maximisation expected energy production as scheduling criterion. These values for the soft constraint violation severity factor and the initial acceptance ratio are also adopted during the second phase of the parameter optimisation experiment involving variation of the cooling parameter, the reheating parameter and the epoch parameter values.

TABLE 8.18: *Selected parameter value combination for the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ , as obtained from the first phase of the parameter optimisation experiment for the IEEE-RTS with maximisation of expected energy production as scheduling criterion.*

χ_0	γ	Optimality gap (%)	Computation time (s)	Infeasibilities
0.6	1.25	2.121	19 701	4 (out of 30)

Phase 2: Cooling parameter, reheating parameter, epoch parameter

The mean optimality gaps (again measured as percentages relative to the optimal objective function values obtained by CPLEX) associated with the piecewise linear approximation in §8.1.2 for the feasible incumbents returned during the second phase of the parameter optimisation experiment, are shown in Table 8.19, while the mean computation times required for the evaluation of the combinations of these parameter values are shown in Table 8.20. These times again include computation times expended during runs that returned infeasible incumbents. The numbers of times (out of 30) that an infeasible incumbent was returned during an SA search run are shown in Table 8.21. Furthermore, box plot comparisons of the mean optimality gaps are presented as functions of the cooling parameter, the reheating parameter and the epoch parameter in Figures 8.25(a), 8.25(b) and 8.25(c), respectively.

From these results, it may be observed that there is no clear correlation between the cooling parameter and average optimality gap. As the cooling parameter increases from 0.85 to 0.90, the average optimality gap decreases, but then increases again as the cooling parameter increases

TABLE 8.19: Mean optimality gaps for all the combinations of parameter values during the second phase of the parameter optimisation experiment (involving variation of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ) for the IEEE-RTS with maximisation of expected energy production as scheduling criterion.

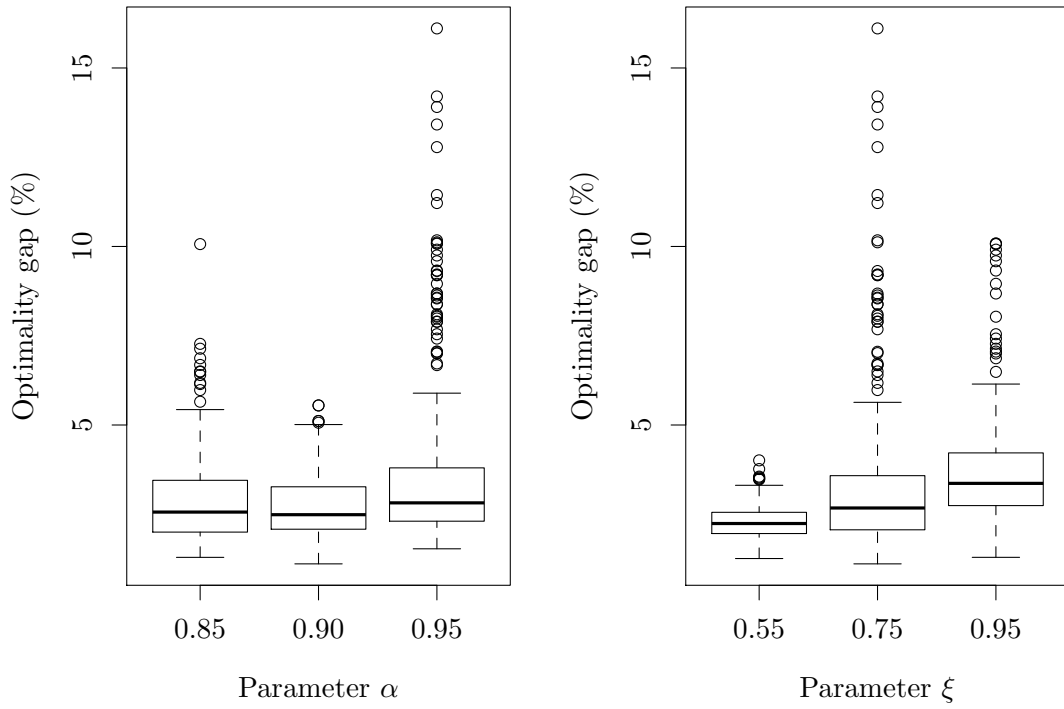
		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	2.0994	1.9131	2.7177
		2	2.1175	3.1025	3.2382
		4	2.0600	3.7495	4.0167
	0.9	1	2.2432	2.1733	2.8007
		2	2.1041	2.3783	3.1243
		4	2.2468	2.9572	4.1530
	0.95	1	3.4076	4.9138	5.9019
		2	2.8054	2.3129	3.0212
		4	2.5069	2.4565	3.4267

TABLE 8.20: Mean computation times required for all the combinations of parameter values during the second phase of the parameter optimisation experiment (involving variation of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ) for the IEEE-RTS with maximisation of expected energy production as scheduling criterion.

		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	28800	23617	9008
		2	28170	6391	3901
		4	24201	3060	1889
	0.9	1	28800	28196	23584
		2	28800	21298	9365
		4	26674	9351	4358
	0.95	1	28800	28800	28800
		2	28800	27864	16923
		4	28800	19601	6349

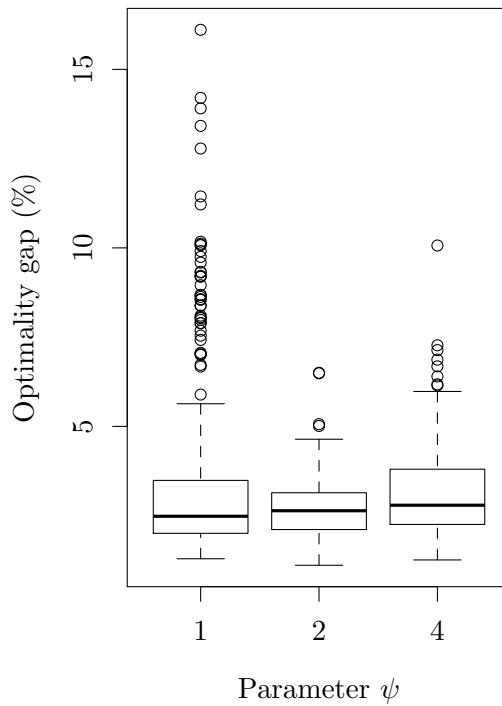
TABLE 8.21: Number of infeasible incumbents (out of 30) returned for all the combinations of parameter values during the second phase of the parameter optimisation experiment (involving variations of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ) for the IEEE-RTS with maximisation of expected energy production as scheduling criterion.

		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	3 (10.00%)	4 (13.33%)	5 (16.67%)
		2	1 (3.33%)	2 (6.67%)	6 (20.00%)
		4	3 (10.00%)	6 (20.00%)	9 (30.00%)
	0.9	1	7 (23.33%)	4 (13.33%)	1 (3.33%)
		2	2 (6.67%)	2 (6.67%)	2 (6.67%)
		4	2 (6.67%)	4 (13.33%)	4 (13.33%)
	0.95	1	5 (16.67%)	9 (30.00%)	5 (16.67%)
		2	6 (20.00%)	4 (13.33%)	4 (13.33%)
		4	5 (16.67%)	1 (3.33%)	5 (16.67%)



(a) Box plot comparison of the optimality gap as a function of the cooling parameter

(b) Box plot comparison of the optimality gap as a function of the reheating parameter



(c) Box plot comparison of the optimality gap as a function of the epoch parameter

FIGURE 8.25: Box plot comparison of the optimality gaps obtained during the second phase of the parameter optimisation experiment (involving variation of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ) for the IEEE-RTS with maximisation of expected energy production as scheduling criterion.

from 0.90 to 0.95. This may be seen in Figure 8.15(a). The mean optimality gaps corresponding to the cooling parameter values of 0.85, 0.90 and 0.95 are 2.892%, 2.7224% and 3.822%, respectively. In Figure 8.15(a) it is also observed that the largest cooling parameter 0.95 exhibits the largest number of outliers which contributes toward the average optimality gap being larger. The cooling parameter with the fewest outliers is observed to be 0.90.

The correlation between the reheating parameter and the mean optimality gap is much more prominent than that between the mean optimality gap and the cooling parameter. As the reheating parameter increases in Table 8.19 (*i.e.* moving from left to right in the table), an increase in the mean optimality gap is observed. This is also observed in the box plot in Figure 8.25(b), which contains a comparison of the mean optimality gaps for the three different reheating parameter values of 0.55, 0.75 and 0.95. Here a clear increase in the mean optimality gap is observed as the reheating parameter increases. The mean optimality gaps corresponding to the reheating parameter values of 0.55, 0.75 and 0.95 are 2.284%, 3.441% and 3.712%, respectively. It is also observed in Figure 8.25(b) that as the reheating parameter increases, an undesirable increase results in the spreads and the number of outliers of the optimality gaps, with the reheating parameter value of 0.75 returning the largest number of outliers.

An analysis of the epoch parameter in terms of the mean optimality gap shows no clear effect on the optimality gap as the epoch parameter increases. This is not so easily observed in Table 8.11, but it is clear in Figure 8.15(c). The mean optimality gaps for epoch parameter values of 1, 2 and 4 are 3.567%, 2.702% and 3.168%, respectively. It is also important to note that there is no clear correlation between the epoch parameter and the spread or number of outliers. The epoch parameter 1 returns the largest number of outliers.

The mean computation time required to solve the IEEE-RTS for each combination of the three parameter values is presented in Table 8.20 and Figure 8.26. In the figure it may be observed that as the cooling parameter increases, an increase in the required computation time results. The mean computation time required for the cooling parameter values of 0.85, 0.90 and 0.95 are 14 337 seconds, 20 047 seconds and 23 860 seconds, respectively. This increase in computation time may be the result of a largest cooling parameter providing a slower decay in the temperature of the SA algorithm which may, in turn, cause the algorithm to terminate after a longer time.

In Table 8.20 and Figure 8.26, the required computation time is observed to decrease as the reheating parameter increases. It is clear that as one moves from left to right in the table, the computation time increases. The average computation time required for evaluation of the reheating parameter values of 0.55, 0.75 and 0.95 are 27 983 seconds, 18 686 seconds and 11 575 seconds, respectively, which represents a very large variation. One reason for observing very short computation times for large reheating parameter values may be that the increase in temperature of the SA algorithm caused by a large reheating parameter is very small and so the algorithm may terminate more easily due to the three-consecutive-reheating termination criterion.

The effect that the epoch parameter has on the computation time required by the SA algorithm to solve the IEEE-RTS may also be observed in Table 8.20 and Figure 8.26. As the epoch parameter value increases, the required computation time decreases. The mean computation time required for epoch parameter values of 1, 2 and 4 are 25 378 seconds, 19 057 seconds and 13 809 seconds, respectively, which again represents a large variation. A reason for this observation may be that a larger epoch parameter value causes shorter epochs which, in turn, results in a decrease in the number of iterations required before cooling or reheating is performed. This may cause the entire SA algorithm to terminate faster, because the temperature will decrease very rapidly and three consecutive reheats may be achieved early on during the algorithmic execution.

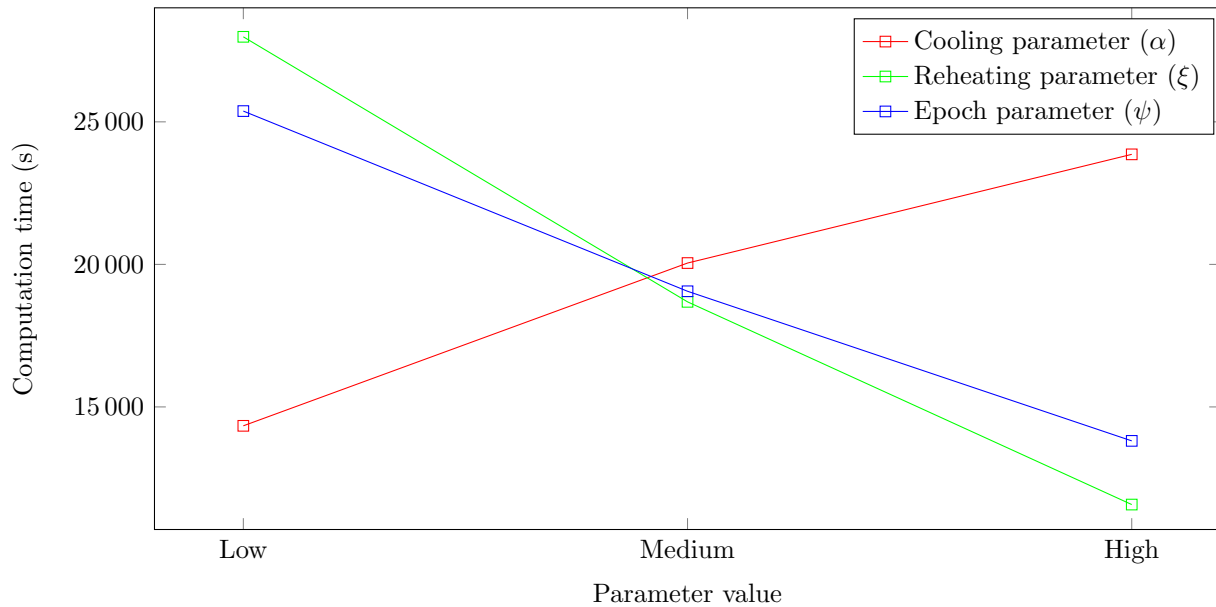


FIGURE 8.26: The computation time required to solve the IEEE-RTS as a function of the cooling parameter, the reheating parameter and the epoch parameter with maximisation of expected energy production as scheduling criterion.

Finally, the number of infeasible incumbents returned (out of 30) during execution of the SA algorithm is reported in Table 8.21. The mean number of infeasible incumbents returned during each search run is presented in Figure 8.27. Here no correlation between the number of infeasible incumbents returned is observed when varying the cooling parameter. The mean number of infeasible incumbents returned (out of 30) for cooling parameter values of 0.85, 0.90 and 0.95 are 4.333, 3.111 and 4.889, respectively. In terms of the reheating parameter and epoch parameter, an increase in the number of infeasible incumbents returned is observed as these parameter values increase. The average number of infeasible incumbents returned (out of 30) for reheating parameter values of 0.55, 0.75 and 0.95 are 3.778, 4.000 and 4.556, respectively. The average number of infeasible incumbents returned (out of 30) for epoch parameter values of 1, 2 and 4 are 4.778, 3.222 and 4.333, respectively. This again does not exhibit a clear correlation between the average number of infeasible incumbents returned and the epoch parameter.

The aim of the parameter optimisation experiment in this section was to obtain a suitable combination of parameters which may be used in the approximate solution approach (*i.e.* the method of SA) so as to obtain good GMS solutions for the IEEE-RTS with maximisation of expected energy production as scheduling criterion. In Figure 8.25(a), it may be observed that the large value for the cooling parameter (*i.e.* 0.95) achieves the largest mean optimality gap and that many outliers are observed in this case. The cooling parameter value of 0.95 was therefore eliminated from the parameter values considered as candidates for the final set of parameters in the context of the IEEE-RTS. It was furthermore observed that a cooling parameter of 0.90 results in an increase in the required computation time (an average of 20 047 seconds), which is significantly more than that associated with a cooling parameter of 0.85 (an average of 14 337 seconds). For this reason, this parameter value was also eliminated from the parameter values considered as candidates for the final set of parameters in respect of the IEEE-RTS. The only cooling parameter value left, is therefore 0.85, which achieves an acceptable mean optimality gap value as well as an acceptable mean computation time. This parameter value also returns, on average, fewer than five infeasible incumbents out of thirty.

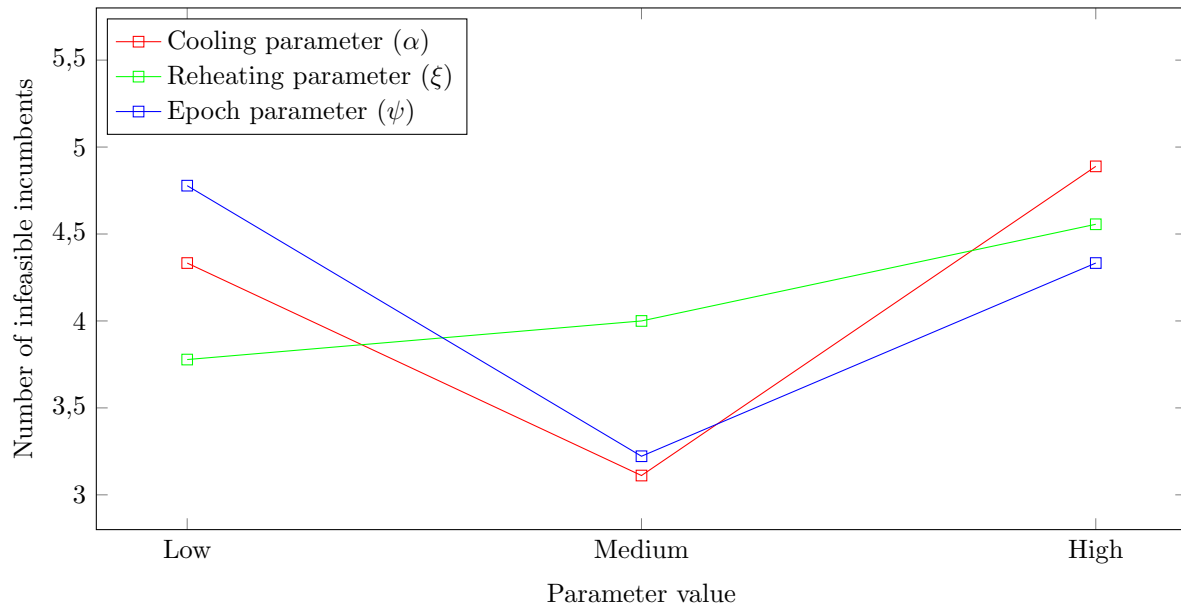


FIGURE 8.27: The mean number of infeasible incumbents (out of 30) as a function of the cooling parameter, the reheating parameter and the epoch parameter for the IEEE-RTS with maximisation of expected energy production as scheduling criterion.

Furthermore, the reheating parameter value of 0.75 was eliminated from the candidates considered due to the parameter value returning the largest number of outliers. The reheating parameter value of 0.95 was also eliminated from the parameter values considered as candidates for the final set of parameters due to the parameter value returning, on average, the largest number of infeasible solutions of the three reheating parameter values (an average of 4.556). Hence the cooling parameter value of 0.55 was selected as the best candidate for the IEEE-RTS. Finally, in terms of the results obtained for the epoch parameter it was observed that the value of 2 returned the largest number of outliers and was therefore eliminated from the parameter values considered as candidates for the final set of parameters in the context of the IEEE-RTS. Of the remaining epoch parameter values, the value 4 was selected as the final epoch parameter as this value resulted, on average, in a smaller required computation time. This value also returns an acceptable number of infeasible solutions (out of 30) on average. The final set of parameters adopted in the SA algorithm for solving the IEEE-RTS is shown in Table 8.22.

TABLE 8.22: The complete set of SA parameters selected for the IEEE-RTS with maximisation of expected energy production as scheduling criterion, as well as the mean optimality gap, the mean computation time required and the number of infeasible incumbents associated with these parameter values during the second phase of the parameter optimisation experiment.

χ_0	γ	α	ξ	ψ	Optimality gap (%)	Computation time (s)	Infeasibilities
0.6	1.25	0.85	0.55	4	2.060	24 201	3 (out of 30)

Numerical results

An approximate solution to the nonlinear GMS model of §4.3 was obtained for the IEEE-RTS by the SA algorithm with the parameter combinations as specified in Table 8.22. The decision variable values of the incumbent are given in integer decision vector form by $\mathbf{x} = [10, 4, 13, 32, 15, 51, 20, 29, 34, 31, 48, 23, 15, 35, 34, 38, 39, 40, 43, 22, 44, 17, 37, 50, 29, 41, 51, 49, 39,$

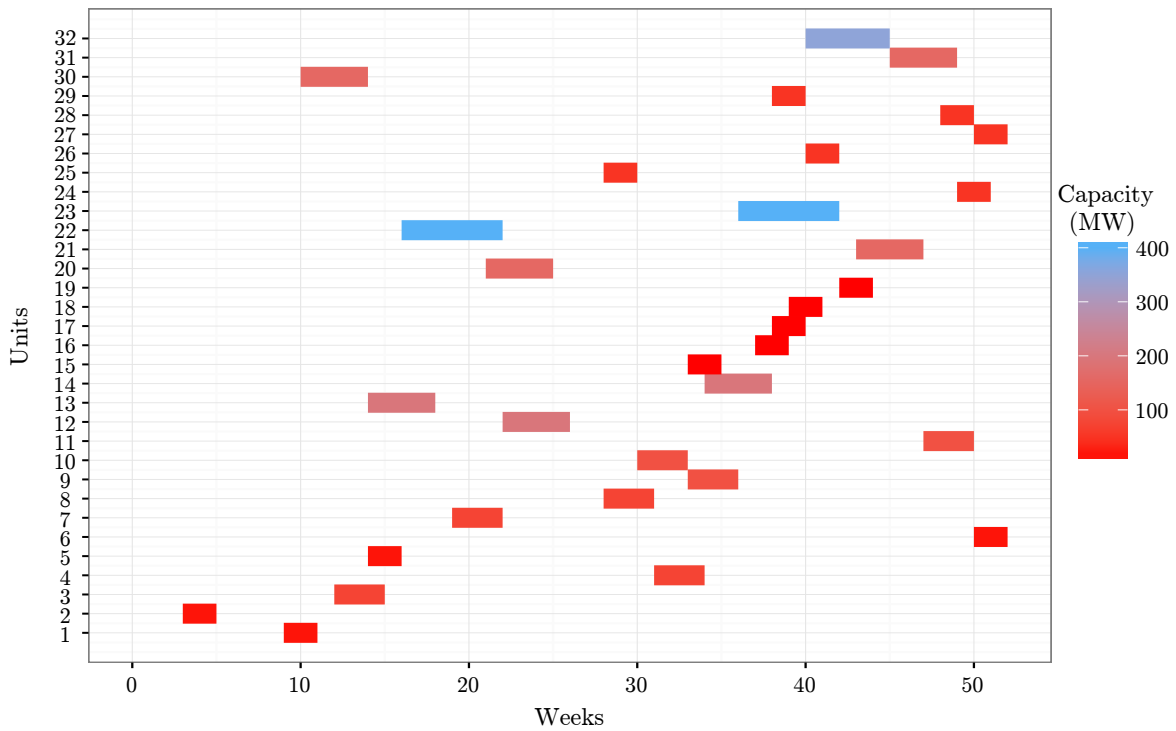
11, 46, 41], which corresponds to an objective function value of 128 516 MW·week (1.618% worse than that of the optimal piecewise linear approximation solution for the IEEE-RTS, as reported in §8.1.2).

A graphical representation of this maintenance schedule is presented in Figure 8.28 with the colour scale in Figure 8.28(a) indicating the rated capacity (in MW) of each PGU and the colour scale in Figure 8.28(b) indicating the failure rate of each PGU. The manpower required over the duration of the scheduling window in order to implement the solution in Figure 8.28 is shown in Figure 8.29(a), while the available system capacity over the duration of the scheduling window associated with this solution is shown in Figure 8.29(b).

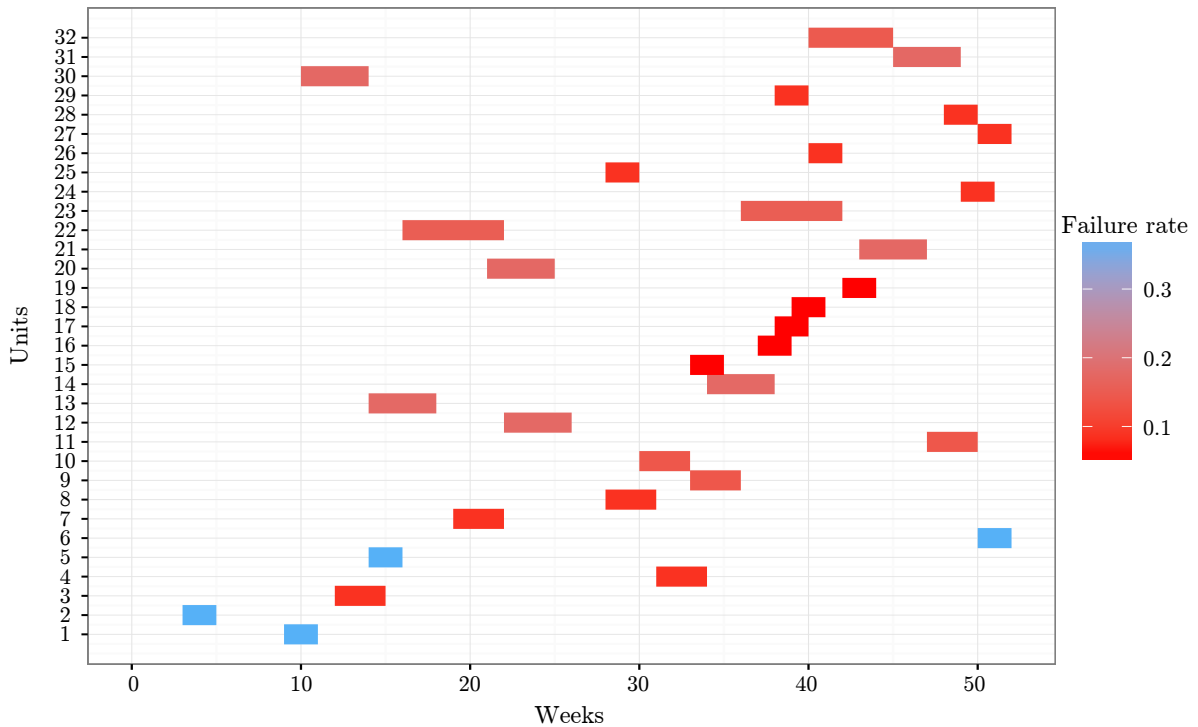
Comparison with results obtained by piecewise linear approximation

The solution reported above may be analysed by comparing it with the optimal piecewise linear approximation solution obtained in §8.1.2 for the IEEE-RTS with maximisation of expected energy production as scheduling criterion. A graphical representation of the incumbent solution returned by the approximate solution methodology of §5.3 (*i.e.* the method of SA) and the optimal piecewise linear approximation solution obtained by the exact solution approach described in §8.1.2 (*i.e.* by means of CPLEX) is shown in Figure 8.30 for the IEEE-RTS. The maintenance schedules of the two solutions are compared in the figure, with the colour scale indicating the rated capacity of the PGUs in the system. The effects of the two maintenance schedules on the manpower required and the available system capacity are shown in Figures 8.31(a) and 8.31(b), respectively.

Comparing these two maintenance schedules, only a small number of differences are observed in the scheduled commencement times of the PGUs, which causes the objective function values to differ by 1.618%. Comparing the available manpower in Figure 8.31(a), it is seen for the solution obtained by the piecewise linear approximation solution approach that no manpower is required during the early stages of the scheduling window after which it increases within a short period to the maximum available manpower. This is due to the proposed objective function scheduling PGUs with large rated capacities close to the peak of their expected energy production curves, resulting in these PGUs being scheduled for maintenance late during the scheduling window and the available manpower being low during the early stages of the scheduling window (and larger towards the end). A similar result is observed in the case of the metaheuristic solution. It is observed that the available manpower is low early on during the scheduling window and then gradually increases toward the end of the first half of the maintenance window. The major difference between these two solutions may be observed during the early stages of the scheduling window where many jumps are observed for the result obtained by the metaheuristic solution. In Figure 8.31(b), it is seen for the solution obtained by the piecewise linear approximation solution approach that the system capacity is equal to the maximum available capacity during the early stages of the maintenance window. This is due to the proposed objective function scheduling PGUs with large rated capacities close to the peaks of their expected energy production curves, resulting in these PGUs being scheduled for maintenance late during the scheduling window and the available system capacity being high. A similar result is observed in the case of the metaheuristic solution. It is observed that the system capacity is equal to the maximum available capacity early during the scheduling window, but a few jumps away from the maximum capacity is, however, observed and then the system capacity gradually decreases toward the end of the first half of the maintenance window. During the second half of the scheduling window, a similar available capacity pattern is observed when comparing the metaheuristic approach with the piecewise linear solution, with only small discrepancies being observed.

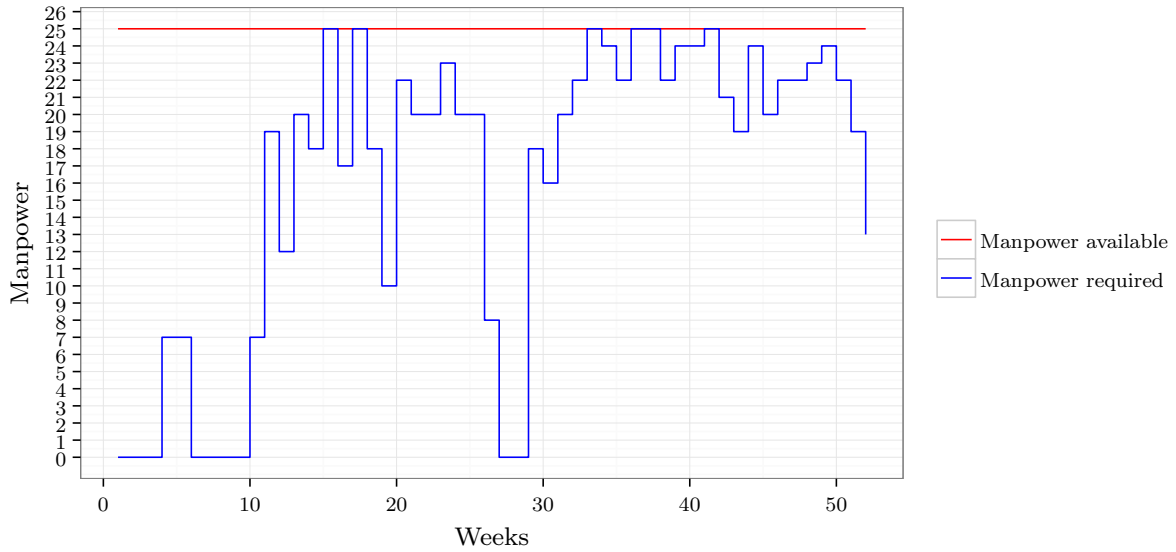


(a) The incumbent returned by the SA algorithm for the nonlinear model of §4.3, with the colour scale indicating the rated capacity of the PGUs

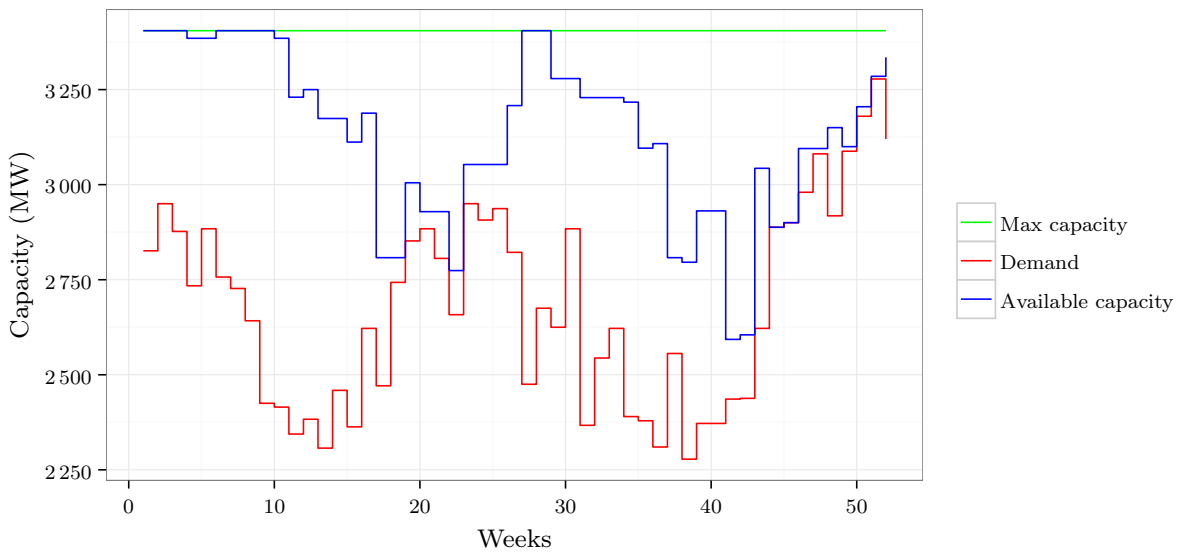


(b) The incumbent returned by the SA algorithm for the nonlinear model of §4.3, with the colour scale indicating the failure rate of the PGUs

FIGURE 8.28: The incumbent returned by the SA algorithm for the nonlinear model of §4.3, in the context of the IEEE-RTS with maximisation of expected energy production as scheduling criterion and for parameter values as indicated in Table 8.22.



(a) The manpower required over the scheduling window corresponding to the maintenance schedule in Figure 8.28



(b) The system capacity over the scheduling window corresponding to the maintenance schedule in Figure 8.28

FIGURE 8.29: Evaluation of the manpower required and system capacity available over the duration of the scheduling window for the IEEE-RTS maintenance schedule in Figure 8.28.

8.3 Chapter summary

In this chapter, the results obtained by employing a piecewise linear solution approach toward solving the nonlinear GMS model of §4.3 were reported in the contexts of the 21-unit test system [54] and the 32-unit IEEE-RTS [188]. The results for the 21-unit test system were presented in §8.1, which included an optimal maintenance schedule for the linearised problem instance, a comparison with a maintenance schedule according to another scheduling objective from the literature and an analysis of the practical feasibility of an exact solution approach in respect of the 21-unit test system in the form of a sensitivity analysis in §8.1.1. The same study

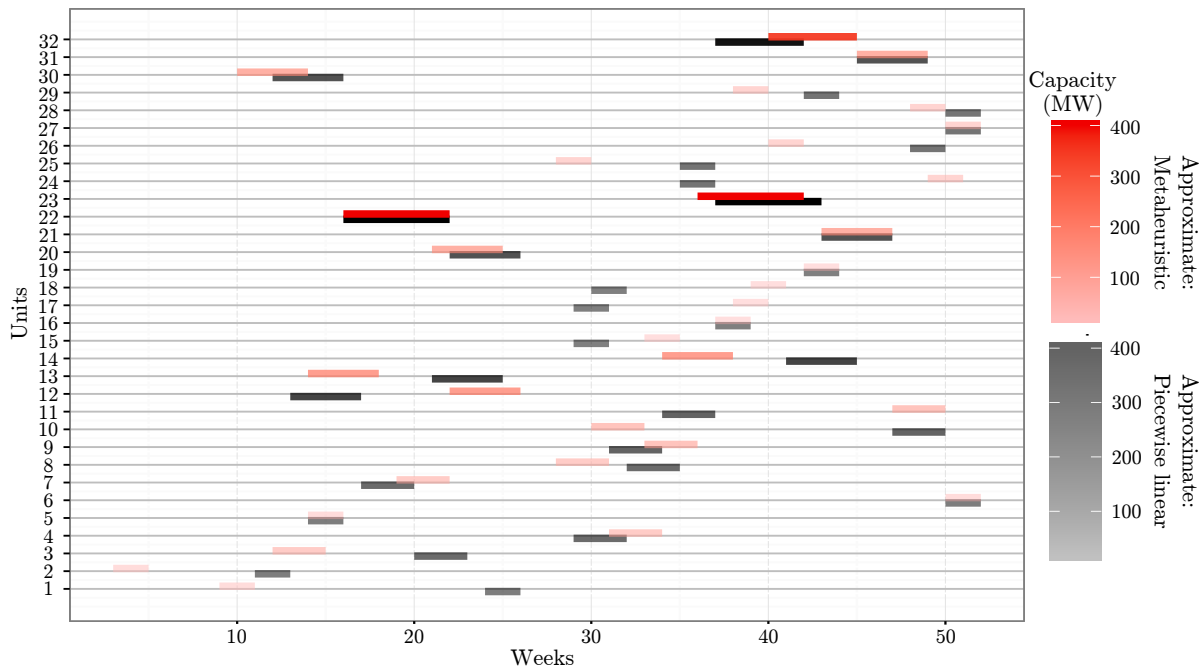
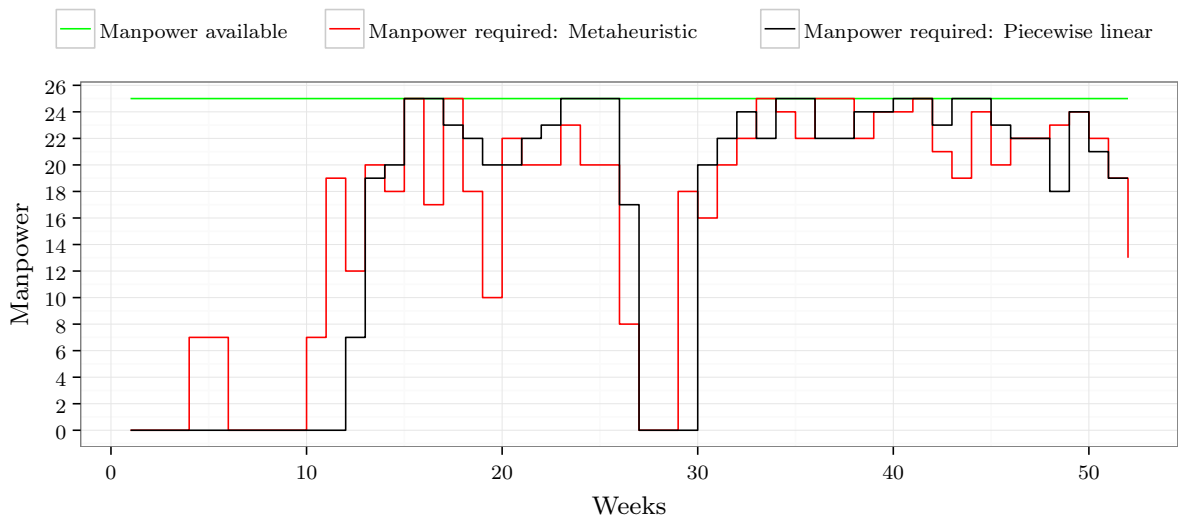


FIGURE 8.30: Two maintenance schedules for the IEEE-RTS returned by two different approaches towards solving the nonlinear GMS model of §4.3 with maximisation of expected energy production as scheduling criterion.

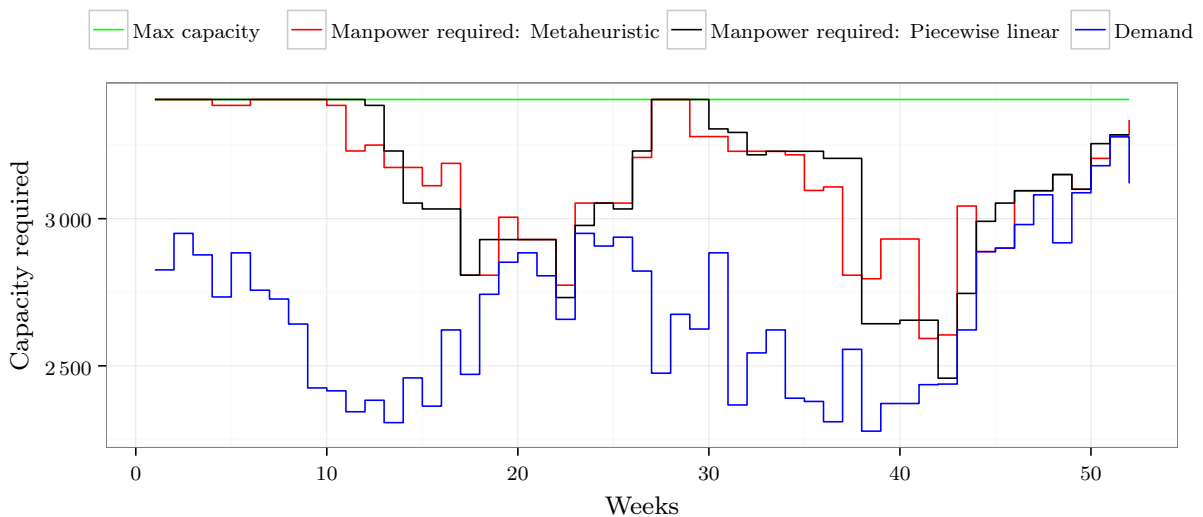
was performed for the IEEE-RTS, with the presentation of an optimal maintenance schedule for the linearised problem instance, a comparison with a maintenance schedule according to another scheduling objective from the literature and an analysis of the practical feasibility of the piecewise linear solution approach in respect of the 32-unit IEEE-RTS, again in the form of a sensitivity analysis in §8.1.2.

It was demonstrated that for small problem instances with tightly constrained maintenance window constraints, such as the original 21-unit test system, a piecewise linear solution approach is practically feasible for the nonlinear GMS model of §4.3 (*i.e.* requires a computation time of less than a hundred seconds). It was observed that a piecewise linear solution approach was not, however, feasible within the time-out budget (of 8 hours) for the IEEE-RTS. An optimal solution was only obtained after 25.983 hours of computation time. It was also observed, for both of these academic test systems, that an increase in the peak demand of the system does not affect the processing time significantly. Relaxation of the maintenance window constraints, however, has a dramatic influence on the computation time of the piecewise linear solution approach *via* CPLEX. It is therefore anticipated that adopting an exact solution approach toward solving the nonlinear GMS model of §4.3 in the context of a real-world problem instance, which may easily contain more than a hundred PGUs, will not be practically feasible. An approximate solution approach was consequently explored in §8.2 in order to be able to accommodate larger power systems.

The results obtained by the approximate solution approach adopted, the method of SA, was presented in §8.2 for the same two academic benchmark systems. This was followed by a presentation of the results obtained from a parameter optimisation experiment for the 21-unit test system in §8.2.1. In this section, the different combinations of parameter values were compared in order to obtain the best combination for the method of SA in the context of the 21-unit test system. This section also contained a description of the best incumbent returned by solving the 21-unit system upon utilisation of the best parameter combination values. A comparison



(a) The required manpower over the scheduling window corresponding to the maintenance schedules in Figure 8.30



(b) The system capacity over the scheduling window corresponding to the maintenance schedules in Figure 8.30

FIGURE 8.31: Comparison between the manpower required and the available system capacity associated with the schedules in Figure 8.20 for the IEEE-RTS with maximisation of expected energy production as scheduling criterion.

of this solution with the optimal solution obtained by the piecewise linear solution approach was also carried out in §8.1.1. It was found that the objective function values of the solution obtained by the piecewise linear solution approach and the approximate solution approach differ by only 2.669%. A similar approach was taken in respect of the IEEE-RTS in §8.2.2 where the results of the parameter optimisation experiment, as well as a comparison of the difference in the combinations of the parameter values, were presented. Finally, the best incumbent returned when solving the IEEE-RTS upon utilisation of the best parameter value combination was presented and a comparison was performed with the optimal solution obtained by the piecewise linear solution approach. For the IEEE-RTS, it was found that the objective function values of the solution obtained by the piecewise linear solution approach and that obtained by the approximate solution approach differ by 1.618%.

Part IV

Real-world model application

CHAPTER 9

Case study data

Contents

9.1	Background	209
9.2	Specifications	209
9.3	Objective function extensions	215
9.4	Chapter summary	224

The real-world case study considered in this dissertation part is called the *157-unit Eskom case study*. This case study involves the PGUs of the South African national power utility, Eskom. The data and constraints specified for the 157-unit Eskom case study are described in detail in this chapter along with some objective function extensions to the model of §4 in order to accommodate all maintenance requirements in such a large and complex real-world case study.

9.1 Background

Eskom is the state-owned national power utility in South Africa and is charged with the responsibility of generation, transmission, distribution, sales, importing and exporting of the country's electricity [75]. Eskom operates both base-load and peak PGUs, and supplies 96% of South Africa with electricity. The utility sells electricity to various customers of whom the largest is the combined municipalities within South Africa who distribute electricity to their respective end users.

The 157-unit Eskom case study, established by Schlünz and Van Vuuren [188] in 2011, is a considerably larger and more realistic GMS problem instance than the academic benchmark systems of §6 and is based on the true energy network of the South African national power utility. This case study is, however, merely a representation of the real Eskom power system. Due to confidentiality concerns, slight alterations have been made to certain data and the result is therefore not an exact representation of the true Eskom power system.

9.2 Specifications

The 157-unit Eskom case study has a total installed rated capacity of 39 949 MW, generated by 105 PGUs. These PGUs are indexed here by the set $\mathcal{U}_A = \{1, \dots, 105\}$. Some of these PGUs, however, require more than one maintenance procedure during the planning horizon (a one-year

scheduling window discretised into three hundred and sixty five one-day time periods). Since the 157-unit Eskom case study originally established by Schlünz and Van Vuuren [188] did not contain specified failure rates for the PGUs in the system, these failure rates were derived from the IEEE-RTS [8], which does include a specification of failure rates for PGUs based on the type of fuel used to generate energy and the rated capacities of the PGUs. The failure rates were generated for each type and size of PGU in the 157-unit Eskom case study by selecting a random failure rate according to a triangular distribution (between 15% below and 15% above the modes) with the failure rates specified in IEEE-RTS [8] as the modes of the distributions. The PGUs at Eskom's disposal in this case study each employs one of five technologies for generating electricity. These five different technologies are presented in Table 9.1. The table contains specifications of the number of power stations per technology as well as an indication of which PGUs employ which technology.

TABLE 9.1: *Description of the PGUs in the 157-unit Eskom case study.*

Description	Number of stations	PGUs
Nuclear	1	{1,2}
Coal subset 1	11	{3,...,54}
Coal subset 2	5	{55,...,80}
Gas-turbine	4	{81,...,93}
Conventional hydroelectric	2	{94,...,99}
Pumped storage schemes	3	{100,...,105}

The original 157-unit Eskom case study of Schlünz and Van Vuuren [188] also did not contain specifications of the number of days that had elapsed between the end of the previous PGU maintenance procedure and the start of the current planning window. These durations, denoted by $|x'_1|, \dots, |x'_{105}|$, were derived from a previous maintenance schedule employed by Eskom. It was assumed, for PGUs requiring two maintenance procedures per scheduling window, that the previous maintenance procedures had occurred within the second half of the previous year. For PGUs requiring three maintenance procedures per scheduling window, it was similarly assumed that the previous maintenance procedures had occurred within the last third of the previous year and for PGUs requiring four maintenance procedures per scheduling window, that the previous maintenance procedures had occurred within the last quarter of the previous year.

Table 9.2 contains the following data related to each PGU $u \in \mathcal{U}_A$ in the 157-unit Eskom case study: Its rated capacity C_u , its estimated failure rate λ_u , the number of days $|x'_u|$ assumed to have elapsed between completion of its previous maintenance procedure and the onset of the current scheduling window, its required number w_u of maintenance procedures during the current scheduling window, the durations $d_{u,1}, \dots, d_{u,w_u}$ of these maintenance procedures, and the earliest and latest starting times of these procedures in the form of intervals $[e_{u,1}, \ell_{u,1}], \dots, [e_{u,w_u}, \ell_{u,w_u}]$.

TABLE 9.2: *Specifications for the 105 actual PGUs of the 157-unit Eskom case study [188].*

\mathcal{U}_A	C_u (MW)	λ_u	$ x'_u $ (days)	w_u	$d_{u,i}$ (days)	$[e_{u,i}, \ell_{u,i}]$
1	900	0.01918	258	1	84	[225,282]
2	900	0.02084	302	1	0	[1,365]
3	615	0.01838	180	2	42, 7	[1,140],[183,359]
4	615	0.01933	127	1	7	[71,92]

Table 9.2 (continued): Specifications for the 105 actual PGUs of the 157-unit Eskom case study [188].

\mathcal{U}_A	C_u (MW)	λ_u	$ x'_u $ (days)	w_u	$d_{u,i}$ (days)	$[e_{u,i}, \ell_{u,i}]$
5	615	0.02003	48	1	35	[323,331]
6	615	0.01743	74	1	28	[43,85]
7	615	0.02025	183	1	28	[85,127]
8	615	0.01931	43	1	7	[190,211]
9	593	0.01925	289	1	28	[43,85]
10	593	0.01824	363	1	0	[1,365]
11	593	0.01980	311	1	28	[99,141]
12	593	0.02032	208	1	7	[29,50]
13	593	0.01960	64	2	7, 28	[1,175],[183,338]
14	593	0.01735	153	2	7, 28	[1,175],[183,338]
15	575	0.02449	317	1	0	[1,365]
16	575	0.02289	223	1	70	[71,148]
17	575	0.02197	353	1	14	[211,239]
18	575	0.02296	257	1	0	[1,365]
19	575	0.02424	60	1	70	[1,78]
20	575	0.02329	362	1	45	[232,288]
21	190	0.01188	235	1	0	[1,365]
22	185	0.01039	139	1	92	[1,78]
23	190	0.01196	102	1	0	[1,365]
24	190	0.01190	8	1	0	[92,120]
25	190	0.01160	3	1	14	[204,267]
26	190	0.01113	249	1	50	[1,365]
27	190	0.01106	268	1	0	[113,141]
28	190	0.01306	151	2	14, 28	[1,154],[183,352]
29	190	0.01032	236	1	14	[281,309]
30	190	0.01239	51	2	28, 28	[1,154],[183,338]
31	575	0.02432	177	1	7	[218,239]
32	575	0.02589	75	1	0	[1,365]
33	575	0.02542	325	1	0	[1,365]
34	575	0.02432	246	1	42	[1,50]
35	575	0.02244	90	1	7	[85,106]
36	575	0.02249	40	1	84	[239,282]
37	475	0.02350	152	2	21, 28	[1,161],[183,338]
38	475	0.02390	285	1	84	[1,78]
39	475	0.02393	83	1	21	[323,345]
40	475	0.02308	247	1	0	[1,365]
41	475	0.02580	56	1	0	[1,365]
42	475	0.02331	132	1	0	[1,365]
43	640	0.01751	139	1	23	[330,343]
44	640	0.01768	210	1	7	[1,22]
45	640	0.01976	158	1	23	[295,330]
46	640	0.02007	321	1	5	[323,344]
47	640	0.02011	45	1	57	[1,71]
48	640	0.02070	300	1	5	[85,106]
49	612	0.02031	53	1	28	[1,43]
50	612	0.01859	165	2	7, 56	[1,175],[183,310]

Table 9.2 (continued): Specifications for the 105 actual PGUs of the 157-unit Eskom case study [188].

U_A	C_u (MW)	λ_u	$ x'_u $ (days)	w_u	$d_{u,i}$ (days)	$[e_{u,i}, \ell_{u,i}]$
51	612	0.02058	21	1	7	[99,120]
52	669	0.01822	273	1	35	[64,113]
53	669	0.01994	6	1	7	[330,351]
54	669	0.01742	299	1	8	[351,358]
55	585	0.02220	6	2	4, 60	[1,178],[183,306]
56	585	0.02433	98	2	12, 4	[1,170],[183,362]
57	585	0.02339	0	2	4, 12	[1,178],[183,354]
58	585	0.02429	140	2	50, 4	[1,132],[183,362]
59	585	0.02302	270	1	4	[71,85]
60	585	0.02208	136	2	4, 4	[1,178],[183,362]
61	190	0.01162	130	1	0	[1,365]
62	190	0.01087	185	1	0	[1,365]
63	185	0.01185	77	1	30	[50,92]
64	180	0.01173	43	1	42	[218,267]
65	180	0.01223	150	1	0	[1,365]
66	160	0.01315	167	1	0	[1,365]
67	170	0.01094	230	1	42	[1,50]
68	180	0.01145	234	1	42	[85,134]
69	330	0.01130	140	2	4, 4	[1,178],[183,362]
70	350	0.02702	91	2	14, 4	[1,168],[183,362]
71	380	0.02259	67	2	4, 4	[1,178],[183,362]
72	350	0.02446	9	2	2, 3	[1,180],[183,363]
73	350	0.02305	99	2	4, 84	[1,178],[183,282]
74	350	0.02454	129	2	2, 3	[1,180],[183,363]
75	190	0.01137	251	1	21	[78,113]
76	190	0.01244	338	1	21	[211,246]
77	190	0.01272	171	1	0	[1,365]
78	190	0.01251	238	1	21	[309,344]
79	190	0.01281	357	1	0	[1,365]
80	190	0.01185	36	1	0	[1,365]
81	148	0.02335	110	1	0	[1,365]
82	148	0.02463	179	1	0	[1,365]
83	148	0.02405	215	1	0	[1,365]
84	148	0.02476	341	1	0	[1,365]
85	148	0.02467	248	1	0	[1,365]
86	148	0.02563	149	1	0	[1,365]
87	148	0.02194	146	1	0	[1,365]
88	57	0.04893	79	4	9, 2, 2, 2	[1,82],[92,180],[183,271],[274,364]
89	57	0.04671	29	4	2, 9, 2, 2	[1,89],[92,173],[183,271],[274,364]
90	57	0.04743	7	4	2, 9, 2, 7	[1,89],[92,173],[183,271],[274,359]
91	57	0.04528	63	4	2, 45, 2, 2	[1,89],[92,137],[183,271],[274,364]
92	57	0.04956	35	4	56, 2, 2, 2	[1,35],[92,180],[183,271],[274,364]
93	57	0.04964	77	3	7, 2, 2	[1,114],[122,240],[245,364]
94	90	0.01298	140	2	11, 120	[1,171],[183,246]
95	90	0.01234	62	2	2, 120	[1,180],[183,246]
96	90	0.01225	162	2	11, 14	[1,171],[183,352]

Table 9.2 (continued): Specifications for the 105 actual PGUs of the 157-unit Eskom case study [188].

\mathcal{U}_A	C_u (MW)	λ_u	$ x'_u $ (days)	w_u	$d_{u,i}$ (days)	$[e_{u,i}, \ell_{u,i}]$
97	90	0.01087	79	2	121, 2	[1,61],[183,364]
98	120	0.01154	106	3	21, 31, 5	[1,100],[122,211],[245,361]
99	120	0.01118	196	1	5	[85,106]
100	250	0.01168	57	4	44, 1, 1, 14	[1,47],[92,181],[183,272],[274,352]
101	250	0.01212	160	2	44, 1	[1,138],[183,365]
102	250	0.01167	30	2	1, 14	[1,181],[183,352]
103	250	0.01239	108	3	1, 3, 14	[1,120],[122,239],[245,352]
104	200	0.01234	27	3	25, 1, 1	[1,96],[122,241],[245,365]
105	200	0.01278	108	3	25, 1, 1	[1,96],[122,241],[245,365]

The reader will notice, from Table 9.2, that some of the PGUs in the system have maintenance duration specifications of zero days. This essentially means that during the current scheduling window these PGUs do not, in fact, require any maintenance. Twenty six PGUs (PGUs 2, 10, 15, 18, 21, 23, 26, 32, 33, 40, 41, 42, 61, 62, 65, 66, 77, 79, 80, 81, 82, 83, 84, 85, 86, 87) are therefore not required to be scheduled for maintenance within the current scheduling window. These PGU indices are captured in a set denoted by \mathcal{M} . Hence only the 79 PGUs in $\mathcal{U}_A \setminus \mathcal{M}$ require maintenance scheduling during the current scheduling window. The PGUs that are not scheduled for maintenance (*i.e.* those in \mathcal{M}), however, still contribute to the overall capacity of the power system.

The load demand of the 157-unit Eskom case study, presented in Table 9.3, exhibits a typical peak demand during the South African winter (with some level of offset due to confidentiality concerns) with the peak load of 36 664 MW being reached during Day 226. The load demand requires a safety margin of 8% for all 365 days, as indicated in Figure 9.1.

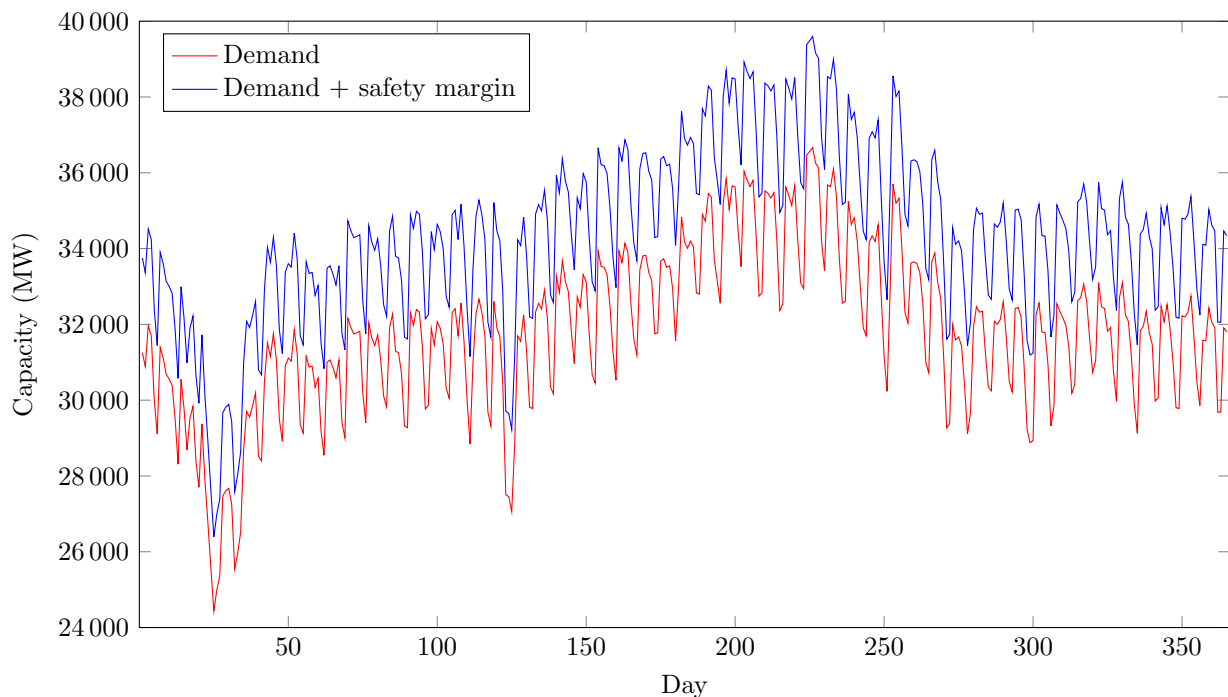


FIGURE 9.1: 157-unit Eskom case study demand, including a safety margin of 8% [188].

TABLE 9.3: Demand required per day for the 157-unit Eskom case study [188].

Day	Demand (MW)	Day	Demand (MW)	Day	Demand (MW)	Day	Demand (MW)	Day	Demand (MW)	Day	Demand (MW)
1	31 252	62	28 546	123	27 505	184	34 008	245	34 179	306	29 321
2	30 890	63	31 000	124	27 454	185	34 197	246	34 333	307	29 839
3	31 962	64	31 062	125	27 056	186	34 043	247	34 187	308	32 542
4	31 704	65	30 857	126	28 862	187	32 829	248	34 625	309	32 344
5	29 997	66	30 594	127	31 694	188	32 798	249	32 802	310	32 173
6	29 114	67	31 061	128	31 554	189	34 902	250	31 153	311	31 974
7	31 389	68	29 416	129	32 251	190	34 722	251	30 232	312	31 424
8	31 116	69	29 007	130	31 283	191	35 453	252	32 849	313	30 186
9	30 684	70	32 158	131	29 811	192	35 352	253	35 701	314	30 414
10	30 558	71	31 917	132	29 779	193	33 711	254	35 199	315	32 626
11	30 390	72	31 747	133	32 324	194	33 140	255	35 326	316	32 706
12	29 515	73	31 780	134	32 555	195	32 558	256	33 948	317	33 061
13	28 311	74	31 820	135	32 404	196	35 203	257	32 314	318	32 610
14	30 548	75	30 198	136	32 907	197	35 841	258	32 033	319	31 614
15	29 778	76	29 399	137	32 138	198	35 034	259	33 618	320	30 730
16	28 690	77	32 039	138	30 351	199	35 654	260	33 651	321	31 031
17	29 537	78	31 650	139	30 164	200	35 621	261	33 605	322	33 101
18	29 835	79	31 440	140	33 279	201	34 464	262	33 359	323	32 462
19	28 380	80	31 722	141	32 842	202	33 522	263	32 557	324	32 416
20	27 703	81	31 073	142	33 674	203	36 036	264	30 996	325	31 818
21	29 374	82	30 108	143	33 133	204	35 806	265	30 715	326	31 919
22	27 893	83	29 823	144	32 864	205	35 636	266	33 649	327	30 761
23	26 775	84	31 917	145	31 830	206	35 799	267	33 884	328	29 970
24	25 588	85	32 275	146	30 963	207	34 406	268	33 102	329	32 701
25	24 438	86	31 295	147	32 722	208	32 743	269	32 686	330	33 105
26	24 992	87	31 249	148	32 455	209	32 834	270	31 114	331	32 253
27	25 364	88	30 709	149	33 312	210	35 522	271	29 267	332	32 076
28	27 468	89	29 318	150	33 114	211	35 457	272	29 380	333	31 155
29	27 609	90	29 275	151	31 759	212	35 334	273	31 994	334	29 880
30	27 674	91	32 305	152	30 672	213	35 478	274	31 586	335	29 126
31	27 249	92	31 975	153	30 451	214	34 220	275	31 670	336	31 840
32	25 544	93	32 390	154	33 942	215	32 366	276	31 427	337	31 946
33	25 955	94	32 321	155	33 531	216	32 535	277	30 396	338	32 344
34	26 510	95	31 425	156	33 509	217	35 619	278	29 107	339	31 781
35	28 693	96	29 767	157	33 316	218	35 400	279	29 673	340	31 472
36	29 705	97	29 864	158	32 459	219	35 136	280	31 929	341	29 975
37	29 552	98	31 895	159	31 256	220	35 659	281	32 468	342	30 062
38	29 846	99	31 456	160	30 527	221	34 236	282	32 321	343	32 479
39	30 191	100	32 071	161	33 938	222	33 097	283	32 348	344	32 070
40	28 515	101	31 894	162	33 627	223	32 942	284	31 243	345	32 521
41	28 397	102	31 483	163	34 135	224	36 463	285	30 344	346	31 878
42	30 494	103	30 314	164	33 903	225	36 559	286	30 245	347	30 996
43	31 515	104	30 051	165	32 549	226	36 664	287	32 083	348	29 800
44	31 149	105	32 312	166	31 609	227	36 256	288	32 003	349	29 781
45	31 757	106	32 413	167	31 188	228	36 127	289	32 129	350	32 225
46	31 064	107	31 702	168	33 431	229	34 199	290	32 594	351	32 198
47	29 494	108	32 563	169	33 806	230	33 408	291	31 676	352	32 321
48	28 916	109	31 525	170	33 822	231	35 680	292	30 495	353	32 751
49	30 910	110	30 029	171	33 373	232	35 632	293	30 223	354	31 669
50	31 111	111	28 841	172	33 151	233	36 104	294	32 420	355	30 478
51	31 030	112	31 007	173	31 752	234	35 364	295	32 446	356	29 857
52	31 862	113	32 273	174	31 774	235	33 695	296	32 189	357	31 587
53	31 204	114	32 663	175	33 665	236	32 559	297	30 878	358	31 564
54	29 352	115	32 301	176	33 731	237	32 611	298	29 227	359	32 430
55	29 132	116	31 699	177	33 504	238	35 252	299	28 879	360	32 055
56	31 156	117	29 685	178	33 543	239	34 638	300	28 932	361	31 912
57	30 873	118	29 318	179	32 954	240	34 811	301	32 250	362	29 682
58	30 899	119	32 605	180	31 560	241	34 183	302	32 562	363	29 684
59	30 338	120	31 914	181	32 974	242	33 005	303	31 798	364	31 908
60	30 577	121	31 675	182	34 836	243	31 906	304	31 785	365	31 798
61	29 170	122	30 420	183	34 173	244	31 701	305	30 952		

9.3 Objective function extensions

In order to accommodate a case study of the complexity of the 157-unit Eskom case study, certain adaptations and extensions are required for both the minimisation of a measure of probability of unit failure and the maximisation of expected energy production GMS criteria of §4. These model adaptations and extensions are discussed in this section.

9.3.1 Minimising of probability of unit failure

As explained in the introduction to this chapter, Assumption 2 of §4.2 results in the undesirable situation where the requirement of multiple maintenance procedures being performed on any PGU in the power system cannot be accommodated in the current linear GMS model formulation of §4, although this is nevertheless required in the 157-unit Eskom case study. Instead of altering the model formulation in order to accommodate this requirement, a modelling construct of PGU duplication is rather adopted. This modelling construct involves the introduction of virtual, duplicate PGUs (which contribute no extra capacity to the total system capacity) in order to accommodate the requirement of multiple PGU maintenance instances — one for each additional maintenance procedure of every PGU over and above its standard single procedure. The total number of PGUs, including the virtual duplicates, is 157 and these PGUs are indexed by an enlarged set $\mathcal{U} = \{1, \dots, 157\}$. The set \mathcal{U} therefore includes both actual and virtual PGUs. The specifications for all the PGUs in the 157-unit Eskom case study, including these virtual copies, are presented in Table 9.4.

Recall that PGU $u \in \mathcal{U}_A$ has to be subjected to maintenance w_u times during the current scheduling window $[0, T]$ of length T . For each PGU $u \in \mathcal{U}_A$, the scheduling window is partitioned into w_u intervals of equal duration (T/w_u), referred to as *maintenance intervals*, and the convention is adopted that exactly one maintenance procedure for that PGU is scheduled in each of these maintenance intervals. Define the set $\mathcal{W}_u = \{1, \dots, w_u\}$ and let

$$\epsilon_{u,i} = (i-1) \frac{T}{w_u}, \quad i = 1, \dots, w_u + 1 \quad (9.1)$$

be a parameter specifying the starting time of the i -th maintenance interval of PGU $u \in \mathcal{U}_A$. Note that each index $i \in \mathcal{W}_u$ in (9.1) induces the starting time of a maintenance interval of PGU $u \in \mathcal{U}$ during the current scheduling window, while the final index value $w_u + 1$ corresponds to a maintenance interval starting time during the next scheduling window. The range $1, \dots, w_u + 1$ of index values is nevertheless convenient in (9.1), because the extremal values yield maintenance interval starting times coinciding with the start and end of the current scheduling window (*i.e.* $\epsilon_{u,1} = 0$ and $\epsilon_{u,w_u+1} = T$ for all $u \in \mathcal{U}_A$).

For any actual PGU $u \in \mathcal{U}_A$, the earliest starting time $e_{u,i}$ and the latest starting time $\ell_{u,i}$ of the i -th maintenance procedure is specified within the i -th maintenance interval $[\epsilon_{u,i}, \epsilon_{u,i+1}]$, for all $i \in \mathcal{W}_u$. These parameters correspond to the starting time $e'_u = e_{u,i}$ and latest starting time $\ell'_u = \ell_{u,i}$ of the i -th virtual PGU copy $u \in \mathcal{U}$. In this way, multiple maintenance procedures of actual PGUs (in \mathcal{U}_A) are accommodated while still allowing for only one maintenance procedure of each virtual PGU (in \mathcal{U}). An instance of the linear GMS model of §4.3 is therefore generated for the 157-unit Eskom case study in terms of PGUs in the enlarged set \mathcal{U} instead of for PGUs in the actual set \mathcal{U}_A . The objective function (4.6) of the linear GMS model of §4 is thus altered to

$$Y(\mathbf{X}) = \sum_{u \in \mathcal{U}, u \notin \mathcal{M}} \frac{C_u}{C_{\max}} \left[\lambda_u \left(\sum_{p \in \mathcal{P}} p x_{u,p} - (e'_u + x'_u) \right) \right], \quad (9.2)$$

TABLE 9.4: *PGU specifications and maintenance requirements for the 157-unit Eskom case study [188].*

PGU in \mathcal{U}	Actual PGU in \mathcal{U}_A	Rated capacity (MW)	Earliest starting time, e'_u	Latest starting time, ℓ'_u	Maintenance duration, d_u	Failure rate λ'_u	$ x'_u $	Maintenance procedures, w_u
1	1	900	225	282	84	0.01918	258	1
2	2	900	1	365	0	0.02084	302	1
3	3	615	1	140	42	0.01838	180	2
4	3	615	182	359	7	0.01838	180	2
5	4	615	71	92	7	0.01933	127	1
6	5	615	323	331	35	0.02003	48	1
7	6	615	43	85	28	0.01743	74	1
8	7	615	85	127	28	0.02025	183	1
9	8	615	190	211	7	0.01931	43	1
10	9	593	43	85	28	0.01925	289	1
11	10	593	1	365	0	0.01824	363	1
12	11	593	99	141	28	0.01980	311	1
13	12	593	29	50	7	0.02032	208	1
14	13	593	1	175	7	0.01960	64	2
15	13	593	182	338	28	0.01960	64	2
16	14	593	1	175	7	0.01735	153	2
17	14	593	182	338	28	0.01735	153	2
18	15	575	1	365	0	0.02449	317	1
19	16	575	71	148	70	0.02289	223	1
20	17	575	211	239	14	0.02197	353	1
21	18	575	1	365	0	0.02296	257	1
22	19	575	1	78	70	0.02424	60	1
23	20	575	232	288	45	0.02329	362	1
24	21	190	1	365	0	0.01188	235	1
25	22	185	1	78	92	0.01039	139	1
26	23	190	1	365	0	0.01196	102	1
27	24	190	92	120	14	0.01190	8	1
28	25	190	204	267	50	0.01160	3	1
29	26	190	1	365	0	0.01113	249	1
30	27	190	113	141	14	0.01106	268	1
31	28	190	1	154	28	0.01306	151	2
32	28	190	182	352	14	0.01306	151	2
33	29	190	281	309	14	0.01032	236	1
34	30	190	1	154	28	0.01239	51	2
35	30	190	182	338	28	0.01239	51	2
36	31	575	218	239	7	0.02432	177	1
37	32	575	1	365	0	0.02589	75	1
38	33	575	1	365	0	0.02542	325	1
39	34	575	1	50	42	0.02432	246	1
40	35	575	85	106	7	0.02244	90	1
41	36	575	239	282	84	0.02249	40	1
42	37	475	1	161	21	0.02350	152	2
43	37	475	182	338	28	0.02350	152	2
44	38	475	1	78	84	0.02390	285	1

Table 9.4 (continued): PGU specifications and maintenance requirements for the 157-unit Eskom case study [188].

PGU in \mathcal{U}	Actual PGU in \mathcal{U}_A	Rated capacity (MW)	Earliest starting time, e'_u	Latest starting time, ℓ'_u	Maintenance duration, d_u	Failure rate λ'_u	$ x'_u $	Maintenance procedures, w_u
45	39	475	323	345	21	0.02393	83	1
46	40	475	1	365	0	0.02308	247	1
47	41	475	1	365	0	0.02580	56	1
48	42	475	1	365	0	0.02331	132	1
49	43	640	330	343	23	0.01751	139	1
50	44	640	1	22	7	0.01768	210	1
51	45	640	295	330	23	0.01976	158	1
52	46	640	323	344	5	0.02007	321	1
53	47	640	1	71	57	0.02011	45	1
54	48	640	85	106	5	0.02070	300	1
55	49	612	1	43	28	0.02031	53	1
56	50	612	1	175	7	0.01859	165	2
57	50	612	182	310	56	0.01859	165	2
58	51	612	99	120	7	0.02058	21	1
59	52	669	64	113	35	0.01822	273	1
60	53	669	330	351	7	0.01994	6	1
61	54	669	351	358	8	0.01742	299	1
62	55	585	1	178	4	0.02220	5	2
63	55	585	182	306	60	0.02220	6	2
64	56	585	1	170	12	0.02433	98	2
65	56	585	182	362	4	0.02433	98	2
66	57	585	1	178	4	0.02339	0	2
67	57	585	182	354	12	0.02339	0	2
68	58	585	1	132	50	0.02429	140	2
69	58	585	182	362	4	0.02429	140	2
70	59	585	71	85	4	0.02302	270	1
71	60	585	1	178	4	0.02208	136	2
72	60	585	182	362	4	0.02208	136	2
73	61	190	1	365	0	0.01162	130	1
74	62	190	1	365	0	0.01087	185	1
75	63	185	50	92	30	0.01185	77	1
76	64	180	218	267	42	0.01173	43	1
77	65	180	1	365	0	0.01223	150	1
78	66	160	1	365	0	0.01315	167	1
79	67	170	1	50	42	0.01094	230	1
80	68	180	85	134	42	0.01145	234	1
81	69	330	1	178	4	0.01130	140	2
82	69	330	182	362	4	0.01130	140	2
83	70	350	1	168	14	0.02702	91	2
84	70	350	182	362	4	0.02702	91	2
85	71	380	1	178	4	0.02259	67	2
86	71	380	182	362	4	0.02259	67	2
87	72	350	1	180	2	0.02446	9	2
88	72	350	182	363	3	0.02446	9	2

Table 9.4 (continued): PGU specifications and maintenance requirements for the 157-unit Eskom case study [188].

PGU in \mathcal{U}	Actual PGU in \mathcal{U}_A	Rated capacity (MW)	Earliest starting time, e'_u	Latest starting time, ℓ'_u	Maintenance duration, d_u	Failure rate λ'_u	$ x'_u $	Maintenance procedures, w_u
89	73	350	1	178	4	0.02305	99	2
90	73	350	182	282	84	0.02305	99	2
91	74	350	1	180	2	0.02454	129	2
92	74	350	182	363	3	0.02454	129	2
93	75	190	78	113	21	0.01137	251	1
94	76	190	211	246	21	0.01244	338	1
95	77	190	1	365	0	0.01272	171	1
96	78	190	309	344	21	0.01251	238	1
97	79	190	1	365	0	0.01281	357	1
98	80	190	1	365	0	0.01185	36	1
99	81	148	1	365	0	0.02335	110	1
100	82	148	1	365	0	0.02463	179	1
101	83	148	1	365	0	0.02405	215	1
102	84	148	1	365	0	0.02476	341	1
103	85	148	1	365	0	0.02467	248	1
104	86	148	1	365	0	0.02563	149	1
105	87	148	1	365	0	0.02194	146	1
106	88	57	1	82	9	0.04893	79	4
107	88	57	91	180	2	0.04893	79	4
108	88	57	182	271	2	0.04893	79	4
109	88	57	272	364	2	0.04893	79	4
110	89	57	1	89	2	0.04671	29	4
111	89	57	91	173	9	0.04671	29	4
112	89	57	182	271	2	0.04671	29	4
113	89	57	272	364	2	0.04671	29	4
114	90	57	1	89	2	0.04743	7	4
115	90	57	91	173	9	0.04743	7	4
116	90	57	182	271	2	0.04743	7	4
117	90	57	272	359	7	0.04743	7	4
118	91	57	1	89	2	0.04528	63	4
119	91	57	91	137	45	0.04528	63	4
120	91	57	182	271	2	0.04528	63	4
121	91	57	272	364	2	0.04528	63	4
122	92	57	1	35	56	0.04956	35	4
123	92	57	91	180	2	0.04956	35	4
124	92	57	182	271	2	0.04956	35	4
125	92	57	272	364	2	0.04956	35	4
126	93	57	1	114	7	0.04964	77	3
127	93	57	121	240	2	0.04964	77	3
128	93	57	242	364	2	0.04964	77	3
129	94	90	1	171	11	0.01298	140	2
130	94	90	182	246	120	0.01298	140	2
131	95	90	1	180	2	0.01234	62	2
132	95	90	182	246	120	0.01234	62	2

Table 9.4 (continued): PGU specifications and maintenance requirements for the 157-unit Eskom case study [188].

PGU in \mathcal{U}	Actual PGU in \mathcal{U}_A	Rated capacity (MW)	Earliest starting time, e'_u	Latest starting time, ℓ'_u	Maintenance duration, d_u	Failure rate λ'_u	$ x'_u $	Maintenance procedures, w_u
133	96	90	1	171	11	0.01225	162	2
134	96	90	182	352	14	0.01225	162	2
135	97	90	1	61	121	0.01087	79	2
136	97	90	182	364	2	0.01087	79	2
137	98	120	1	100	21	0.01154	106	3
138	98	120	121	211	31	0.01154	106	3
139	98	120	242	361	5	0.01154	106	3
140	99	120	85	106	5	0.01118	196	1
141	100	250	1	47	44	0.01168	57	4
142	100	250	91	181	1	0.01168	57	4
143	100	250	182	272	1	0.01168	57	4
144	100	250	272	352	14	0.01168	57	4
145	101	250	1	138	44	0.01212	160	2
146	101	250	182	365	1	0.01212	160	2
147	102	250	1	181	1	0.01167	30	2
148	102	250	182	352	14	0.01167	30	2
149	103	250	1	120	1	0.01239	108	3
150	103	250	121	239	3	0.01239	108	3
151	103	250	242	352	14	0.01239	108	3
152	104	200	1	96	25	0.01234	27	3
153	104	200	121	241	1	0.01234	27	3
154	104	200	242	365	1	0.01234	27	3
155	105	200	1	96	25	0.01278	108	3
156	105	200	121	241	1	0.01278	108	3
157	105	200	242	365	1	0.01278	108	3

for the PGUs in $\mathcal{U}_A \setminus \mathcal{M}$, where the meanings of the variables $x_{u,p}$ and the parameters C_{\max} and λ_u are as before. Recall that no PGU $u \in \mathcal{M}$ requires maintenance within the current scheduling window. These PGUs, however, still have a probability of failure within the current scheduling window, but this probability is not a function of any maintenance schedule. These constant probabilities may be determined by calculating the probability of PGU failure from the end of the previous maintenance procedure x'_u until the end of the current scheduling window T . The sum of these probabilities for all PGUs in the set \mathcal{M} (which do not require any maintenance within the current scheduling window) is given by

$$M = \sum_{u \in \mathcal{M}} \frac{C_u}{C_{\max}} \left[\lambda_u (T - (e'_u + x'_u)) \right].$$

The objective function (4.6) of the linear GMS model of §4 thus finally becomes

$$N(\mathbf{X}) = Y(\mathbf{X}) + M, \tag{9.3}$$

where $Y(\mathbf{X})$ is as defined in (9.2).

A difficulty, however, arises when attempting to estimate the reliability function of a PGU in the modelling approach described above. This difficulty arises because of the fact each PGU

reliability function is considered a model parameter function — it does not contain any variables. When the i -th maintenance procedure has to be performed on some PGU $u \in \mathcal{U}_A$, with $i > 1$, then the desirability of its timing $k_{u,i}$ unfortunately depends on when the PGU was last returned into operation after maintenance (*i.e.* on the time $k_{u,i-1} + d_{u,i-1} - 1$). But this date is not known in advance (*i.e.* before the model is solved), because $k_{u,i-1}$ is a model variable.

In order to remedy this problem, the length $|x'_u|$ of time that has elapsed between the previous maintenance procedure of a PGU $u \in \mathcal{U}_A$ and the onset of its earliest maintenance starting time $\epsilon_{u,i}$ is therefore assumed to be the same for all of its maintenance intervals in the current scheduling window. The time period during which PGU u is 100% reliable in its i -th maintenance interval is consequently estimated to be $\epsilon_{u,i} + x'_u$ while, in fact, it is 100% reliable after the PGU returns to operation after completion of its $(i - 1)$ -th maintenance procedure. This discrepancy is illustrated in Figure 9.2

9.3.2 Maximising expected energy production

A different modelling approach to that described in §9.3.1 is adopted in the maximisation of the expected energy production scheduling criterion extension in respect of accommodating multiple maintenance procedures per PGU. In this extension to the nonlinear objective function of §4.3, a subset of PGUs is required which contains real PGUs (*e.g.* the set should only include actual PGUs). The reason why a model construct of duplicating PGUs cannot be employed in this case in order to accommodate the requirement of multiple maintenance procedures for certain PGUs is that this would cause a skewed representation of the expected amount of energy generated in the system. This was not a problem in §9.3.1, because there it was merely assumed in the demand satisfaction constraint that virtual copies of PGUs have zero rated capacity. Therefore, the set \mathcal{U}_A of actual PGUs is applicable in the nonlinear model of §4.3 instead of the set \mathcal{U} . The model is therefore altered explicitly to accommodate multiple PGU maintenance procedures during the scheduling window. For this model extension, it is still assumed, as per Assumption 3, that each PGU will fail no more than once during the current scheduling window. Let $k_{u,i}$ be the starting date of the i -th maintenance procedure performed on PGU $u \in \mathcal{U}_A$.

The three cases described in §4.3.2 are still considered in this more general setting, although in a slightly altered form. The three cases become:

- I A failure of PGU $u \in \mathcal{U}_A$ occurs at some time X_u within maintenance interval $i \in \mathcal{W}_u$, but before planned maintenance is performed on the PGU during that maintenance interval, and maintenance is either performed when the failure occurs or sometime thereafter, but still during maintenance interval i (*i.e.* $\epsilon_{u,i} \leq X_u \leq k_{u,i} \leq \epsilon_{u,i+1}$, where $\epsilon_{u,i}$ is as defined in (9.1)), or
- II a failure of PGU $u \in \mathcal{U}_A$ occurs at some time X_u within maintenance interval $i \in \mathcal{W}_u$, but planned maintenance on the PGU is completed before the failure occurs (*i.e.* $k_{u,i} + d_{u,i} - 1 < X_u \leq \epsilon_{u,i+1}$), or
- III the entire scheduling window elapses before a failure of PGU $u \in \mathcal{U}_A$ occurs, *i.e.* $T < X_u$.

The total energy produced by PGU $u \in \mathcal{U}_A$ over the scheduling window in Case I is the energy produced by the PGU during maintenance interval $i \in \mathcal{W}_u$, in which the failure is observed, together with the energy produced by the PGU during all the maintenance intervals other than interval i . Therefore, the total energy produced by PGU $u \in \mathcal{U}_A$ over the entire scheduling window in Case I is

$$A_{u,i}(X_u) = C_u[(X_u - \epsilon_{u,i}) + ((\epsilon_{u,i+1} - 1) - (k_{u,i} + d_{u,i} - 1))] + R_{u,i}, \quad (9.4)$$

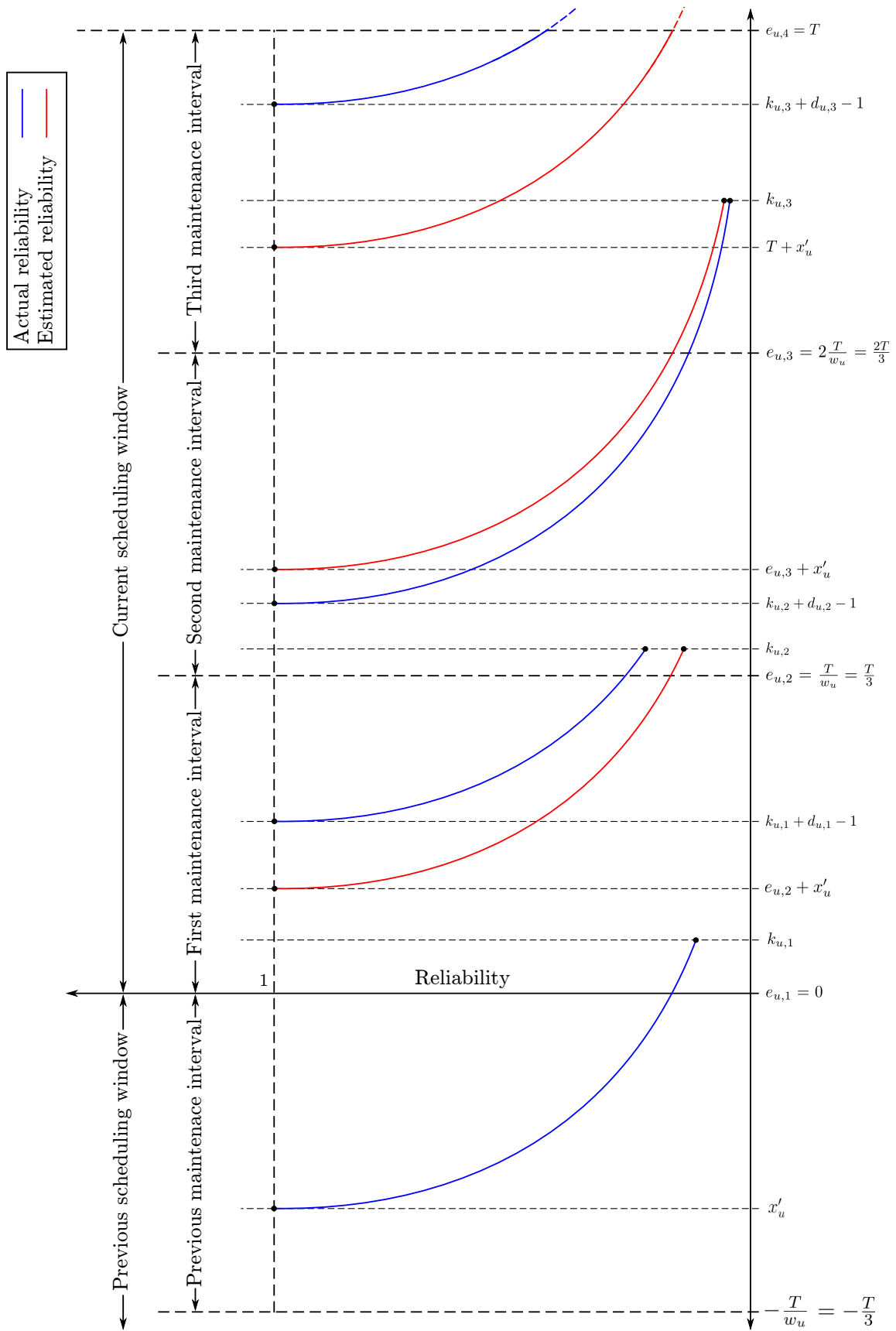


FIGURE 9.2: The actual and estimated reliability of a PGU $u \in \mathcal{U}$ which requires maintenance three times (i.e. $w_u = 3$) within the current scheduling window.

where

$$R_{u,i} = \sum_{j=1, j \neq i}^{w_u} C_u [(k_{u,j} - \epsilon_{u,j}) + (\epsilon_{u,j+1} - 1 - (k_{u,j} + d_{u,j} - 1))] \quad (9.5)$$

denotes the total energy produced by PGU $u \in \mathcal{U}_A$ within all failure-free maintenance intervals of the scheduling window $[0, T]$, that is, during all maintenance intervals other than the maintenance interval $i \in \mathcal{W}_u$. The quantity in the first term of (9.4) is illustrated graphically as the shaded area in Figure 9.3. The probability that a failure of PGU $u \in \mathcal{U}_A$ occurs at time t during the maintenance interval $i \in \mathcal{W}_u$ before scheduled maintenance takes place during that maintenance interval is $P(X_u = t) = \lambda_u e^{-\lambda_u(t - (\epsilon_{u,i} + x'_u))}$ for some $t \in [\epsilon_{u,i}, k_{u,i}]$.

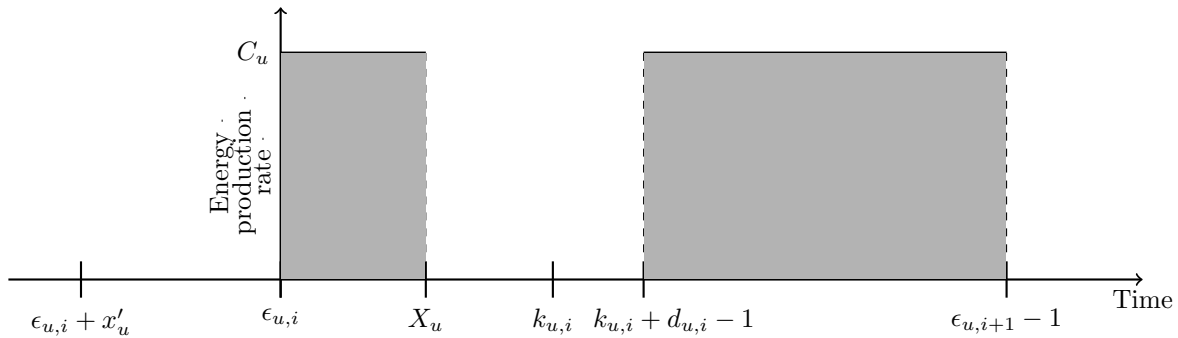


FIGURE 9.3: A failure of PGU $u \in \mathcal{U}_A$ occurs within maintenance interval $i \in \mathcal{W}_u$, but before planned maintenance is performed on the PGU during that maintenance interval, and maintenance is either performed when the failure occurs or sometime thereafter, but still during maintenance interval i (i.e. $\epsilon_{u,i} \leq X_u \leq k_{u,i} \leq \epsilon_{u,i+1}$) in Case I. The total energy produced during maintenance interval i is represented by the shaded area.

The total expected energy produced by PGU $u \in \mathcal{U}_A$ over the entire scheduling window in Case I above is therefore

$$\begin{aligned} E_{u,i}^I &= \int_{\epsilon_{u,i}}^{k_{u,i}} A_{u,i}(t) P(X_u = t) dt \\ &= \int_{\epsilon_{u,i}}^{k_{u,i}} C_u [(t - \epsilon_{u,i}) + ((\epsilon_{u,i+1} - 1) - (k_{u,i} + d_{u,i} - 1))] \lambda_u e^{-\lambda_u(t - (\epsilon_{u,i} + x'_u))} dt \\ &\quad + R_{u,i} \int_{\epsilon_{u,i}}^{k_{u,i}} \lambda_u e^{-\lambda_u(t - (\epsilon_{u,i} + x'_u))} dt. \end{aligned} \quad (9.6)$$

The total energy produced by PGU $u \in \mathcal{U}_A$ over the scheduling window in Case II is the energy produced by the PGU during maintenance interval $i \in \mathcal{W}_u$, in which the failure is observed, together with the energy produced by the PGU during all the maintenance intervals other than interval i . Therefore, the total energy produced by PGU $u \in \mathcal{U}_A$ over the entire scheduling window in Case II is

$$B_{u,i}(X_u) = C_u [(k_{u,i} - \epsilon_{u,i}) + ((X_u - (k_{u,i} + d_{u,i} - 1)))] + R_{u,i}, \quad (9.7)$$

where $R_{u,i}$ is as defined in (9.5). The quantity in the first term of (9.7) is illustrated graphically as the shaded area in Figure 9.4. The probability that a failure occurs at time t during maintenance interval $i \in \mathcal{W}_u$ after scheduled maintenance takes place during that maintenance interval is $P(X_u = t) = \lambda_u e^{-\lambda_u(t - (k_{u,i} + d_{u,i} - 1))}$ for some $t \in (k_{u,i} + d_{u,i} - 1, \epsilon_{u,i+1} - 1]$.

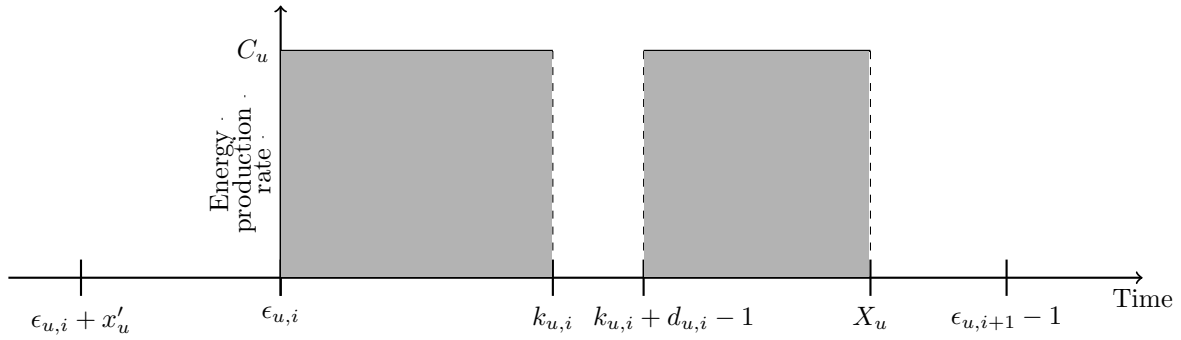


FIGURE 9.4: A failure of PGU $u \in \mathcal{U}_A$ occurs within maintenance interval $i \in \mathcal{W}_u$, but planned maintenance on the PGU is completed before the failure occurs (i.e. $k_{u,i} + d_{u,i} - 1 < X_u \leq \epsilon_{u,i+1}$) in Case II. The total energy produced during maintenance interval i is represented by the shaded area.

The total expected energy produced by PGU $u \in \mathcal{U}_A$ over the entire scheduling window in Case II above is therefore

$$\begin{aligned}
 E_{u,i}^{\text{II}} &= \int_{k_{u,i} + d_{u,i} - 1}^{\epsilon_{u,i+1} - 1} B_{u,i}(t) P(X_u = t) dt \\
 &= \int_{k_{u,i} + d_{u,i} - 1}^{\epsilon_{u,i+1} - 1} C_u [(k_{u,i} - \epsilon_{u,i}) + (t - (k_{u,i} + d_{u,i} - 1))] \lambda_u e^{-\lambda_u(t - (k_{u,i} + d_{u,i} - 1))} dt \\
 &\quad + R_{u,i} \int_{k_{u,i} + d_{u,i} - 1}^{\epsilon_{u,i+1} - 1} \lambda_u e^{-\lambda_u(t - (k_{u,i} + d_{u,i} - 1))} dt.
 \end{aligned} \tag{9.8}$$

The total energy produced by PGU $u \in \mathcal{U}_A$ over the scheduling window in Case III is the energy produced by the PGU during maintenance interval $j \in \mathcal{W}_u$, summed over all $j \in \mathcal{W}_u$. Therefore, the total energy produced by PGU $u \in \mathcal{U}_A$ over the entire scheduling window in Case III is

$$R_u = \sum_{j=1}^{w_u} C_u [(k_{u,j} - \epsilon_{u,j}) + (\epsilon_{u,j+1} - 1 - (k_{u,j} + d_{u,j} - 1))]. \tag{9.9}$$

The quantity in the first term of (9.9) is illustrated graphically as the shaded area in Figure 9.5. The probability that a failure occurs at time t after the scheduling window has ended is $P(X_u = t) = \lambda_u e^{-\lambda_u(t - (k_{u,w_u} + d_{u,w_u} - 1))}$ for some $t \in [T, \infty)$.

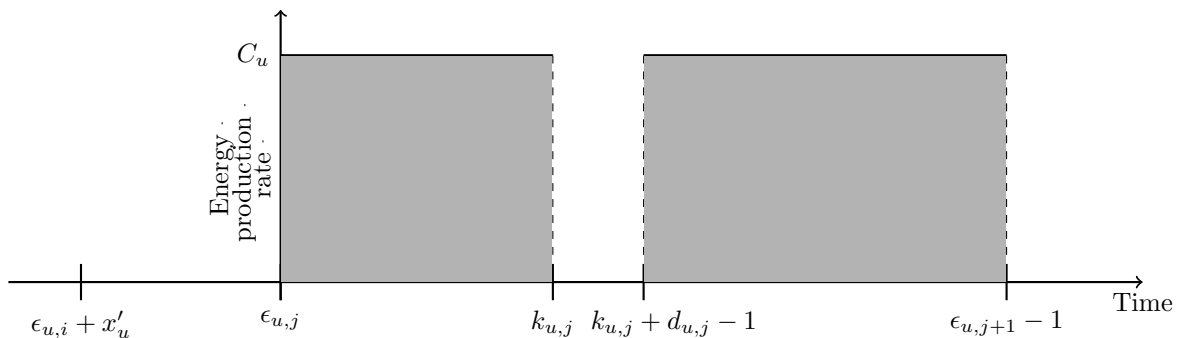


FIGURE 9.5: The entire scheduling window elapses before a failure of PGU $u \in \mathcal{U}_A$ occurs (i.e. $T < X_u$) in Case III. The total energy produced during maintenance interval $j \in \mathcal{W}_u$ is represented by the shaded area.

The total expected energy produced by PGU $u \in \mathcal{U}_A$ over the entire scheduling window in Case III above is therefore

$$\begin{aligned} E_{u,i}^{\text{III}} &= \int_T^\infty R_u(t)P(X_u = t) dt \\ &= R_u \int_T^\infty \lambda_u e^{-\lambda_u(t-(k_{u,w_u}+d_{u,w_u}-1))} dt. \end{aligned}$$

Hence the expected total energy produced by PGU $u \in \mathcal{U}_A$ during the entire scheduling window is

$$\sum_{i \in \mathcal{W}_u} (E_{u,i}^{\text{I}} + E_{u,i}^{\text{II}} + E_{u,i}^{\text{III}}). \quad (9.10)$$

Recall, again, that no PGU $u \in \mathcal{M}$ requires maintenance within the current scheduling window. These PGUs are still, however, expected to produce energy over the scheduling window. The expected production of energy by these PGUs is not influenced by maintenance and therefore the expected energy is constant for any maintenance schedule. This constant expected energy may be determined by calculating the energy generated by these PGUs (on which maintenance is performed) multiplied by the probability of PGU failure. The sum of these constants for all PGUs in the set \mathcal{M} is given by

$$\begin{aligned} G &= \sum_{u \in \mathcal{M}} \sum_{i \in \mathcal{W}_u} C_u [(\epsilon_{u,i+1} - 1) - \epsilon_{u,i}] \int_0^\infty \lambda_u e^{-\lambda_u(t-(\epsilon_{u,i}+x'_u))} dt \\ &= \sum_{u \in \mathcal{M}} \sum_{i \in \mathcal{W}_u} C_u [(\epsilon_{u,i+1} - 1) - \epsilon_{u,i}] e^{\lambda_u(\epsilon_{u,i}+x'_u)}. \end{aligned} \quad (9.11)$$

The extended objective function of the nonlinear GMS model of §4 thus finally becomes

$$\sum_{u \in \mathcal{U}_A} \sum_{i \in \mathcal{W}_u} (E_{u,i}^{\text{I}} + E_{u,i}^{\text{II}} + E_{u,i}^{\text{III}}) + G. \quad (9.12)$$

An example of the expected energy produced by PGU 105 of the 157-unit Eskom case study, which requires three maintenance procedures, is shown in Figure 9.6. The expected amount of energy produced during the three maintenance intervals of PGU 105 are different due to the fact that the durations of the three maintenance procedures differ, as do the earliest and latest possible starting times of these maintenance procedures.

9.4 Chapter summary

This chapter contained a description of a case study by which the effectiveness of the GMS models proposed in §4 and the solution approach described in §5 is to be tested in the next chapter of this dissertation. This study is called the 157-unit Eskom case study. The case study comprises a power generating system containing 105 PGUs of which certain PGUs require multiple maintenance procedures within the scheduling window. Duplicate virtual PGUs, were created for these PGUs in one of the GMS models, resulting in a total of 157 PGUs in the case study.

A short background was provided in §9.1 on the national power utility, Eskom, on which the case study is based. This was followed by a detailed specification of the 157-unit Eskom case study in §9.2, which included data on the daily demand over a one-year scheduling window. The

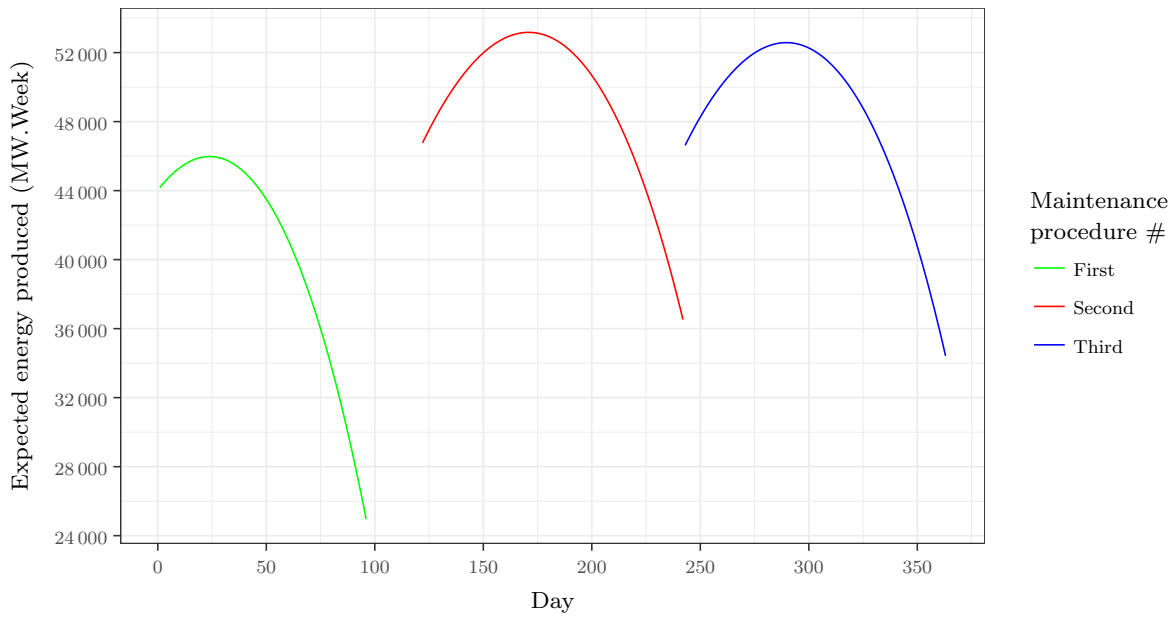


FIGURE 9.6: The expected energy produced, as stated in (9.10), by actual PGU 105 which requires three maintenance procedures over the scheduling window $[0, 365]$.

case study requires a much higher planning resolution than that of the academic benchmark systems described in §6. Finally, various extensions required in the linear and nonlinear GMS objective functions were described and motivated in §9.3 in order to address the problem that the GMS models of §4 do not accommodate the situation where PGUs have to be scheduled for maintenance multiple times within the scheduling window.

CHAPTER 10

Numerical results

Contents

10.1 Parameter optimisation experiments	227
10.2 Approximate solutions	229
10.3 Chapter summary	239

The numerical results obtained for the 157-unit Eskom case study (introduced in §9.2) when adopting minimisation of the probability of unit failure and maximisation of expected energy production as GMS objective functions (presented in §9.3.1 and §9.3.2 respectively), are presented and interpreted in this chapter. These numerical results are based on a prior parameter evaluation for the relevant approximate solution approaches, as described in §5.

10.1 Parameter optimisation experiments

In this section, an experimental design is carried out according to which suitable parameter values for the SA algorithm may be selected for application to the linear and nonlinear models of §4.3. This takes the form of an extensive parameter optimisation experiment in the context of the 157-unit Eskom case study described in §9.2. The SA solution approach described in §5.3 was again implemented in the software package R [175] in combination with RStudio [184] as the *integrated development environment* (IDE) for R.

10.1.1 Minimising the probability of unit failure

This section is devoted to an application of the parameter optimisation experiment described in §5.3.3 within the context of the linear GMS objective of §9.3 for the 157-unit Eskom case study of §9.2. The best algorithmic parameter combination thus uncovered is then employed in the next section to solve the case study instance approximately.

The mean objective function values obtained for the feasible incumbents returned during the first phase of the parameter optimisation experiment involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ are shown in Table A.1. The mean computation times involved in evaluating combinations of these parameter values are furthermore shown in Table A.2, including computation times expended during runs that returned infeasible incumbents. The numbers of times (out of 30) that an infeasible incumbent was returned during an SA search run are finally shown in Table A.3. From these results, it was found that a suitable

parameter value combination for the first phase of the parameter optimisation experiment is $\chi_0 = 0.6$ and $\gamma = 1$. This combination of parameter values was selected based on the fact that it returned zero infeasibilities (out of 30) and among all the combinations returning zero infeasibilities, achieved the smallest mean objective function value. The computation times were not considered as all of the parameter value combinations caused the SA search to time-out (*i.e.* reach a duration of 43 200 seconds). These values for the initial acceptance ratio and the soft constraint violation severity factor were then used to conduct the second phase of the parameter experiment.

The mean objective function values obtained for the feasible incumbents returned during the second phase of the parameter optimisation experiment, involving the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ , are shown in Table A.4. The mean computation times involved in evaluating combinations of these parameter values are furthermore shown in Table A.5, once again including computation times expended during runs that returned infeasible incumbents. The numbers of times (out of 30) that an infeasible incumbent was returned during an SA search run for Phase 2 are finally shown in Table A.6. From these results, it may be observed that only five parameter value combinations did not cause the SA search to reach the time-out of 43 200 seconds. In all of these combinations, the reheating parameter had a value of 0.95. These parameter combinations, however, all returned a large number of infeasible solutions (*i.e.* between four and six out of thirty) and none of these parameter value combinations were therefore selected as the final parameter value combination. The remaining parameter value combinations all caused the SA search to reach the time-out. The final parameter value combination for the second phase of the parameter optimisation experiment was selected as $\alpha = 0.9$, $\xi = 0.75$ and $\psi = 4$. This combination of parameter values was selected based on the fact that it returned only one infeasibility (out of 30) and also achieved a small mean objective function value (only 0.045% worse than the smallest mean obtained). The final set of parameter values adopted in the SA algorithm for solving the 157-unit Eskom case study with minimisation of the probability of unit failure as scheduling criterion is shown in Table 10.1.

TABLE 10.1: *The complete set of parameter values selected for the 157-unit Eskom case study with minimisation of the probability of unit failure as scheduling criterion, as well as the mean objective function value, the mean required computation time and the number of infeasible incumbents associated with these parameter values during the second phase of the parameter optimisation experiment.*

χ_0	γ	α	ξ	ψ	Objective function value	Computation time (s)	Infeasibilities
0.6	1	0.9	0.75	4	321.665	43 200	1 (out of 30)

10.1.2 Maximising expected energy production

This section is devoted to an application of the parameter optimisation experiment described in §5.3.3 within the context of the nonlinear GMS objective of §9.3 for the 157-unit Eskom case study of §9.2. The best algorithmic parameter combination thus uncovered is then employed in the next section to solve the case study GMS instance approximately.

The mean objective function values obtained for the feasible incumbents returned during the first phase of the parameter optimisation experiment involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ are shown in Table A.7. The mean computation times involved in evaluating combinations of these parameter values are furthermore shown in Table A.8, including computation times expended during runs that returned infeasible incumbents. The numbers of times (out of 30) that an infeasible incumbent was returned during an SA search run are finally shown in Table A.9. From these results, it was found that a suitable

parameter value combination for the first phase of the parameter optimisation experiment is $\chi_0 = 0.4$ and $\gamma = 0.5$. This combination of parameter values was selected based on the fact that it returned zero infeasibilities (out of 30) and among all the combinations returning zero infeasibilities, achieved the largest mean objective function value. The computation times were not considered as all of the parameter value combinations caused the SA search to time-out (*i.e.* reach a duration of 43 200 seconds). These values for the initial acceptance ratio and the soft constraint violation severity factor were then used to conduct the second phase of the parameter experiment.

The mean objective function values obtained for the feasible incumbents returned during the second phase of the parameter optimisation experiment, involving the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ , are shown in Table A.10. The mean computation times involved in evaluating combinations of these parameter values are furthermore shown in Table A.11, once again including computation times expended during runs that returned infeasible incumbents. The numbers of times (out of 30) that an infeasible incumbent was returned during an SA search run for Phase 2 are finally shown in Table A.12. From these results, it may be observed that only three parameter value combinations did not cause the SA search to reach the time-out of 43 200 seconds. In all of these combinations, the reheating parameter had a value of 0.95. These parameter combinations, however, all returned a number of infeasible solutions (*i.e.* between two and four out of thirty) and therefore, none of these parameter value combinations were selected as the final parameter value combination. The remaining parameter value combinations all caused the SA search to reach the time-out. The final parameter value combination for the second phase of the parameter optimisation experiment was selected as $\alpha = 0.85$, $\xi = 0.75$ and $\psi = 2$. This combination of parameter values was selected based on the fact that it returned zero infeasible solutions (out of 30) and also achieved a small mean objective function value (only 0.0023% worse than the smallest mean obtained). The final set of parameter values adopted in the SA algorithm for solving the 157-unit Eskom case study with maximisation of the expected energy production as scheduling criterion is shown in Table 10.2.

TABLE 10.2: *The complete set of parameter values selected for the 157-unit Eskom case study with maximisation of the expected energy production as scheduling criterion, as well as the mean objective function value, the mean required computation time and the number of infeasible incumbents associated with these parameter values during the second phase of the parameter optimisation experiment.*

χ_0	γ	α	ξ	ψ	Objective function value	Computation time (s)	Infeasibilities
0.4	0.5	0.85	0.75	2	19 481 895	43 200	0 (out of 30)

10.2 Approximate solutions

In this section, the best algorithmic parameter combinations uncovered in §10.1 are employed to obtain approximate solutions to the linear and nonlinear GMS models of §4.3 (extended as described in §9.3.1 and §9.3.2) by means of the SA algorithm described in §5 in the context of the 157-unit Eskom case study of §9.2.

10.2.1 Minimising the probability of unit failure

An approximate solution to the linear GMS model of §4.3 (extended as described in §9.3.1) was obtained for the 157-unit Eskom case study by the SA algorithm with the parameter combi-

nations as specified in Table 10.1. The decision variable values of the incumbent are given in integer decision vector form in Table 10.3, and corresponds to an objective function value of 321.524.

TABLE 10.3: *The decision variable values of the incumbent returned by the SA algorithm with parameter values as specified in Table 10.1 for the 157-unit Eskom case study with minimisation of the probability of unit failure as scheduling criterion. Asterisks denote PGUs that are not scheduled for maintenance in the current scheduling window.*

PGU in \mathcal{U}	k_u	PGU in \mathcal{U}	k_u	PGU in \mathcal{U}	k_u	PGU in \mathcal{U}	k_u	PGU in \mathcal{U}	k_u
1	269	33	282	65	185	97	*	129	3
2	*	34	23	66	1	98	*	130	208
3	17	35	198	67	182	99	*	131	30
4	183	36	227	68	2	100	*	132	216
5	71	37	*	69	191	101	*	133	9
6	324	38	*	70	72	102	*	134	208
7	46	39	7	71	2	103	*	135	25
8	86	40	85	72	182	104	*	136	189
9	207	41	271	73	*	105	*	137	23
10	47	42	4	74	*	106	27	138	146
11	*	43	235	75	51	107	106	139	242
12	102	44	14	76	255	108	195	140	90
13	30	45	323	77	*	109	285	141	11
14	2	46	*	78	*	110	15	142	96
15	196	47	*	79	13	111	94	143	183
16	12	48	*	80	113	112	192	144	293
17	256	49	332	81	16	113	275	145	10
18	*	50	3	82	194	114	2	146	182
19	76	51	295	83	13	115	91	147	10
20	234	52	324	84	184	116	195	148	182
21	*	53	5	85	8	117	277	149	28
22	4	54	87	86	199	118	18	150	122
23	239	55	7	87	2	119	94	151	285
24	*	56	1	88	186	120	193	152	25
25	11	57	258	89	8	121	272	153	124
26	*	58	99	90	255	122	20	154	246
27	97	59	65	91	5	123	102	155	24
28	263	60	331	92	187	124	189	156	124
29	*	61	353	93	93	125	274	157	249
30	116	62	2	94	231	126	17		
31	24	63	259	95	*	127	123		
32	186	64	6	96	333	128	247		

A graphical representation of this maintenance schedule is presented in Figures 10.1 and 10.2 with the colour scale indicating the rated capacity (in MW) and the failure rate of each PGU, respectively. The available system capacity associated with this solution over the duration of the scheduling window is shown in Figure 10.3.

Analysing the approximate GMS solution in Table 10.3 for the 157-unit Eskom case study, it is observed in Figure 10.1 that some maintenance preference is given to PGUs with large rated capacities, as expected. PGUs that contribute large capacities to the overall capacity

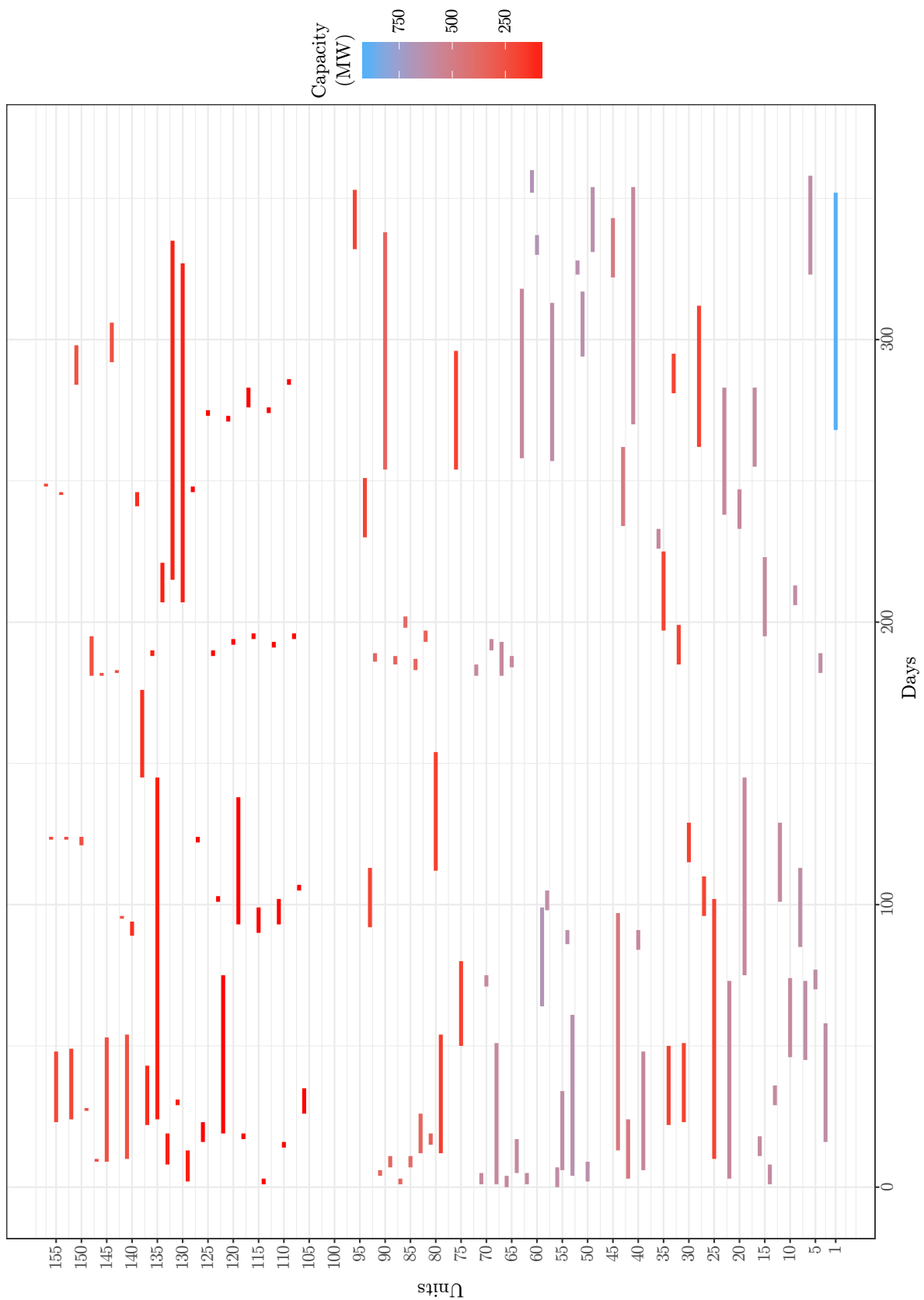


FIGURE 10.1: The incumbent returned by the SA algorithm (as presented in Table 10.3) for the linear model of §4.3 (extended as described in §9.3.1), in the context of the 157-unit Eskom case study with minimisation of the probability of unit failure as scheduling criterion and for parameter values as indicated in Table 10.1, with the colour scale indicating the rated capacity of the PGUs.



FIGURE 10.2: The incumbent returned by the SA algorithm (as presented in Table 10.3) for the linear model of §4.3 (extended as described in §9.3.1), in the context of the 157-unit Eskom case study with minimisation of the probability of unit failure as scheduling criterion and for parameter values as indicated in Table 10.1, with the colour scale indicating the failure rate of the PGUs.

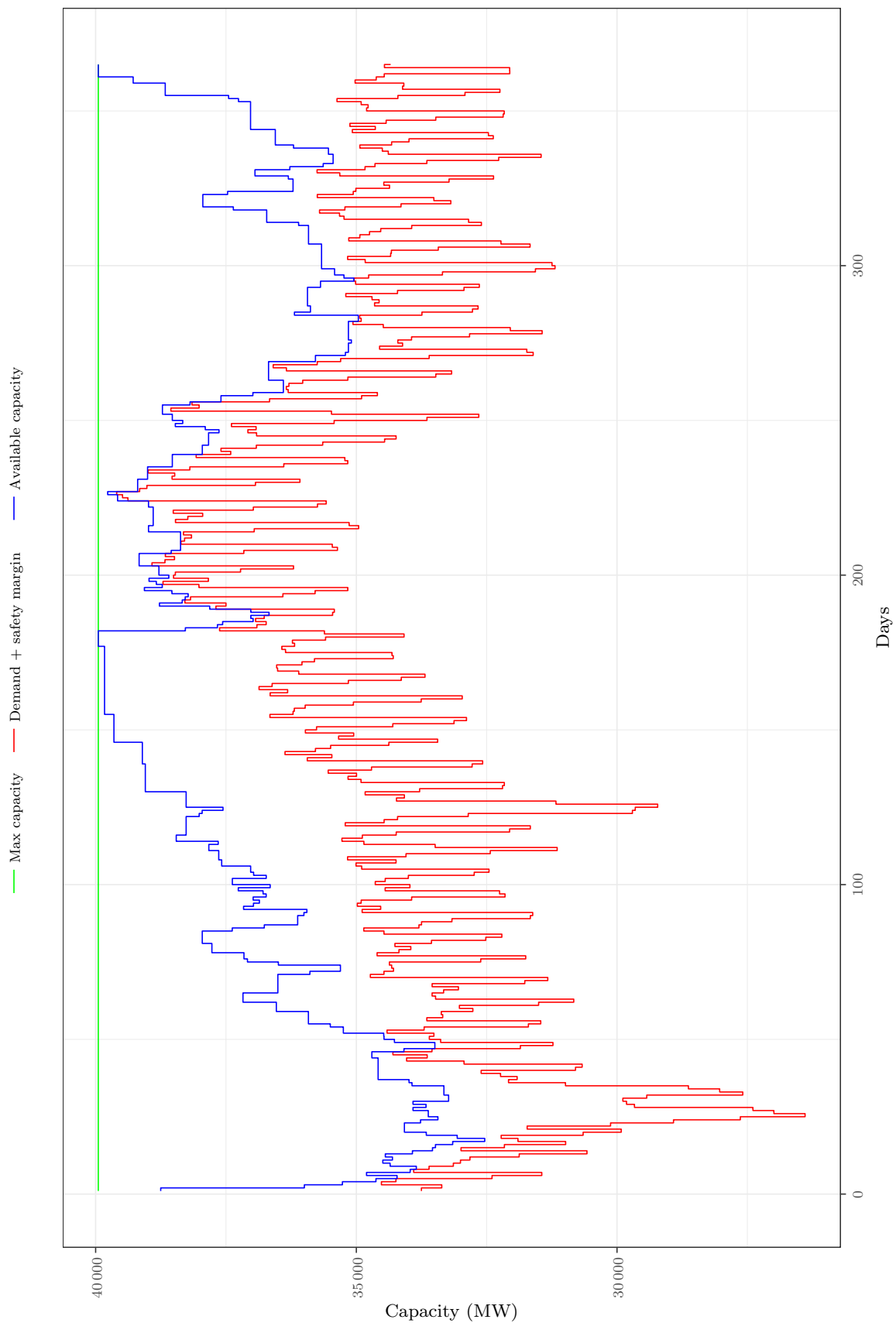


FIGURE 10.3: The system capacity available over the duration of the scheduling window for the 157-unit Eskom case study maintenance schedule in Figures 10.1 and 10.2 with minimisation of the probability of unit failure as scheduling criterion.

of the system are typically scheduled for maintenance early in the scheduling window or early within the PGUs' earliest and latest starting maintenance times. This is observed for PGU 1, for example, which has the largest rated capacity (900 MW) in the system, and for which maintenance is scheduled to start during time period 269. One of the reasons why PGU 1 is scheduled for maintenance so late in the scheduling window is due to its earliest and latest maintenance starting times, which are 225 and 282, respectively. Another aspect that contributes to PGU 1 being scheduled for maintenance so late in the scheduling window is the fact that the energy demand during the middle to late stages of the scheduling window is very high, and so scheduling this PGU for maintenance may cause the system not to be able to satisfy the demand. Other PGUs with large rated capacities (such as PGUs 3, 10, 14, 16 and 22) are all scheduled for maintenance early during the scheduling window.

It is also observed in Figure 10.2, that maintenance preference is given to PGUs with large failure rates. The subset of PGUs which utilise gas as fuel source (*i.e.* gas turbines) typically have high failure rates, as specified for PGU 106 to PGU 128 in Table 9.2. In Figure 10.2, these PGUs are typically scheduled early during the scheduling window or within the PGUs' earliest and latest starting maintenance times.

A similar phenomenon is observed in Figure 10.3 as was observed in the context of the two benchmark systems in §7. During the early stages of the scheduling window, the difference between the available capacity and the demand together with the safety margin is very small as many PGUs with large capacities are scheduled for maintenance then. The available capacity then steadily increases until the middle of the scheduling window where again the difference between the available capacity and the demand together with the safety margin is very small. The reason for this is that many of the PGUs have earliest and latest maintenance starting times limiting their maintenance to the second half of the scheduling window.

10.2.2 Maximising expected energy production

An approximate solution to the linear GMS model of §4.3 (extended as described in §9.3.2) was obtained for the 157-unit Eskom case study by the SA algorithm with the parameter combinations as specified in Table 10.2. The decision variable values of the incumbent are given in integer decision vector form in Table 10.4, and corresponds to an objective function value of 19 507 890 MW·week (468 189 355 MW·h).

A graphical representation of this maintenance schedule is presented in Figures 10.4 and 10.5 with the colour scale indicating the rated capacity (in MW) and the failure rate of each PGU, respectively. The available system capacity associated with this solution over the duration of the scheduling window is shown in Figure 10.6.

Analysing the approximate GMS solution in Table 10.4 for the 157-unit Eskom case study, it is observed in Figure 10.4 that PGUs with large rated capacities are scheduled closer to the end of the scheduling window, as expected. PGUs that contribute large capacities to the overall capacity of the system are typically scheduled for maintenance later in the scheduling window or late within the PGUs' earliest and latest starting maintenance times. This is observed for PGU 1, for example, which has the largest rated capacity (900 MW) in the system, and for which maintenance is scheduled to start during time period 234. One of the reasons why PGU 1 is scheduled for maintenance so late in the scheduling window is due to the fact that the peaks of the energy production curves are typically located towards the end of the scheduling window. Another aspect that contributes to PGU 1 being scheduled for maintenance so late in the scheduling window is its earliest and latest maintenance starting times, which are 225 and

TABLE 10.4: The decision variable values of the incumbent returned by the SA algorithm with parameter values as specified in Table 10.2 for the 157-unit Eskom case study with maximisation of the expected energy production as scheduling criterion. Asterisks denote PGUs that are not scheduled for maintenance in the current scheduling window.

PGU in \mathcal{U}_A	$k_{u,i}$	PGU in \mathcal{U}_A	$k_{u,i}$	PGU in \mathcal{U}_A	$k_{u,i}$	PGU in \mathcal{U}_A	$k_{u,i}$
1	234	28	47,346	55	96,270	82	*
2	*	29	282	56	70,361	83	*
3	38,354	30	57,338	57	94,318	84	*
4	92	31	238	58	29,360	85	*
5	323	32	*	59	85	86	*
6	84	33	*	60	80,359	87	*
7	127	34	49	61	*	88	45,112,217,358
8	209	35	106	62	*	89	39,116,229,327
9	84	36	256	63	92	90	42,125,222,333
10	*	37	71,335	64	254	91	20,101,209,363
11	139	38	78	65	*	92	2,128,205,342
12	50	39	323	66	*	93	48,186,360
13	76,289	40	*	67	49	94	49,246
14	61,338	41	*	68	132	95	98,244
15	*	42	*	69	60,357	96	64,300
16	147	43	330	70	86,362	97	5,331
17	239	44	21	71	90,351	98	25,145,347
18	*	45	295	72	88,319	99	100
19	78	46	324	73	98,267	100	9,124,204,351
20	256	47	71	74	95,355	101	29,362
21	*	48	104	75	113	102	87,352
22	78	49	43	76	215	103	28,148,332
23	*	50	62,296	77	*	104	41,175,360
24	120	51	119	78	309	105	13,143,364
25	228	52	112	79	*		
26	*	53	330	80	*		
27	139	54	351	81	*		

282, respectively. Other PGUs with large rated capacities (such as PGUs 4, 6, 15, 17 and 23) are all scheduled for maintenance late during the scheduling window.

It is also observed in Figure 10.5 that PGUs with large failure rates are either scheduled early or late during the scheduling window. The subset of PGUs which utilise gas as fuel source (*i.e.* gas turbines) typically exhibit high failure rates, as seen for PGU 106 to PGU 128 in Table 9.2. In Figure 10.5, these PGUs are typically scheduled either early or late within the PGUs' earliest and latest starting maintenance times as these PGUs are not expected to contribute large amounts of energy to the overall system.

A similar phenomenon is observed in Figure 10.6 as was observed for the two benchmark systems in §7. During the early stages of the scheduling window, the difference between the available capacity and the demand together with the safety margin is very large as not many PGUs are scheduled for maintenance then. The available capacity then steadily decreases towards the middle of the scheduling window where the difference between the available capacity and the demand together with the safety margin is very small due to PGUs with large capacities being scheduled towards the end of the allowable scheduling window. The difference between the available capacity and the demand together with the safety margin remains small as the demand

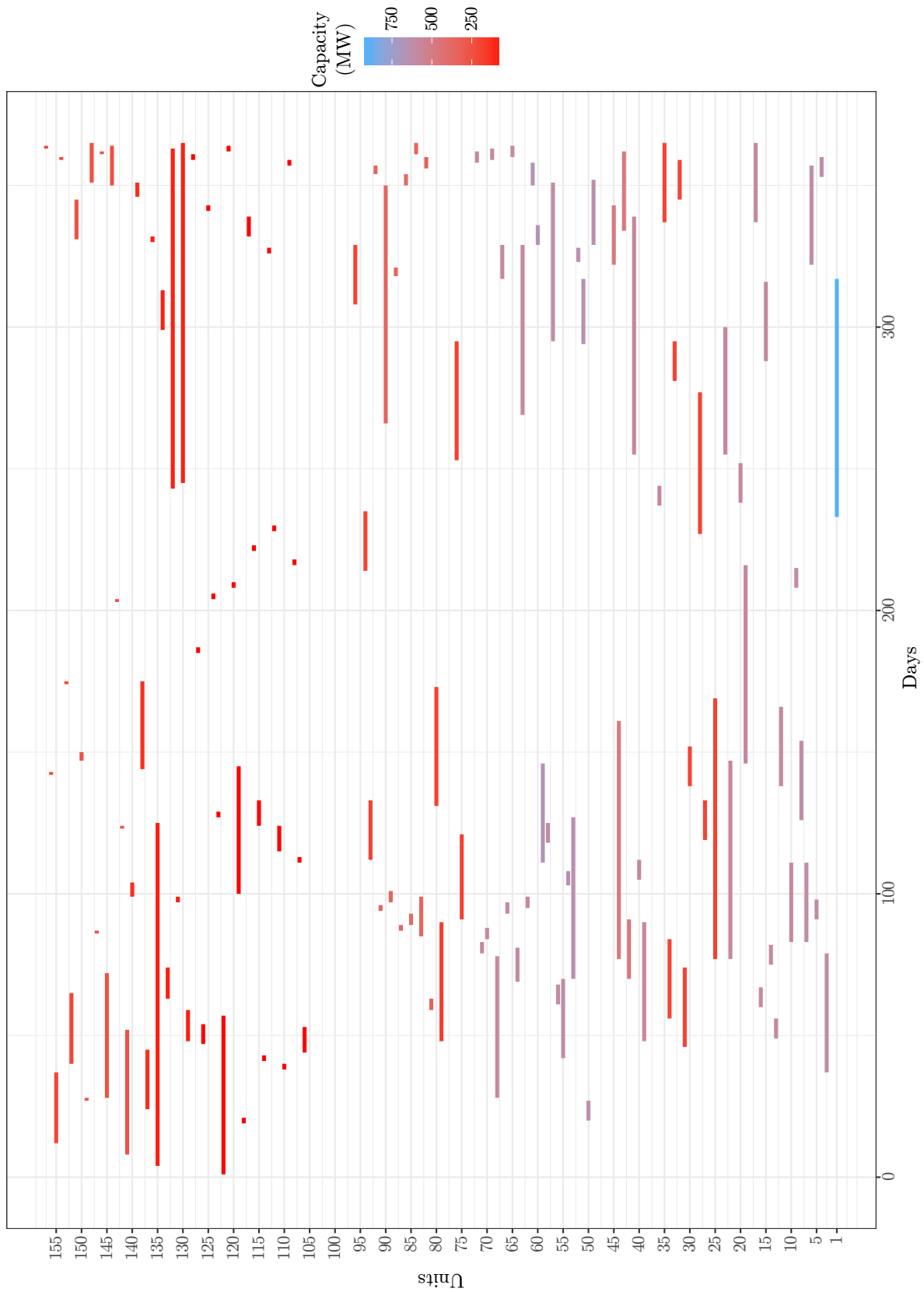


FIGURE 10.4: The incumbent returned by the SA algorithm (as presented in Table 10.4) for the nonlinear model of §4.3 (extended as described in §9.3.2), in the context of the 157-unit Eskom case study with maximisation of the expected energy production as scheduling criterion and for parameter values as indicated in Table 10.2, with the colour scale indicating the rated capacity of the PGUs.

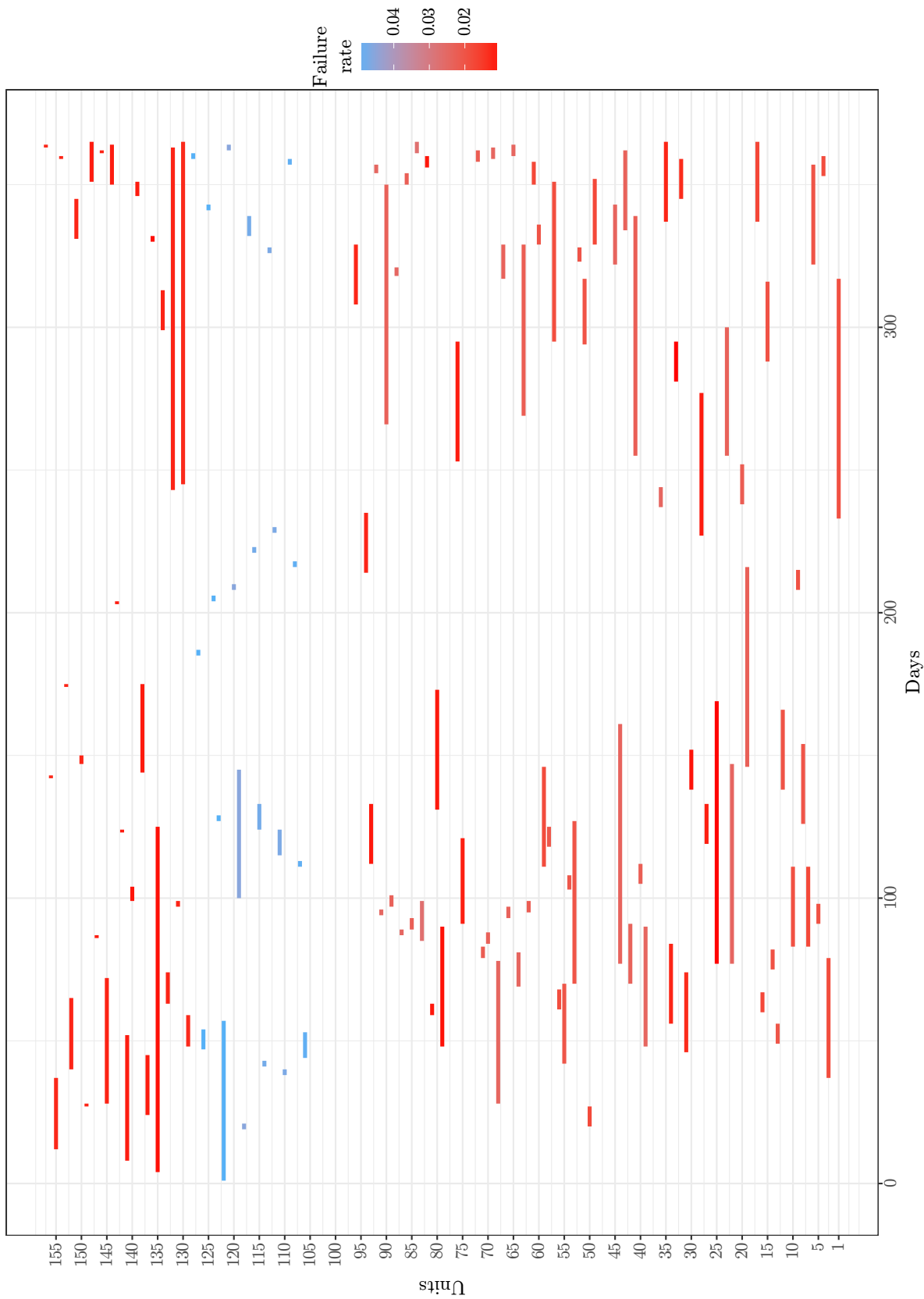


FIGURE 10.5: The incumbent returned by the SA algorithm (as presented in Table 10.4) for the nonlinear model of §4.3 (extended as described in §9.3.2), in the context of the 157-unit Eskom case study with maximisation of the expected energy production as scheduling criterion and for parameter values as indicated in Table 10.1, with the colour scale indicating the failure rate of the PGUs.

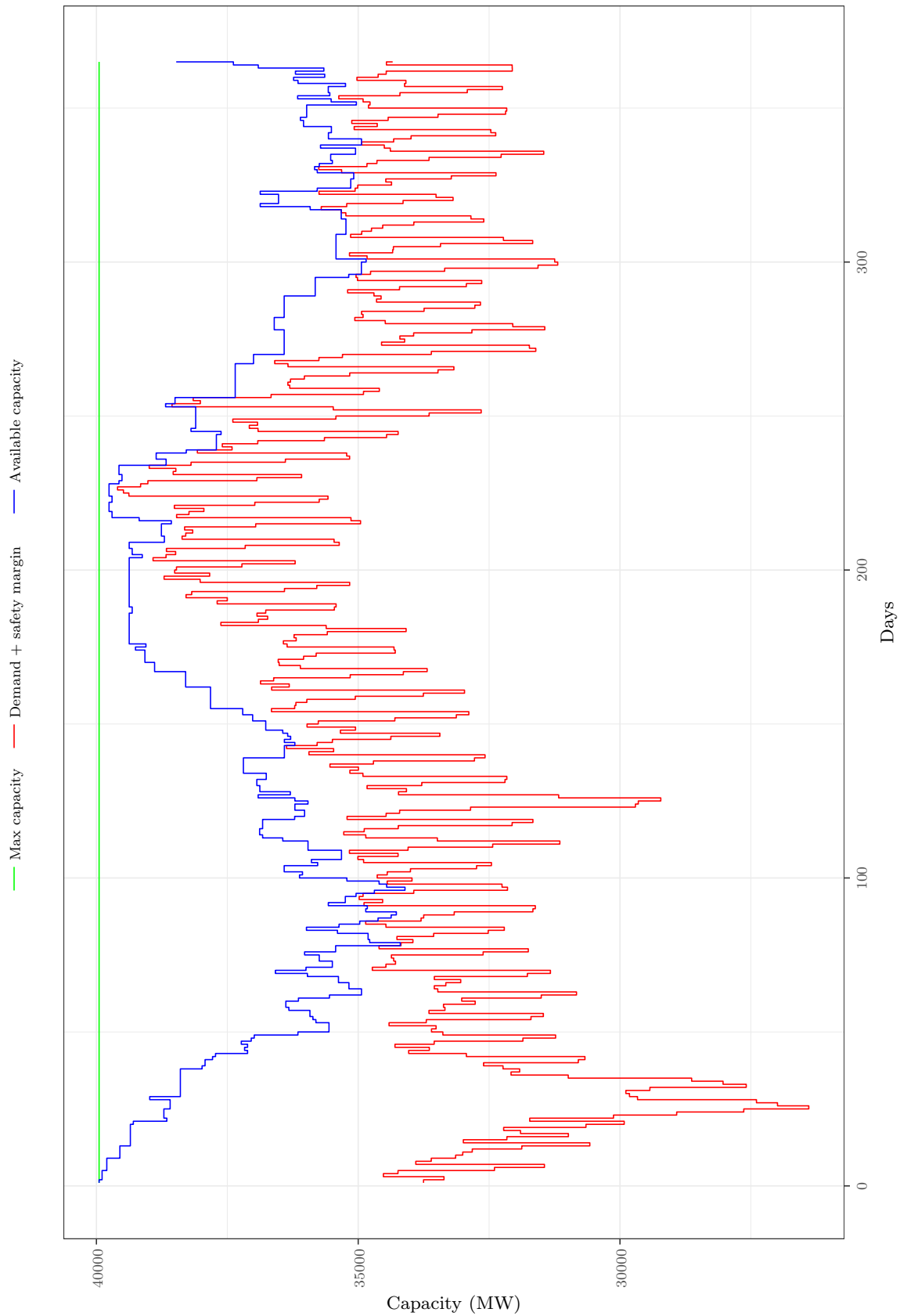


FIGURE 10.6: The system capacity available over the duration of the scheduling window for the 157-unit Eskom case study maintenance schedule in Figures 10.4 and 10.5 with maximisation of the expected energy production as scheduling criterion.

and safety margin reaches a peak during the middle of the scheduling window and thereafter (when the demand decreases again) many PGUs with large rated capacities are scheduled for maintenance as the peaks of the expected energy production curves are located here.

10.3 Chapter summary

The numerical results obtained by employing the method of SA were presented in this chapter for a real-world case study, the 157-unit Eskom case study (described in §9), with regards to both the linear and nonlinear models presented in §9.3. The results obtained for parameter optimisation experiments conducted in the context of both the linear and nonlinear models, were presented in §10.1. For the minimisation of the probability of unit failure objective function, a suitable parameter combination was reported in Table 10.1. For the maximisation of the expected energy production objective function, a suitable parameter combination was similarly reported in Table 10.2. Descriptions of the incumbents returned by the SA algorithm for the linear and nonlinear models, employing the optimal parameter combinations described in §10.1, were presented in §10.2.1 and §10.2.2, respectively. It was found that for the minimisation of unit failure objective function, the incumbent solution had an associated objective function value of 321.524. In respect of the maximisation of expected energy production objective function, the incumbent solution had an associated objective function value of 19 507 890 MW·week (468 189 355 MW·h).

CHAPTER 11

Decision support framework

Contents

11.1 General considerations in decision support systems	241
11.2 Detailed process description	242
11.3 System development	243
11.4 Chapter summary	258

The models proposed in §4 of this dissertation were implemented within a newly proposed computerised decision support tool aimed at facilitating GMS decisions. The design of this tool is described in detail in this chapter. Some general considerations with regards to general *decision support system* (DSS) development are briefly discussed, after which a detailed process description of the decision support tool is provided, elucidating the working of the system in a step-by-step walk-through fashion.

11.1 General considerations in decision support systems

A DSS is a computerised information system that may be employed to support employees of companies with business and organisational decision making activities [31]. A properly designed DSS does not necessarily specify which decision to make in respect of business or organisational activities, but rather aids the user in making an informative decision. Such a DSS typically compiles useful information, which is presented to the user, by analysing raw data, documents and knowledge elicited from industry experts in order to identify or solve complex problems [194]. DSSs provide many benefits which include improved efficiency when solving problems in the face of rapidly changing variables or system inputs.

According to Stair and Reynolds [194], a DSS typically consists of three main components namely, a *database*, a *graphical user interface* and a *model base*. These three components are described in this section in order to provide some insight into the design of a novel, generic and user-friendly DSS aimed at improved GMS that is put forward in this chapter.

11.1.1 The database

The database is one of the most crucial components of a DSS [195] and allows data to be stored in a structured manner [123]. An accompanying database management system also allows for

the data to be accessible to the other relevant components of the DSS [194]. An efficient database management system provides four main benefits to the design of a DSS [216]. The first benefit is that the database management system ensures that there are no redundant data in the database. Duplicates and faulty entries in the database are avoided by a properly designed database management system. The second benefit, according to Watt and Eng [216], is that the data are accessible by multiple users at the same time, which may improve the efficiency with which tasks are completed. Thirdly, the design of such a database management system ensures the integrity of the data in the database. All the entries in the database are guaranteed to be in the same format, ensuring that the data cannot be formatted incorrectly by accident. Finally, the security of a database is easy to manage if an effective database management system is in place. Restrictions may be imposed on who is allowed to access the database and what transformations or operations are permitted in respect of the data within the database [195].

11.1.2 The graphical user interface

In order to facilitate effective *human-computer interaction* (HCI), a *graphical user interface* (GUI) is required. The GUI provides a user-friendly link between a computerised DSS and a human operator. This increases the ease with which operators are able to interact with the computer and gain access to results returned by complex computerised systems. In short, the GUI allows the operator to provide the DSS with the required input for the problem at hand and then obtain the relevant output, which may be presented effectively in an intuitive and understandable manner to the operator [196]. An operator should be able to engage easily with a DSS by means of the GUI, which should provide the operator with the ability to solve the required problem effectively and efficiently.

An important aspect in the design of a GUI is the manner in which information is presented to the operator. Although it is important to present all the relevant information to the operator, this has to be done in such a way as to not overwhelm the operator. On the other hand, the presentation of information should also not be so sparse and unintelligible that it discourages operators from utilising the DSS [151]. It is therefore important to find a good balance between information overload and a vague communication of information between machine and man [195]. An aspect that may aid the analyst in obtaining this balance is a thorough understanding of the capabilities and competencies of the operators who are anticipated to use the system.

11.1.3 The model base

The model base is the workhorse of the DSS and is the component that provides the operator with access to one or multiple models which aid him during the decision making process [194]. These models may be embedded in the DSS in many different forms, such as mathematical algorithms, techniques or methods. The embedded models may even consist of a combination of these forms which raises the problem of how to combine the outputs returned by these models as intelligible decision support to the user [150].

11.2 Detailed process description

A high-level overview of the interaction between the three DSS components described in §11.1 is illustrated graphically in Figure 11.1 within the context of a GMS DSS.

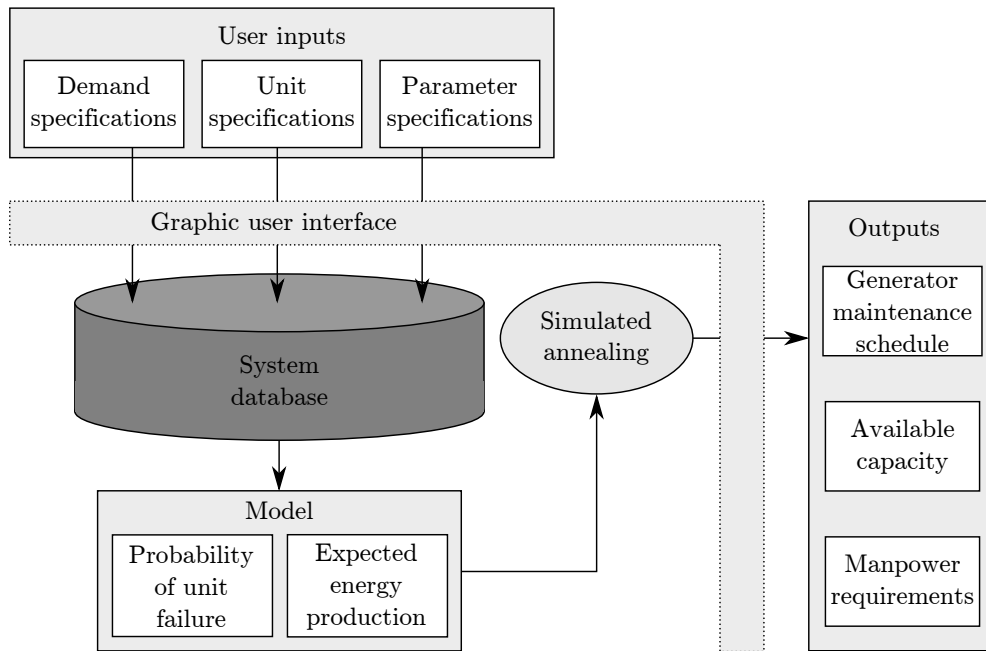


FIGURE 11.1: A high-level graphical overview of the DSS design proposed in this dissertation.

The first activity within the DSS is concerned with the user inputs required. The user specifies information related to the demand of some power system, the PGUs contained in the specified power system, and the parameters of an SA algorithm employed in the models of the DSS by means of a user-friendly GUI. Secondly, these user-specified inputs are stored in the system database of the DSS. The model base, which either aims to minimise the probability of unit failure or maximise the expected energy production (as described in §4) of the system under consideration, then requests these inputs from the system database. The model base next employs the method of SA to solve a user-specified GMS problem instance. The result obtained by the SA algorithm is then returned to the user by means of the GUI. This result includes the generator maintenance schedule corresponding to the incumbent solution obtained, the available system capacity associated with the incumbent solution and the corresponding manpower requirements over the duration of the scheduling window.

11.3 System development

The software environment within which the DSS proposed in §11.2 was developed by the author is a package supported by RStudio [184], called **Shiny** [41]. **Shiny** is a web framework used to construct elegant and powerful web applications displaying interactive reports and data visualisations based in R. The package enables users without web development backgrounds to develop web applications, but remains sufficiently sophisticated for the development of usable and relevant applications. The package **Shiny** was adopted in the development of the DSS proposed in this dissertation due to its ability to create elegant GUIs capable of changing dynamically, based on R script files.

11.3.1 Data preparation

In order to standardise the procedures of the DSS, the required input data have to be prepared in a specific format before the DSS can be utilised. The DSS requires two user-specified input

files. This first contains the demand specifications of the power system and the second its PGU specifications. Both of these files have to be prepared in a *comma separated values* (CSV) format. An example of the exact input format required for the demand specifications file is presented in Figure 11.2, while an example of the format for PGU specification is presented in Figure 11.3.

	A	B
1	Time Period	Demand
2	1	2457
3	2	2565
4	3	2502
5	4	2377
6	5	2508
7	6	2397
8	7	2371

FIGURE 11.2: Required input format of the demand specification for a power system.

A portion of the demand specifications data file for the 32-unit IEEE-RTS benchmark system introduced in §6.2 is shown in Figure 11.2. This file contains the demand of the power system for all fifty two time periods of the scheduling window. Note that this demand specification does not include the safety margin of the power system. The PGU specifications data file for the same benchmark system is similarly shown in Figure 11.3. This file contains the PGU number, capacity, actual capacity (specified as zero for duplicate PGUs, as described in §9), maintenance duration, earliest and latest starting times (*e.g.* maintenance window), failure rate, previous maintenance scheduling date, number of required maintenance procedures, allowable earliest and latest maintenance starting times (as described in §9.3 and calculated by means of (9.1)) for each PGU, as well as manpower required for each week of maintenance.

11.3.2 System walk-thorough

After having prepared the required input data in the specified format, as described in the previous section, the DSS can be utilised to recommend good generator maintenance schedules for the power system specified. Once the DSS is initialised, the user is presented with the “Home screen” shown in Figure 11.4. On this screen, a short introduction to the DSS is provided to the user, as well as seven steps to be followed in order to utilise the DSS to its full potential. The user can navigate between the seven steps by selecting the steps from the dropdown list on this page. When selecting a step from the dropdown list, the instructions for the selected step appear below the dropdown list.


After the instructions have been read and understood, the user can navigate to the “System specifications” window on the left-hand side of the screen, which displays the window seen in Figure 11.5. The user may, however, navigate back to the “Instructions” window at any subsequent time if some of the instructions have to be reviewed. In the “System specifications” window, the user can input the demand specifications and PGU specifications in the formats specified above. The user can also select the required safety margin for the power system under consideration by moving the slider between zero and a maximum of thirty percent. The final input is whether or not a manpower requirement is to be considered for the specified system. This can be selected by clicking on the “Include manpower requirements” checkbox and specifying the maximum number of manpower available in the textbox provided. Once the user is satisfied with the inputs, he can click on the “Accept” button which will upload the specifications to the DSS database.

A	B	C	D	E	F	G	H
1	Capacity	Actual capacity	Duration	Earliest starting date	Latest starting date	Failure rate	Previous maintenance date
2	20	20	2	1	25	0.3733333	6
3	20	20	2	1	25	0.3733333	25
4	76	76	3	1	24	0.0857143	1
5	76	76	3	27	50	0.0857143	44
6	20	20	2	1	25	0.3733333	3
7	20	20	2	27	51	0.3733333	31
8	76	76	3	1	24	0.0857143	22

	I	J	K	L	M	N
1	Number of maintenance procedures	New earliest	New latest	Manpower		
2	1	0	52	7	7	0
3	1	0	52	7	7	0
4	1	0	52	12	10	10
5	1	0	52	12	10	10
6	1	0	52	7	7	0
7	1	0	52	7	7	0
8	1	0	52	12	10	10

FIGURE 11.3: Required format of the PGU specifications for a power system.

Decision support



- Instructions
- System specifications
- Algorithm specifications

Instructions for using the decision support system

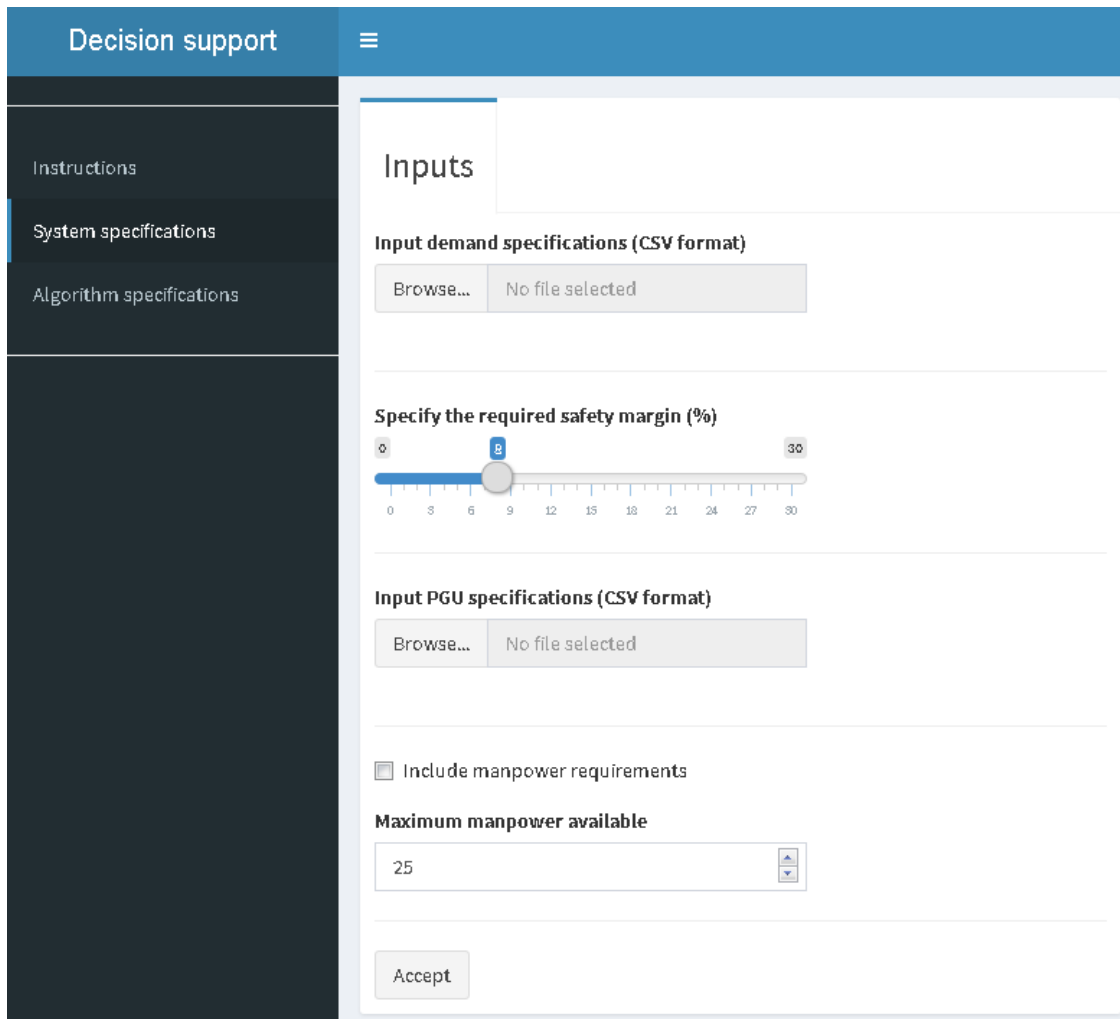
The purpose of this decision support system is to aid decision makers which are tasked with finding good power generating unit maintenance schedules for a specified power system. The system is capable of providing good maintenance scheduling by employing an approximate solution approach based on the power system provided by the decision maker.

In order to utilise the decision support system to its full potential, a number of steps are specified which has to be followed. The user of this decision support system may return to this page during any time of operation. Please follow the following steps carefully:

Step 1

Navigate to the 'Systems specifications' page on the left of the window. On this page, the user can input the demand specifications, including the required safety margin, and the PGU specifications of the system. Both of these inputs should be in 'comma-separated values' (.csv) format in the table form specified. The demand specifications can be selected by click on the 'Browse...' button and navigating to the file containing the demand specifications. The required safety margin (as a percentage of the specified demand) can be specified by moving the slider. The PGU specifications can be selected by click on the 'Browse...' button and navigating to the file containing the PGU specifications.

FIGURE 11.4: The “Home screen” presented to the user when the DSS is initialised.



The screenshot displays a web application interface titled "Decision support". On the left, a dark sidebar contains navigation links: "Instructions", "System specifications" (which is highlighted), and "Algorithm specifications". The main content area is titled "Inputs" and contains several sections:

- Input demand specifications (CSV format):** A "Browse..." button and a "No file selected" status.
- Specify the required safety margin (%):** A horizontal slider ranging from 0 to 30. The slider is currently set to 8, with a blue marker and the number "8" displayed above it.
- Input PGU specifications (CSV format):** A "Browse..." button and a "No file selected" status.
- Include manpower requirements:** An unchecked checkbox.
- Maximum manpower available:** A numeric input field containing the value "25".
- Accept:** A button at the bottom of the form.

FIGURE 11.5: GUI through which the user can upload the demand and PGU specifications of the power system under consideration.

Once the specifications have been uploaded successfully to the database, an overview of the system can be seen in the “Summary of system” tab, as shown in Figure 11.6. In this tab, the number of PGUs specified in the system is displayed, as well as the number of time periods for which demand has been specified. A graphical representation of the system demand is also shown in this tab, as illustrated in Figure 11.6. This graph displays the forecast demand, the demand together with the specified safety margin and the maximum system capacity. The demand specification graph is fully interactive, allowing the user to hover with the computer mouse over a graph, upon which information about the specific graph is provided. Clicking on the legend for a specific graph (located in the top right-hand corner of the graph), for example, will cause the specific graph to become hidden. Furthermore, by moving the safety margin slider shown in Figure 11.5, the “Demand + safety margin” graph will automatically update on the plot, as shown in Figure 11.7 where the slider has been moved from an eight percent safety margin to a fifteen percent safety margin.

Navigating to the “PGU specifications tab,” the user can review the specifications of the PGUs in the power system under consideration, as shown in Figure 11.8. The PGU specifications table in this tab is also interactive. The user can, for example, sort the data by a selected column as well as search the table for certain data. Similarly, navigating to the “Demand specifications” tab, the user can review the demand specifications of the system, as shown in Figure 11.9. This

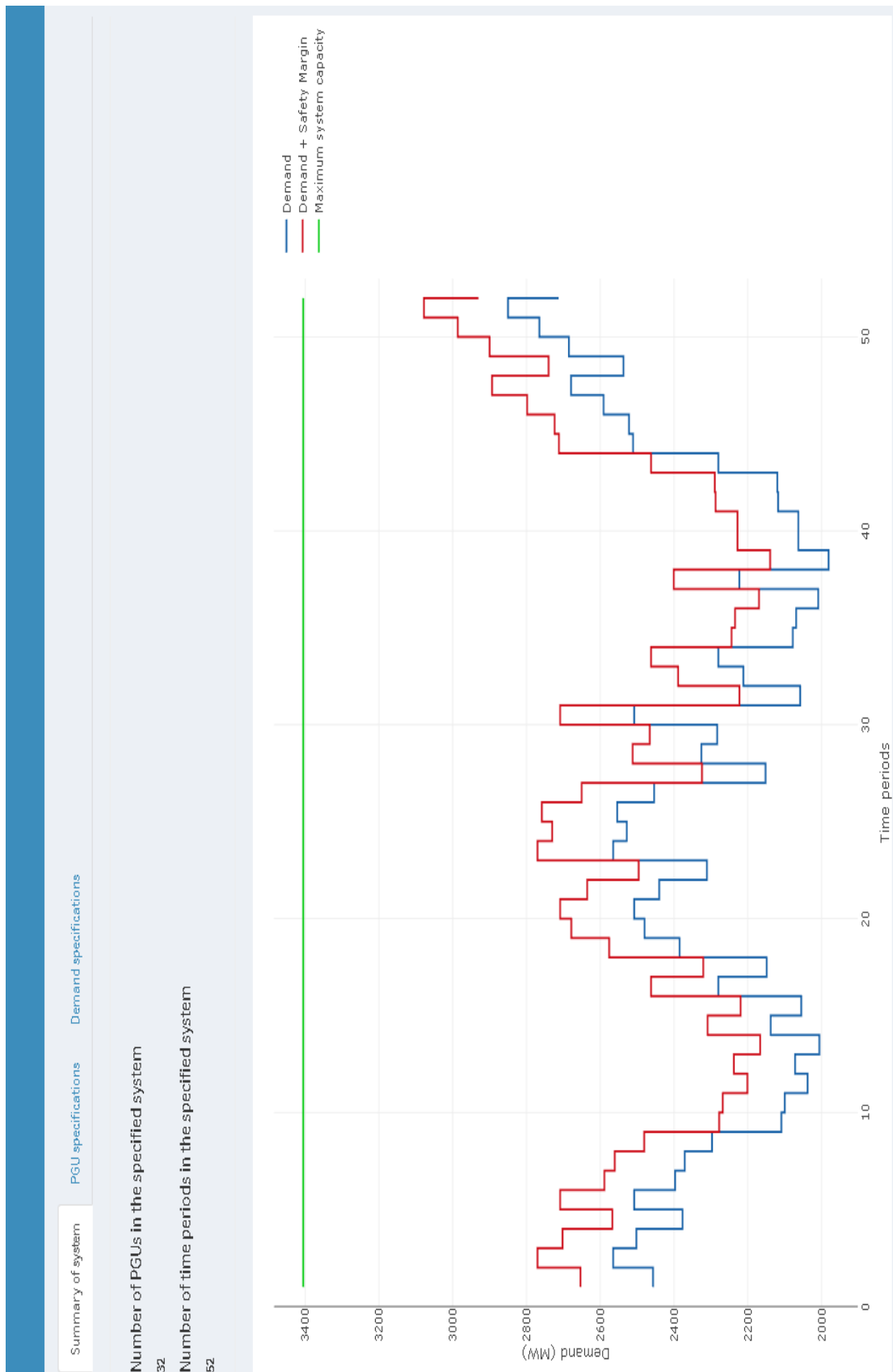


FIGURE 11.6: An overview of the specified system in the “Summary of system” tab, displaying the number of PGUs in the system, the number of scheduling time periods as well as a graphical representation of the system demand over the scheduling window together with an eight percent safety margin.

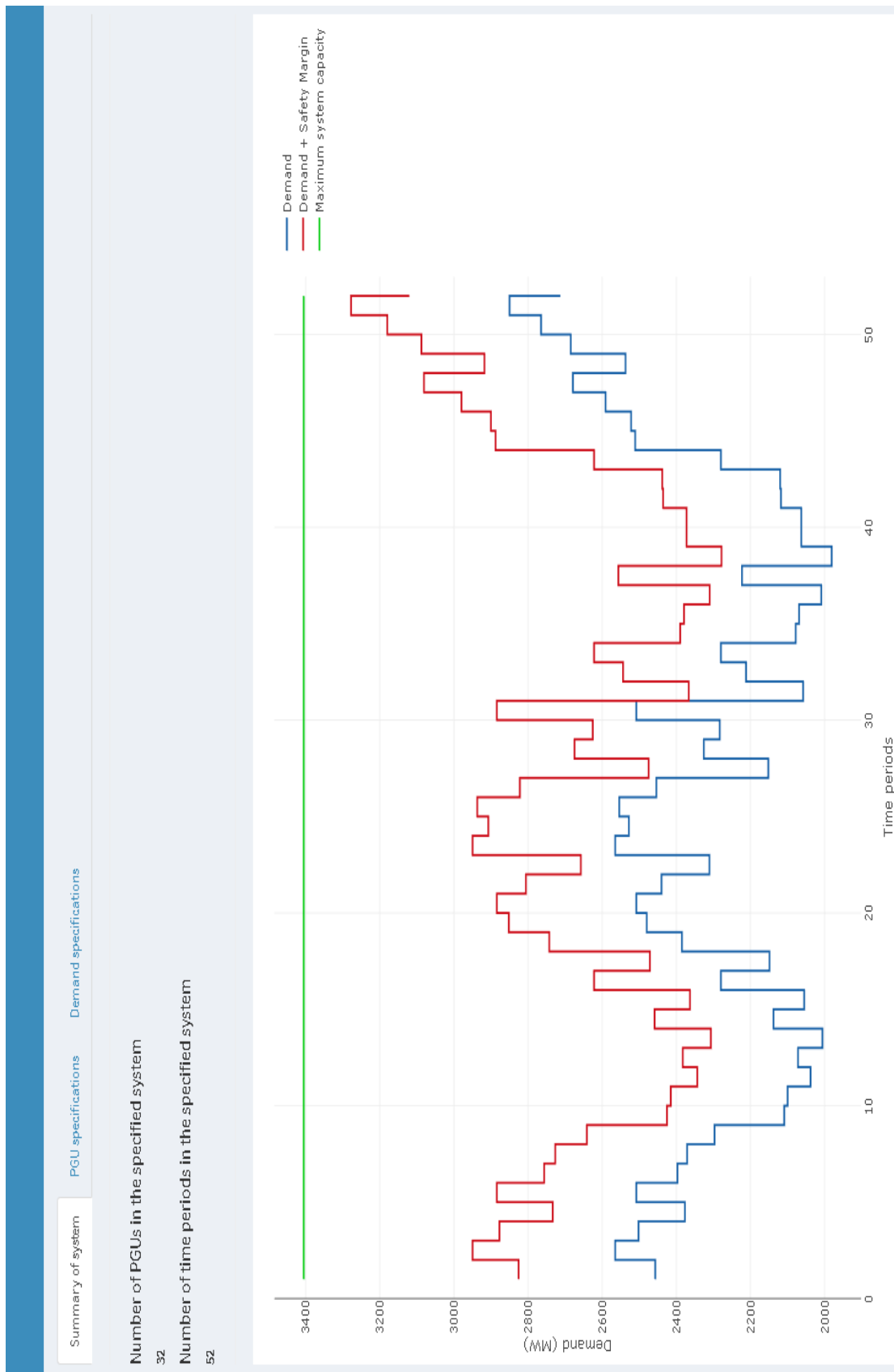


FIGURE 11.7: An overview of the system specified in the “Summary of system” tab, displaying the number of PGUs in the system, the number of scheduling time periods as well as a graphical representation of the system demand over the scheduling window together with a fifteen percent safety margin.

Summary of system **PGU specifications** Demand specifications

Show 10 entries Search:

Unit.Number	Capacity	Actual.capacity	Duration	Earliest.starting.date	Latest.starting.date	Failure.rate	Previous.maintenance.date	Ni
1	20	20	2	1	25	0.3733333333		6
2	20	20	2	1	25	0.3733333333		25
3	76	76	3	1	24	0.085714286		1
4	76	76	3	27	50	0.085714286		44
5	20	20	2	1	25	0.3733333333		3
6	20	20	2	27	51	0.3733333333		31
7	76	76	3	1	24	0.085714286		22
8	76	76	3	27	50	0.085714286		36
9	100	100	3	1	50	0.14		41
10	100	100	3	1	50	0.14		27

Showing 1 to 10 of 32 entries

Previous 1 2 3 4 Next

FIGURE 11.8: The “PGU specifications” tab, summarising the specifications of the PGUs in tabular format.

Time.Period	Demand	Demand + safety margin
1	2457	2653.56
2	2565	2770.2
3	2502	2702.16
4	2377	2567.16
5	2508	2708.64
6	2397	2588.76
7	2371	2560.68
8	2297	2480.76
9	2109	2277.72
10	2100	2268

Showing 1 to 10 of 52 entries

Previous 1 2 3 4 5 6 Next

FIGURE 11.9: The “Demand specifications” tab, summarising the system demand specified in tabular format, including the safety margin prescribed.

Decision support

Instructions

System specifications

Algorithm specifications

Inputs

Select an objective function

Minimising probability of PGU failure

Maximising expected energy production

Select maximum processing time in hours

0.5 6 12

0.5 2 3.5 5 6.5 8 9.5 11 12

Select the cooling parameter

0.9

Select the reheating parameter

0.75

Select the epoch parameter

2

Select the soft constraint violation severity factor

0.75

Select the initial acceptance ratio

0.6

Initiate algorithm

FIGURE 11.10: The “Algorithm specifications” window through which the user can specify the objective function and parameter settings before initialising the SA algorithm in pursuit of a GMS solution.

table displays the demand for each time period of the scheduling window as well as the demand together with the specified safety margin, which will automatically update as the safety margin slider is altered. If the user would like to alter some of the data provided for either the system demand or PGU specifications of the particular power system, these changes have to be made in the CSV files uploaded after which the user will have to re-upload these altered files through the GUI shown in Figure 11.5. Once the user is completely satisfied with the data for both the system demand and the PGU specifications, he can navigate to the final window, called “Algorithm specifications.”

In the “Algorithm specifications” window, shown in Figure 11.10, the user is required to specify the desired parameter and other requirements of the SA algorithm. This includes the desired GMS objective function (selected by clicking on the radio buttons associated with the objective functions), the maximum allowable processing time in hours (selected by moving the slider associated with the processing time), as well as the five parameter values (selected by choosing values from the dropdown list) associated with the cooling parameter, the reheating parameter, the epoch management parameter, the soft constraint violation severity factor and the initial acceptance ratio, respectively. Once the user is satisfied with the selected objective function, maximum processing time and SA algorithm parameter values, he can click on the “Initiate algorithm” button. Clicking this button will execute the SA algorithm and the DSS will subsequently be occupied, solving the model. The duration for which the DSS may thus be occupied, can be as much as the maximum processing time specified, but can also terminate before the specified time, depending on the termination criteria specified, as described in §5.3.2. Whilst the algorithm is running, the status of the search can be monitored in the “Algorithm overview” tab shown in Figure 11.11. Whilst the algorithm is running, the status of the algorithm and the incumbent solution will be updated as the search progresses and will be indicated in the progress bar displayed. The progress bar will display the objective function of the current best solution, the processing time and whether or not the current solution is feasible. In this tab, a number of aspects of the algorithmic search can be followed, the current best objective function value, whether or not the solution associated with this objective function value is feasible, the processing time that has elapsed (in seconds) and the number of search iterations completed.

Once the algorithm has stopped running (when the progress bar disappears), the user can navigate to the “Maintenance schedule” tab, shown in Figure 11.12, where the incumbent solution is represented graphically. Like all the other graphs embedded in the DSS, the graphical represen-

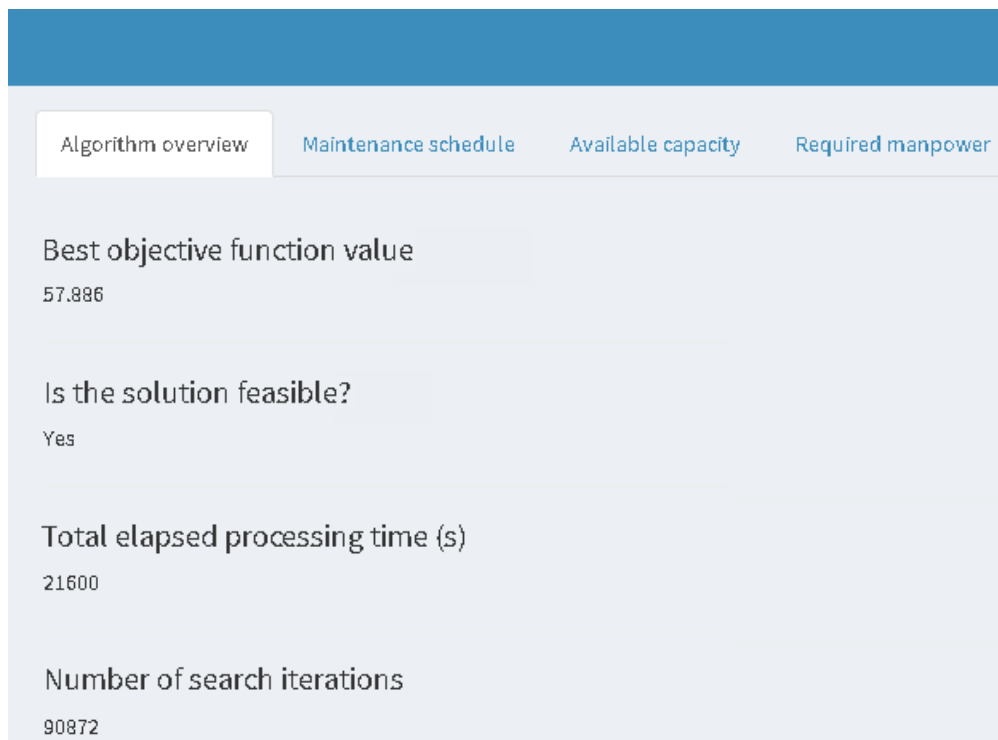


FIGURE 11.11: The “Algorithm overview” tab, containing information about the current incumbent solution (whether or not this solution is feasible) the processing time elapsed and the number of search iterations carried out.

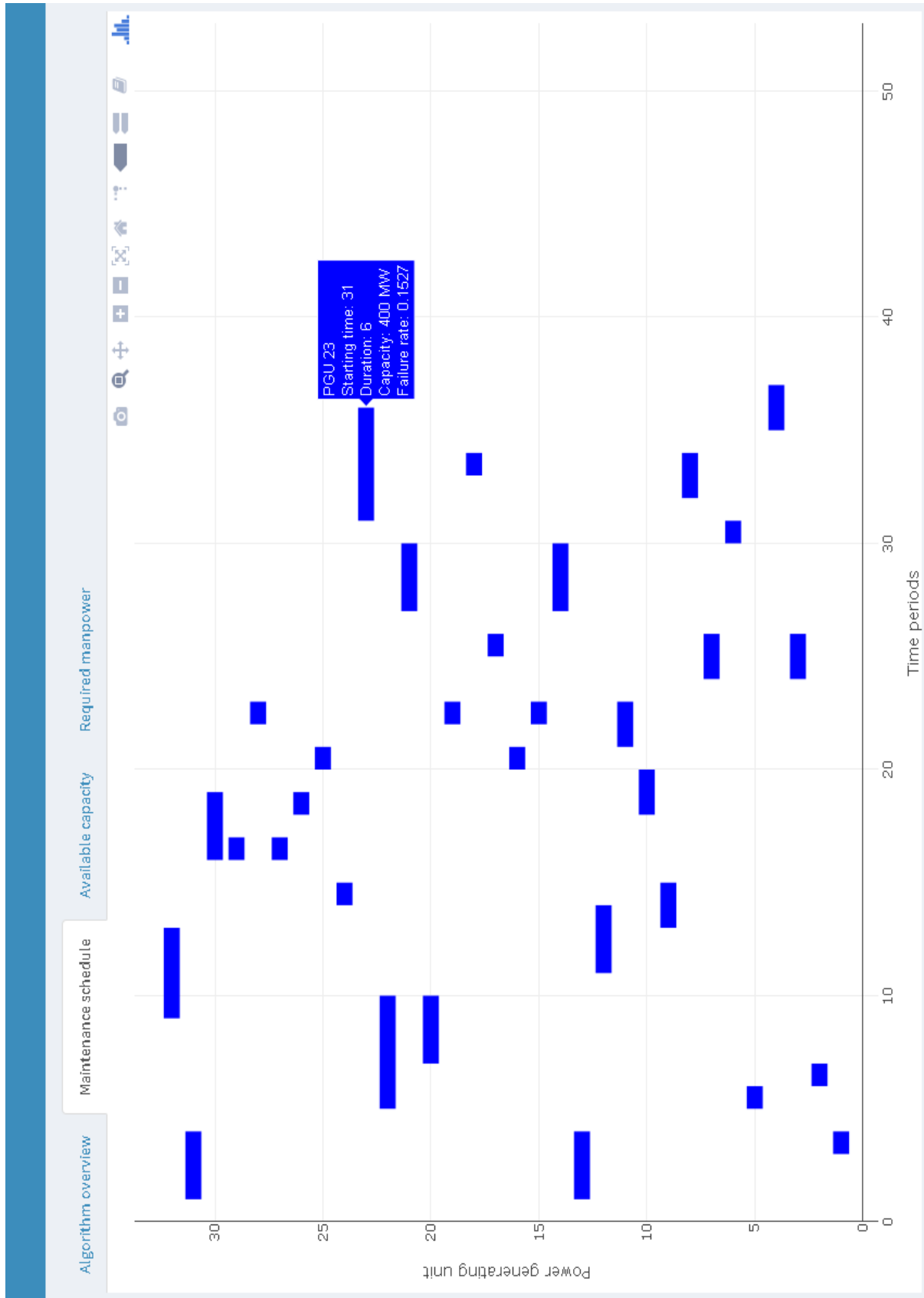


FIGURE 11.12: The “Maintenance schedule” tab, containing a graphical representation of the current incumbent solution.

Search:

Show 10 entries

PGU Starting times

PGU	Starting times
1	3
2	6
3	24
4	35
5	5
6	30
7	24
8	32
9	13
10	18

Showing 1 to 10 of 32 entries

Download solution

Previous 1 2 3 4 Next

FIGURE 11.13: The maintenance starting times for each PGU according to the final incumbent solution in tabular form.

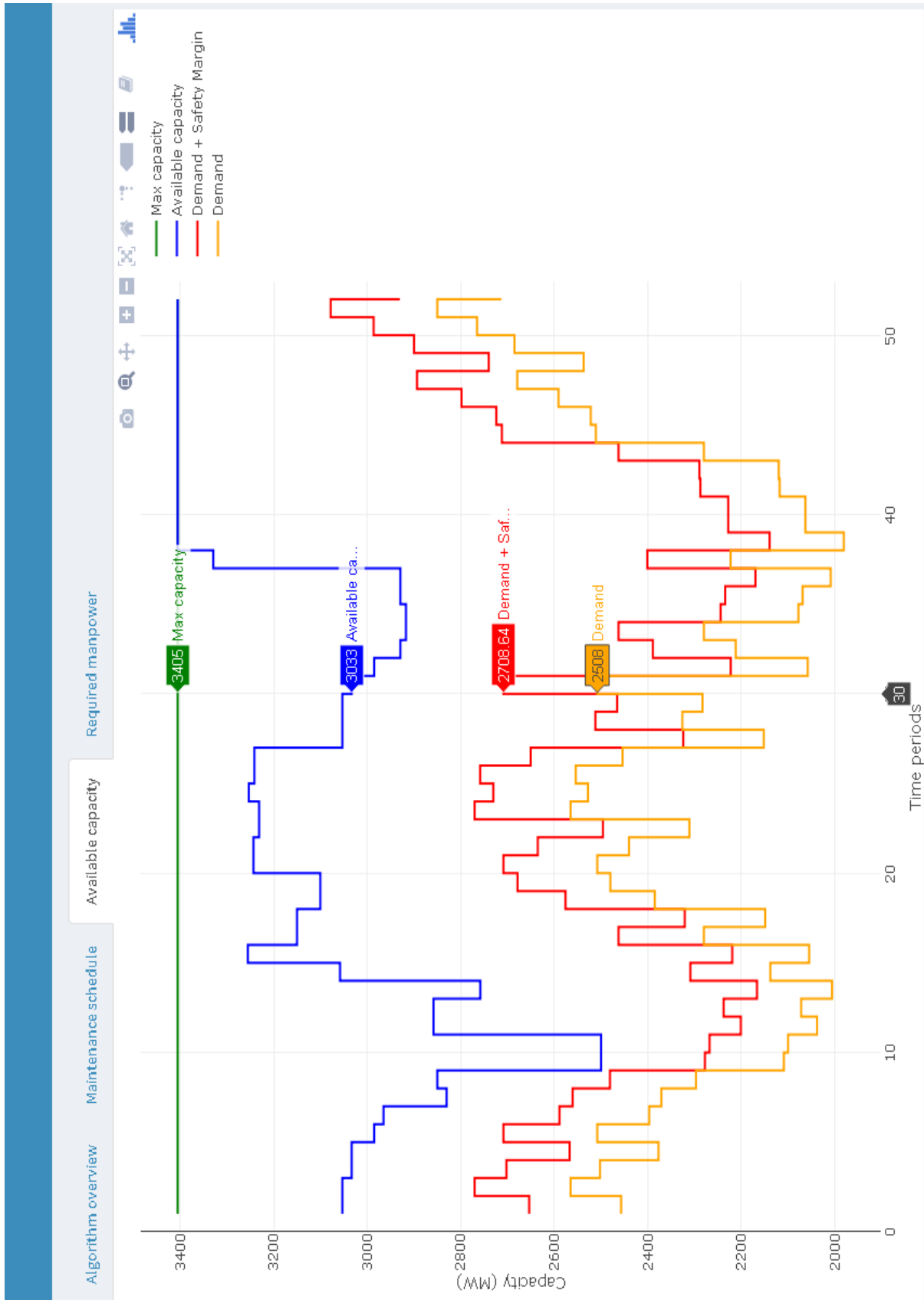


FIGURE 11.14: The “Available capacity” tab, containing a graphical representation of the maximum system capacity, the available system capacity, the system demand (together with the safety margin specified) and the system demand itself.



FIGURE 11.15: The “Required manpower” tab, containing a graphical representation of the manpower required to implement the final incumbent solution, as well as the maximum available manpower.

tation of the maintenance schedule is interactive. By hovering over a PGU with the computer mouse, the specifications of the specific PGU can be seen in a textbox appearing next to the PGU, as shown in Figure 11.12. This textbox displays the PGU's number/name, its starting time of maintenance, its duration of maintenance, its rated capacity (in MW) and its failure rate. From here, the user can scroll downwards to find the maintenance starting time of each PGU corresponding to the incumbent solution obtained. This solution is displayed in tabular format and can also be downloaded in CSV format to the user's personal computer, as shown in Figure 11.13.

The available capacity over the scheduling window associated with the incumbent solution can be obtained by navigating to the "Available capacity" tab, as shown in Figure 11.14. In this tab the user is presented with a graphical representation of the available capacity of the incumbent GMS solution, the maximum system capacity, the demand, and the demand together with the specified safety margin. This graph is again interactive and hovering over a certain point on the graph with the computer mouse will result in the value of each graph at that point being displayed. Similarly, for the manpower requirements (if checked previously, as shown in Figure 11.5), an interactive graphical representation of the manpower required for the maintenance schedule corresponding to the incumbent solution can be found by navigating to the "Manpower required" tab, as shown in Figure 11.15.

11.4 Chapter summary

A computerised DSS aimed at facilitating GMS decisions was designed by the author in RStudio [184] by employing a package called Shiny [41]. This DSS was presented in this chapter. Some general considerations of a typical DSS were discussed in §11.1, including a description of the three main DSS components (the database, the GUI and the model base). This was followed by a detailed process description of the DSS in §11.2. Finally, the system development was described in §11.3. This description included specifications on how to prepare the input data as well as a comprehensive system walk-through aimed at informing users of the DSS how to utilise the system to its full potential. The DSS may be found on the compact disc included at the back of this dissertation (the contents of which are described in Appendix B).

Part V

Conclusion

CHAPTER 12

Dissertation summary

Contents

12.1 Dissertation contents	261
12.2 Appraisal of dissertation contributions	264

This penultimate chapter contains a summary of the research reported in this dissertation and an appraisal of the contributions made in the dissertation.

12.1 Dissertation contents

In the introductory chapter to this dissertation, a brief historical overview was provided of the electric power system of South Africa, which is operated by the national power utility, Eskom. The overview included a brief description of the production and consumption of energy in South Africa over the last seventeen years as well as a description of the current state of the South African power system. This was followed by an informal description of the problem considered in this dissertation, namely to contribute models based on PGU failure probability minimisation and expected energy generation maximisation to the body of literature on GMS. The scope and objectives pursued during the completion of this research were also presented.

After the introductory chapter, the dissertation comprised a further eleven chapters (up to the chapter preceding this chapter) which were organised into five parts. In pursuit of the fulfilment of Dissertation Objective I of §1.3, a two-chapter part, Part I, was dedicated to a review of the relevant literature. In §2, which was the first chapter of that part, the reader was introduced to the GMS problem and a number of considerations to be taken into account when formulating a mathematical model for the GMS problem. This was followed by a description of the mathematical programming formulations of popular objective functions and constraint sets for the GMS problem encountered in the literature. A review was also presented of the various approaches adopted in the literature towards solving GMS models, including mathematical programming techniques, expert systems, fuzzy logic approaches, as well as the use of heuristics and meta-heuristics. A more detailed description was included, because this method was employed as an approximate solution methodology later in this dissertation.

The second chapter in the literature review part, §3, was devoted to a discussion on central notions in the field of reliability theory. Basic mathematical notations were reviewed which are typically adopted to represent ideas within the realm of reliability theory. This was followed by a description of two types of systems usually considered in reliability theory, namely non-repairable systems and repairable systems. A number of popular distribution models traditionally adopted

for each of these systems were also touched upon. The next section of the chapter contained an overview of the nature of system failure data and how these data may be employed to select appropriate failure models. Various methods were also discussed for determining whether a system is non-repairable or repairable, by employing trend tests. Before appropriate failure models can be applied, however, the parameters of these models have to be estimated, and various methods for doing so were presented, including the maximum likelihood method, the least squares method and the Bayesian parameter estimation method. An overview of acceleration models adopted in the literature was finally presented — these models may be employed to model systems that operate under high stress.

In the following chapter, §4, which was the first chapter of another two-chapter part on mathematical modelling, two mathematical models were formulated for the GMS problem which are capable of quantifying the reliability of a power system in terms of PGU failure and in terms of expected energy production, in fulfilment of Dissertation Objective II. The first section of the chapter contained a motivation for the choice of two newly proposed objective functions. In order to derive these objective functions mathematically, a number of assumptions had to be made and these were discussed in the next section. This was followed by a section dedicated to the actual GMS models adopted in this dissertation. It included a detailed derivation of the newly proposed objective functions (in fulfilment of Dissertation Objective III) — one linear and the other nonlinear — as well as the mathematical representation of the GMS model constraints. The constraint sets included in the mathematical models are energy demand satisfaction constraints, maintenance window constraints, maintenance resource constraints and maintenance exclusion constraints.

In order to solve the GMS models of §4, two methodologies were adopted. These methodologies were presented in the second chapter of Part II, §5. The first of these methodologies is an exact solution methodology (in partial fulfilment of Dissertation Objective VI) and the second is an approximate solution methodology (in final fulfilment of Dissertation Objective VI). The nonlinear model was linearised by carrying out a piecewise linear approximation of the nonlinear objective function. The resulting linearised model was solved exactly. This piecewise linearisation process was described in the first section of the chapter. In the second section, a description followed of the exact solution approach adopted, including a motivation for the choice of optimisation platform (CPLEX) as well as an elucidation of the model implementation within this platform. The description of the approximate solution methodology adopted in this dissertation, namely the method of SA, also contained a motivation for the choice of this solution methodology as well as a discussion on the implementation of the method of SA within the context of the GMS models of §4. This discussion covered the method employed for generating an initial solution and determining the initial temperature for the SA algorithm, the cooling and reheating schedules adopted, the constraint handling technique implemented, the epoch management protocol employed, the neighbourhood move operator incorporated in the algorithm and the manner in which the algorithm is terminated.

The effectiveness of the newly proposed GMS models, as well as that of the two model solution approaches, were tested in the contexts of two academic test systems from the literature. The two systems considered were a 21-unit test system and the celebrated 32-unit IEEE-RTS, both described in §6, which is the first chapter of the third part of this dissertation, devoted to a presentation of numerical results. The chapter was dedicated to a description of the input data and parameters pertaining to the aforementioned two test systems, in fulfilment of Dissertation Objective VII. The specifications for the 21-unit test system included a system demand (which is constant) over a scheduling horizon of fifty two weeks, while that for the IEEE-RTS exhibits a varying demand over a similar scheduling horizon.

The numerical results obtained when pursuing minimisation of the probability of PGU failure were presented in the second chapter of Part III, §7, in partial fulfilment of Dissertation Objective VIII. In the first section of the chapter, the results obtained when applying the exact solution approach to both the 21-unit test system and the IEEE-RTS were presented. This section included a presentation of optimal maintenance schedules in respect of both newly proposed GMS objective functions (obtained within 23 seconds of computation time and 382 seconds, respectively), an analysis of the manpower required and the available system capacity associated with these solutions, as well as comparisons between these solutions and solutions from the literature for the same test instance, but pursuing minimisation of the well-known sum of squared reserve margins as scheduling criterion. Sensitivity analyses were finally performed in order to analyse the feasibility of the exact solution approach in respect of small power systems such as the 21-unit system and the IEEE-RTS. The following section contained the numerical results obtained when employing the approximate solution approach of SA, in partial fulfilment of Dissertation Objective VIII. In this section, the results obtained in the parameter optimisation experiments for the 21-unit test system were presented in some detail and the best incumbent returned when solving the 21-unit system upon adoption of the best parameter combination values was described. A comparison of this solution with the optimal solution obtained by the exact solution approach was also carried out. It was found that the objective function values differed by only 0.0689%. A similar approach was taken with the presentation of the results of a corresponding parameter optimisation experiment for the IEEE-RTS, in final fulfilment of Dissertation Objective VIII. The best incumbent returned when solving the IEEE-RTS, and using the best combination of parameter values, was presented and again compared with the optimal solution obtained by the exact solution approach. It was found that the objective function values differed by 1.507% in this case.

The final chapter in Part III, §8, was dedicated to a presentation of the numerical results obtained by pursuing maximisation of the expected energy production, in final fulfilment of Dissertation Objective VIII. This chapter followed the same structure as §7 in that the first section was dedicated to a presentation of the exact solution obtained for the piecewise linear approximation of the nonlinear model in respect of both the 21-unit test system and the IEEE-RTS. An optimal solution to the piecewise linear approximation of the nonlinear model was obtained for the 21-unit system in 58 seconds. The difference in objective function values for the piecewise linear approximation and the nonlinear function amounts to 0.127%. The optimal solution obtained for the IEEE-RTS was found to require 93 538 seconds (25.983 hours) of computation time and corresponded to a 0.046% difference in objective function values for the piecewise linear approximation and the nonlinear function. Both optimal solutions were also compared with solutions from the literature for the same test instance, but aimed at minimisation of the well-known sum of squared reserve margins as scheduling criterion. These solutions were also compared to the optimal solutions obtained for the minimisation of the probability of unit failure scheduling criterion. Sensitivity analyses were finally performed for both the 21-unit test system and the IEEE-RTS in order to analyse the feasibility of the exact solution approach for small power systems. The second section of §8 contained a presentation of the numerical results obtained by employing the approximate solution approach of SA. The results obtained in the parameter optimisation experiments for the 21-unit test system were presented in some detail and the best incumbent returned when solving the 21-unit system upon adoption of the best combination of parameter values was described. A comparison of this solution with the optimal solution obtained by applying the exact solution approach (after piecewise linear approximation) was also carried out. It was found that the objective function values differed by only 2.669%. A similar approach was taken in the presentation of the results of a corresponding parameter optimisation experiment for the IEEE-RTS. The best incumbent returned by the SA algorithm

when adopting the best parameter combination values was presented and again compared with an optimal solution (after piecewise linear approximation). It was found that the objective function values differed by 1.618% in this case.

Part VI of this dissertation was dedicated to the application of the GMS models proposed to a real-world scenario. This part consisted of three chapters, the first of which contained a description of the real-world case study in some detail, in fulfilment of Dissertation Objective IX. The real-world case study was based on the energy grid of the national power utility in South Africa and is referred to as the 157-unit Eskom case study. In the first section of §9, some background was provided on the case study and this was followed by detailed specifications of the power system. Some extensions to both the linear and nonlinear GMS objectives were proposed in the final section of this chapter in order to accommodate the possibility of PGUs being scheduled for maintenance multiple time within a scheduling window.

The second chapter of Part VI, §10, contained a presentation and interpretation of the results obtained for the real-world case study by employing the approximate solution approach in the contexts of both the linear and nonlinear models, in fulfilment of Dissertation Objective X. In the first section, the results of a parameter optimisation experiment were presented for both scheduling criteria. In the second section, the incumbent solutions returned by the SA algorithm when employing the best combination of parameter values for both the proposed scheduling criteria were presented, in fulfilment of Dissertation Objective XI.

Part IV finally closed in §11 with the proposal of a computerised DSS, designed by the author, which is aimed at facilitating effective GMS decision making, in fulfilment of Dissertation Objectives IV and V. In the first section of the chapter, some general consideration was given to typical DSS development. This included a description of the three main components of a typical DSS and this was followed by a detailed process description of the particular GMS DSS proposed. The system was finally described in a comprehensive system walk-through fashion aimed at informing potential users how to utilise the DSS to its full potential.

12.2 Appraisal of dissertation contributions

This section contains a brief appraisal of the contributions of this dissertation. The dissertation contains a total of seven novel contributions. In each case the contribution is discussed in terms of its value and significance.

Contribution 1 *A novel scheduling criterion for the GMS problem which involves minimising the probability of PGU failure in the system over the scheduling window.*

The author could not find any GMS scheduling criteria in the literature which are based specifically on the risk of PGU failure. A novel GMS scheduling criterion was therefore developed which seeks to minimise the probability of PGU failure in the system. It was also desirable that the newly developed objective function should take into account the rated capacity of each PGU in the system. The reason for this is that when a PGU with a large capacity were to fail, the power system would lose a large proportion of its total capacity, whereas if a PGU with a small capacity were to fail, the effect of this event on the overall system would be relatively minor. Hence, the objective function proposed in this dissertation is also weighted by the normalised rated capacities of all the PGUs in the system. This weighting gives some preference with respect to earlier maintenance times of PGUs with large rated capacities.

In order to test the effectiveness of the proposed GMS objective function, it was included in a mathematical model and this model was solved in the context of two GMS test systems from the literature. The numerical results thus obtained were presented and analysed, and were also compared with solutions found in the literature according to a different scheduling criterion, in order to ascertain the extent of differences in scheduling approaches.

The proposed scheduling criterion was presented to a select group of experts at the Production Assurance Department of the South African national power utility, Eskom, in September 2016. These individuals found the proposed scheduling criterion very useful and positive feedback was received in respect of the modelling approach taken.

Contribution 2 *A novel scheduling criterion for the GMS problem which involves maximisation of the expected energy production over the scheduling window.*

The author could also not find any GMS scheduling criteria in the literature which are based specifically on the total amount of energy that the entire system is expected to generate over the scheduling window when taking into account PGU failures. A novel GMS scheduling criterion was therefore developed which seeks to maximise the expected energy produced over the scheduling window. For this criterion, the timing of a failure of any PGU in the system was included as a random variable. The expected energy may then be calculated based on when a PGU is scheduled for maintenance, taking into account the probability of a failure occurring.

In order to test the effectiveness of the proposed GMS objective function, this criterion was also included in a mathematical model for the GMS problem and this model was solved in the context of the same two GMS test systems from the literature mentioned above. The results were presented and analysed, and were also compared with solutions in the literature found according to a different scheduling criterion in order to ascertain the extent of differences in scheduling approaches.

Although this proposed scheduling criterion was not presented to the South African national power utility, Eskom, it is envisaged that a formal meeting will be arranged in the near future to present this scheduling criterion to the Production Assurance Department of Eskom.

Contribution 3 *An exact approach towards solving instances of the GMS problem in which the newly proposed objective functions are adopted as scheduling criteria.*

As mentioned, the author could not find any instances in the literature of a GMS scheduling criterion based specifically on the reliability of the PGUs in the system. It was therefore desirable to establish high-quality benchmark results for scheduling criteria based on PGU failure rates. For both scheduling criteria, an exact solution approach was consequently implemented within the context of the aforementioned two GMS test instances. In this manner, solutions obtained by employing another solution approach may in the future be compared to optimal solutions established in this dissertation in order to analyse the effectiveness of other solution approaches. The exact solution approach adopted for solving instances of the mathematical model of §4, was implemented in CPLEX, which employs a branch-and-cut method in combination with various heuristics to obtain optimal solutions.

The novel scheduling criterion which seeks to minimise the probability of PGU failure is of a linear nature and therefore could be implemented “as-is” in CPLEX, which only accommodates linear models in its basic form. The scheduling criterion which seeks to maximise the expected

energy production is, however, nonlinear and therefore the scheduling criterion was subjected to piecewise linearisation in order to be able to apply the branch-and-cut approach embedded in CPLEX.

The results obtained by means of the exact solution approach for the linear model of §4 were summarised in a peer-reviewed journal paper [78], which was accepted for publication in August 2017 and will be published in 2018. The results obtained from the exact solution approach for the nonlinear model of §4 was also summarised in a second peer-reviewed journal paper [79], which has been submitted for publication.

Contribution 4 *A sensitivity analysis in terms of the feasibility of the exact solution approach for medium to large instances of both GMS models of §4.*

In many cases in the literature, an exact approach toward solving realistic instances of GMS models is not feasible and therefore many authors employ (meta)heuristic solution approaches to solve their models. The computation time expended to solve GMS models may vary, based on the problem instances considered. A sensitivity analysis was performed in this dissertation in order to ascertain the feasibility of the exact solution approach of Contribution 3. This was achieved by creating various GMS problem instances, similar to the two test systems of §6, but with either tighter or more relaxed demand constraints as well as variations in the earliest and latest maintenance starting times of the various PGUs.

It was found that the exact solution approach applied to the linear model of §4.3 is viable for small problem instances that are tightly constrained. As the constraints are relaxed, however, or the number of PGUs in the system increases, the exact solution becomes exorbitantly expensive in terms of computation time required — hence necessitating the development of a (meta)heuristic solution approach for the linear model in §4.3.

For the exact solution approach applied to the nonlinear model of §4.3, it was found, even for some of the basic cases where none of the constraints were relaxed, that the approach (employing piecewise linearisation) is expensive in terms of computation time required. This, once again, necessitated the development of a (meta)heuristic solution approach for the linear nonlinear model in §4.3.

The results of the sensitivity analysis for both the newly proposed scheduling criteria were also included in the peer-reviewed papers mentioned in respect of Contribution 3 [79, 78].

Contribution 5 *An approximate solution approach, based on the method of SA, for solving instances of the GMS problem in which the newly proposed objective functions are adopted as scheduling criteria.*

The trajectory-based method of SA was implemented as an approximate solution approach in this dissertation. This metaheuristic is flexible and may be employed to solve even large and/or unconstrained instances of the GMS problem. After having performed an extensive parameter optimisation experiment to find the best combination of SA parameters for both test systems of §6 in the contexts of both scheduling criteria proposed, the results obtained upon solving the model of §4.3 for the test systems of §6 were compared with solutions obtained by the exact solution approach. It was found, for both proposed scheduling criteria, that although the SA algorithm required more computation time in some instances, it is a viable solution approach capable of uncovering solutions of very high quality.

Contribution 6 *Application of both scheduling criteria to a real-world case study based on the power system of the national power utility of South Africa.*

The two newly proposed scheduling criteria of §4.3 were applied to a real-world case study based on the power system of Eskom, the national power utility of South Africa. This power system contains 105 PGUs, which is a much larger number of PGUs than that contained in either of the two academic benchmark systems previously considered. The resulting 157-unit Eskom case study contains PGUs that require multiple maintenance procedures as well as PGUs that require no maintenance within the scheduling window. In order to accommodate these specifications, a number of adaptations and extensions were made to the proposed scheduling criteria. These adaptations and extensions were only made to the scheduling criteria themselves and not to the mathematical model constraints of §4.3.

After having carried out an extensive parameter evaluation for the SA algorithm with respect to the 157-unit Eskom case study (for both scheduling criteria), the best parameter combinations were employed to solve the problem instance. The numerical results thus obtained were reported. It is envisaged that these results will be summarised in a third peer-reviewed journal paper emanating from the work in this dissertation.

Contribution 7 *A computerised GMS decision support system in §11.*

A computerised GMS decision support system was developed in this dissertation which is capable of solving user-specified instances of the GMS problem, including the real-world case study of Contribution 6. This DSS may be employed by an operations scheduler tasked with the responsibility of scheduling PGUs for maintenance. The system may assist the user by proposing good maintenance schedules based on either of the two scheduling criteria proposed in §4.3

CHAPTER 13

Possible future work

Contents

13.1 Incorporating PGU generation capability variation	269
13.2 Enforcing a maintenance interval constraint	270
13.3 Failure frequency generalisation	270
13.4 Incorporating PGUs with increasing failure rates	270
13.5 The possibility of failures being dependent on one another	271
13.6 Reliability accuracy improvement in the linear GMS model	271
13.7 Adopting constraint programming as solution approach	271
13.8 Adopting a multi-objective GMS paradigm	272

During the process of completing the research reported in this dissertation, a number of aspects were identified as areas of possible improvements and enhancements of the modelling approach adopted. This chapter therefore contains a number of suggestions with respect to possible future research which may be pursued as follow-up work to the contributions of this dissertation.

13.1 Incorporating PGU generation capability variation

It was assumed, for the purposes of the GMS models in this dissertation, that during each time period of the scheduling window, each of the PGUs generate 100% of their rated capacities when in operation. This is a simplification of how PGUs operate in real energy systems. In practice, there are a minimum and a maximum percentage of the rated capacity between which each PGU should generate energy when in operation. It is, for instance, not desirable to operate a very large and expensive PGU at full capacity if only 10% of its rated capacity is, in fact, required to meet the demand plus the safety margin of the power system.

For this reason, it is suggested that a capability of specifying the allowable range within which each PGU can generate energy be added to the GMS model of §4. Such a capability may be introduced in the form of a parameter or a decision variable. If introduced as a parameter, the allowable range within which a PGU can generate energy should be specified, as part of the problem instance specifications, for each PGU during each time period of the scheduling window. If introduced as a decision variable, a method of giving generation preference should also be introduced in order to dedicate certain PGUs to generate energy during certain time periods. This generation preference may be based on the cost of energy generation (*i.e.* PGUs may be ranked from cheapest to most expensive and be employed in this order to generate

energy based on generation cost) or on the type of fuel a PGU uses (*i.e.* certain PGUs may be employed to generate energy rather than others based on the type of PGUs).

This research suggestion therefore in effect involves including the UC problem (described in §2.1.5) within the current GMS model formulation of §4 so as to identify which PGUs should be employed to generate energy during which time period of the scheduling window and at what generating capacity they should operate.

13.2 Enforcing a maintenance interval constraint

It was found in this dissertation that scheduling planned maintenance for PGUs according to the minimisation of probability of unit failure scheduling criterion proposed in §4 results in maintenance of a PGU being scheduled as early as possible during the scheduling window. This may cause a situation where no PGUs are scheduled for maintenance toward the end of the scheduling window (as was, for example, seen in Figure 7.5). During these final time periods, when no maintenance is scheduled for any PGUs, the maximum system capacity is available and no manpower is required to perform maintenance. This may cause a problem for a power utility as the personnel required to perform maintenance on the PGUs will be idle during these time periods when no maintenance is scheduled if maintenance personnel are not employed on a contractual basis.

It is therefore suggested that an additional maintenance interval constraint be introduced to the linear GMS model of §4. This constraint may specify a shortest time interval between consecutive maintenance operations scheduled for any PGU. This time interval may either be specified in absolute terms (*i.e.* in terms of units of time) or may be specified in terms of the reliability level of the PGUs, based on the power utility's maintenance philosophy.

13.3 Failure frequency generalisation

It was assumed, for the purposes of modelling the GMS problem in this dissertation, that each PGU fails at most once during a scheduling window. In practice, however, this may not be the case. It might instead be the case that certain PGUs fail more than once during the scheduling window and that others very rarely fail during a scheduling window. This may occur due to certain PGUs exhibiting different failure rates (PGUs with high failure rates are expected to fail more frequently than those with lower failure rates).

It is therefore suggested that a capability be included in the GMS models of §4 to account for the situation where a PGU may fail more than once during the scheduling window. This may be achieved by determining the expected failure dates of each PGU and from this, derive the expected number of failures within the scheduling window for each PGU.

13.4 Incorporating PGUs with increasing failure rates

In this dissertation, it was assumed that the failures of the PGUs in a specified power system follow an exponential distribution with a constant failure rate. In practice, however, PGUs do not always exhibit constant failure rates. In power systems which contain ageing PGUs, for example, the failures may follow a different distribution (as described in §3.3 and §3.4) with a rate that is either increasing or decreasing. PGUs may therefore start to fail more frequently as

they age when exhibiting a increasing failure rate or may fail less frequently when exhibiting a decreasing failure rate.

A number of distributions other than the exponential distribution were described in §3.3 and §3.4 which may be adopted in conjunction with constant, increasing and decreasing failure rates. An appropriate distribution may be selected for each PGU in the system and, depending on its historical failure data, may exhibit either a constant an increasing or a decreasing failure rate.

13.5 The possibility of failures being dependent on one another

One of the fundamental assumptions of this dissertation is that the failures of PGUs in the generation system are independent from one another. This may be considered a fair assumption as the failure of one PGU does not directly influence that of another PGU. Failure of one PGU may, however, cause the other PGUs in the power system to operate under higher stress conditions in order to meet the demand of the system. There may also be some other indirect dependencies of the PGUs, such as the weather conditions affecting their operation or PGUs at the same power station being subjected to the same operating conditions.

Therefore, the scheduling criteria proposed in this dissertation may be generalised so as to be able to accommodate situations where the failure rates of certain PGUs are dependent on one another. In this case, the probability of PGU failure cannot merely be determined by taking the product of the individual probabilities of unit failure.

13.6 Reliability accuracy improvement in the linear GMS model

In order to accommodate the possibility of PGUs being scheduled for maintenance more than once, certain adaptations had made to the scheduling criteria in §9.3. The adaptation made with respect to the linear model (minimising the probability of unit failure) involved the introduction of virtual, duplicate PGUs which contribute no additional capacity to the total system capacity. A difficulty arose in this regard, however, due to PGU reliability being dependent on the time duration since the last maintenance procedure carried out on a PGU, which is not known beforehand for PGUs that require multiple maintenance procedures, as it depends on the decision variables. This was remedied by assuming a fixed length of time $|x'_u|$ that had elapsed between the previous maintenance procedure and the latest maintenance procedure. It was therefore assumed that the time intervals between multiple maintenance procedures of the same PGU during a scheduling window are the same.

This is, however, not the case in reality, as was described in §9.3.1. Each PGU is actually 100% reliable after its returns to operation upon completion of its $(i-1)$ -th maintenance procedure and not, as estimated, after a time duration of $e_{u,i} + x'_u$. The PGU reliability may more accurately be determined by incorporating the previous maintenance date of a PGU that requires multiple maintenance procedures as a decision variable. This would, of course, increase the complexity of the model.

13.7 Adopting constraint programming as solution approach

Constraint programming (CP) is a computer programming paradigm in which relationships between variables of a model are stated in the form of constraints. Typically, the main focus

in an IP context is to obtain the optimal objective function value, whereas CP is more focused on obtaining feasible solutions by satisfying each of the constraints of the model [116, 183]. CP may easily be implemented in a situation where the model is nonlinear, therefore avoiding the need for piecewise linear approximations of nonlinear functions. CP has been shown to find good solutions to scheduling problems in the literature — so much so that the celebrated off-the-shelf software suite IBM ILOG CPLEX Optimization Studio [112, 115] contains a package for solving scheduling problems by means of CP [117].

A CP solution approach may therefore be adopted in order to possibly improve upon the computation time required to solve large instances of GMS models (linear or nonlinear) to optimality. This approach may also allow an optimal solution to be obtained for a nonlinear model in stead of an optimal solution for the piecewise linear approximation of the nonlinear model.

13.8 Adopting a multi-objective GMS paradigm

There are not many cases in the literature where the GMS problem is solved within a multi-objective optimisation paradigm. The two newly proposed scheduling criteria in this dissertation are, however, somewhat conflicting. This might also be the case for a number of other GMS criteria proposed in the literature. It is therefore suggested that the scheduling criteria proposed in this dissertation be employed within a multi-objective modelling approach, which is capable of yielding trade-off solutions. These solutions may then be analysed by the user on a *post hoc* basis in order to decide subjectively which solution within the set of trade-off schedules is best suited to meet all the scheduling requirements.

Examples of scheduling criteria from the literature that may be considered for inclusion on a multi-objective GMS modelling paradigm are:

- minimising the probability of unit failure,
- maximising the expected energy production,
- minimising the SSR,
- minimising the energy production cost,
- minimising the energy production cost and maintenance cost,
- minimising the LOLP, and
- minimising the EUE.

References

- [1] AARTS EHL, VAN LAARHOVEN PJM, LENSTRA JK & ULDER NLJ, 1994, *A computational study of local search algorithms for job shop scheduling*, ORSA Journal on Computing, **6**(2), pp. 118–125.
- [2] ABRAMSON D, AMOORTHY MK & DANG H, 1999, *Simulated annealing cooling schedules for the school timetabling problem*, Asia-Pacific Journal of Operational Research, **16**(1), pp. 1–22.
- [3] ADAMIDIS K & LOUKAS S, 1998, *A lifetime distribution with decreasing failure rate*, Statistical and Probability Letters, **39**(1), pp. 35–42.
- [4] AHMAD A & KOTHARI DP, 2000, *A practical model for generator maintenance scheduling with transmission constraints*, Electric Machines and Power Systems, **28**(6), pp. 501–513.
- [5] AHMAD A & KOTHARI DP, 1998, *A review of recent advances in generator maintenance scheduling*, Electric Machines and Power Systems, **26**(4), pp. 373–387.
- [6] AIMSS, 2016, *AIMMS Optimization modeling*, [Online], [Cited October 1st, 2016], Available from <https://goo.gl/836TYv>.
- [7] ALARDHI M & LABIB A, 2008, *Preventive maintenance scheduling of multi-cogeneration plants using integer programming*, Journal of the Operational Research Society, **60**, pp. 503–509.
- [8] ALBRECHT PF, BHAVARAJU MP, BIGGERSTAFF BE, BILLINGTON R, JORGENSEN GE, REPPEN ND & SHORTLEY PB, 1979, *IEEE Reliability test system*, IEEE Transactions on Power Apparatus and Systems, **PAS-98**(6), pp. 2047–2054.
- [9] ALLAN RN, BILLINGTON R & ABDEL-GAWAD NMK, 1986, *The IEEE Reliability test system — Extensions to and evaluation of the generating system*, IEEE Transactions on Power Systems, **PWRS-1**(4), pp. 1–7.
- [10] ANDERSON TW & DARLING DA, 1954, *A test of goodness of fit*, Journal of the American Statistical Association, **49**(268), pp. 765–769.
- [11] ASCHER H & FEINGOLD H, 1984, *Repairable systems-modeling, inference, misconceptions and their causes*, Marcel Dekker, New York (NY).
- [12] ASIF M & MUNEER T, 2007, *Energy supply, its demand and security issues for developed and emerging economies*, Renewable and Sustainable Energy Reviews, **11**, pp. 1388–1413.
- [13] BADRI A & NIAZI AN, 2012, *Preventive generation maintenance scheduling considering system reliability and energy purchase in restructured power systems*, International Journal of Basic and Applied Scientific Research, **12**, pp. 12773–12786.
- [14] BAI J & PERRON P, 2003, *Computation and analysis of multiple structural change models*, Journal of Applied Econometrics, **18**(1), pp. 1–22.

- [15] BAI J, 1997, *Estimation of a change point in multiple regression models*, Review of Economics and Statistics, **79(4)**, pp. 551–563.
- [16] BAKIRTZIS A, PETRIDIS V & KAZARLIS S, 1994, *Genetic algorithm solution to the economic dispatch problem*, Generation, Transmission and Distribution, **141(4)**, pp. 377–382.
- [17] BALAS E, 1965, *An additive algorithm for solving linear programs with zero-one variables*, Operations Research, **13(4)**, pp. 517–546.
- [18] BANDYOPADHYAY S, SAHA S, MAULIK U & DEB K, 2008, *A simulated annealing-based multiobjective optimization algorithm: AMOSA*, IEEE Transactions on Evolutionary Computation, **12(3)**, pp. 269–283.
- [19] BARLOW RE & PROSCHAN F, 1975, *Statistical theory of reliability and life testing: Probability models*, (Unpublished) Technical Report, Department of Industrial Engineering & Operations Research, Florida State University, Tallahassee (FL).
- [20] BARLOW RE, PROSCHAN F & HUNTER LC, 1965, *Mathematical theory of reliability*, 1st Edition, John Wiley & Sons, New York (NY).
- [21] BAROT H & BHATTACHARYA K, 2008, *Security coordinated maintenance scheduling in deregulation based on GENCO contribution to unserved energy*, IEEE Transactions on Power Systems, **23(4)**, pp. 1871–1882.
- [22] BASKAR S, SUBBARAJ P, RAO M & TAMILSELVI S, 2003, *Genetic algorithms solution to generator maintenance scheduling with modified genetic operators*, Proceedings of the 37th Generation, Transmission and Distribution Conference, Staffordshire, pp. 56–60.
- [23] BASU AP & SUN K, 1997, *Multivariate exponential distributions with constant failure rates*, Journal of Multivariate Analysis, **61(2)**, pp. 159–169.
- [24] BELLMAN R & ROTH R, 1969, *Curve fitting by segmented straight lines*, Journal of the American Statistical Association, **64(327)**, pp. 1079–1084.
- [25] BERALDI P & BRUNI ME, 2010, *An exact approach for solving integer problems under probabilistic constraints with random technology matrix*, Annals of Operations Research, **177(1)**, pp. 127–137.
- [26] BERGER JO, 2013, *Statistical decision theory and Bayesian analysis*, 2nd Edition, Springer, New York (NY).
- [27] BILLINTON R & ABDULWHAB A, 2003, *Short-term generating unit maintenance scheduling in a deregulated power system using a probabilistic approach*, Generation, Transmission and Distribution, **150(4)**, pp. 463–468.
- [28] BISANOVIC S, HAJRO M & DLAKIC M, 2011, *A profit-based maintenance scheduling of thermal power units in electricity market*, International Journal of Electrical and Electronics Engineering, **5(3)**, pp. 156–164.
- [29] BLUM A & GUPTA A, 2013, *Design & analysis of algorithms*, [Lecture notes], Algorithms (CMU 15-451/651), Carnegie Mellon University, New York (NY).
- [30] BLUM C & ROLI A, 2003, *Metaheuristics in combinatorial optimization: Overview and conceptual comparison*, ACM Computing Surveys (CSUR), **35(3)**, pp. 268–308.
- [31] BOYLAN C, 2016, *Decision support systems — DSS (definition)*, [Online], [Cited June 15th, 2017], Available from <http://www.informationbuilders.com/decision-support-systems-dss>.

- [32] BRANDT F, BAUER R, VÖLKER M & CARDENEO A, 2013, *A constraint programming-based approach to a large-scale energy management problem with varied constraints*, Journal of Scheduling, **16(6)**, pp. 629–648.
- [33] BROWN RL, DURBIN J & EVANS JM, 1975, *Techniques for testing the constancy of regression relationships over time*, Journal of the Royal Statistical Society, Series B (Methodological), pp. 149–192.
- [34] BROWNLEE J, 2015, *Tabu search*, [Online], [Cited March 1st, 2016], Available from http://www.cleveralgorithms.com/nature-inspired/stochastic/tabu_search.html.
- [35] BULJUBASI M & GAVRANOVIĆ H, 2012, *Orchestrating constrained programming and local search to solve a large scale energy management problem*, Proceedings of the 2012 Federated Conference on Computer Science and Information Systems (FedCSIS), Wroclaw, pp. 371–378.
- [36] BURKE EK, CLARKE JA & SMITH AJ, 1998, *Four methods for maintenance scheduling*, Proceedings of the Artificial Neural Nets and Genetic Algorithms Conference, Norwich, pp. 264–269.
- [37] BURKE E & SMITH A, 2000, *Hybrid evolutionary techniques for the maintenance scheduling problem*, IEEE Transactions on Power Systems, **15(1)**, pp. 122–128.
- [38] BUSETTI F, 2003, *Simulated annealing overview*, [Online], [Cited December 7th, 2015], Available from <https://goo.gl/RadpTc>.
- [39] CANTO SP, 2008, *Application of Benders' decomposition to power plant preventive maintenance scheduling*, European Journal of Operational Research, **184(2)**, pp. 759–777.
- [40] CASSADY C & POHL E, 2003, *Introduction to repairable systems modeling*, [Tutorial], Annual Reliability and Maintainability Symposium, Fayetteville (AR).
- [41] CHANG W, CHENG J, ALLAIRE JJ, XIE Y & MCPHERSON J, 2016, *Shiny: Web application framework for R*, [Online], [Cited September 20th, 2017], Available from <https://cran.r-project.org/web/packages/shiny/index.html>.
- [42] CHAREST M & FERLAND JA, 1993, *Preventive maintenance scheduling of power generating units*, Annals of Operations Research, **41(3)**, pp. 185–206.
- [43] CHATTOPADHYAY D, 1998, *A practical maintenance scheduling program mathematical model and case study*, IEEE Transactions on Power Systems, **13(4)**, pp. 1475–1480.
- [44] CHEN GJ & TANG GQ, 2005, *Tabu search-ant colony optimization hybrid algorithm based distribution network planning*, Power System Technology, **2**, pp. 1–5.
- [45] CHOWDHURY A, AGARWAL SK & KOVAL DO, 2003, *Reliability modeling of distributed generation in conventional distribution systems planning and analysis*, IEEE Transactions on industry applications, **39(5)**, pp. 1493–1498.
- [46] CONTAXIS GC, KAVATZA SD & VOURNAS CD, 1989, *An interactive package for risk evaluation and maintenance scheduling*, IEEE Transactions on Power Systems, **4(2)**, pp. 389–395.
- [47] COX DR, 1955, *Some statistical methods connected with series of events*, Journal of the Royal Statistical Society, **17(2)**, pp. 129–164.
- [48] COX DR & LEWIS PA, 1966, *Statistical analysis of series of events*, Methuen's Monographs on Applied Probability and Statistics, John Wiley, London.

- [49] CROWLEY K & JANSE VAN VUUREN A, 2014, *Eskom declares power emergency, asks South Africa to cut use*, [Online], [Cited February 18th, 2015], Available from <http://www.bloomberg.com/news/articles/2014-02-20/eskom-declares-power-emergency-and-asks-south-africa-to-cut-use>.
- [50] CZYZŻAK P & JASZKIEWICZ A, 1998, *Pareto simulated annealing — A metaheuristic technique for multiple-objective combinatorial optimization*, *Journal of Multi-Criteria Decision Analysis*, **7(1)**, pp. 34–47.
- [51] DA SILVA E, SCHILLING MT & RAFAEL M, 2000, *Generation maintenance scheduling considering transmission constraints*, *IEEE Transactions on Power Systems*, **15(2)**, pp. 838–843.
- [52] DAHAL KP, ALDRIDGE CJ & McDONALD JR, 1999, *Generator maintenance scheduling using a genetic algorithm with a fuzzy evaluation function*, *Fuzzy Sets and Systems*, **102(1)**, pp. 21–29.
- [53] DAHAL KP & CHAKPITAK N, 2007, *Generator maintenance scheduling in power systems using metaheuristic-based hybrid approaches*, *Electric Power Systems Research*, **77(7)**, pp. 771–779.
- [54] DAHAL KP & McDONALD JR, 1997, *A review of generator maintenance scheduling using artificial intelligence techniques*, *Proceedings of the Universities Power Engineering Conference (UPEC '97)*, University of Manchester, Manchester.
- [55] DAHAL KP & McDONALD JR, 1997, *Generator maintenance scheduling of electric power systems using genetic algorithms with integer representation*, *Proceedings of the Second International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications*, Hong Kong, pp. 456–461.
- [56] DAHAL K, AL-ARFAJ K & PAUDYAL K, 2015, *Modelling generator maintenance scheduling costs in deregulated power markets*, *European Journal of Operational Research*, **240(2)**, pp. 551–561.
- [57] DIGALAKIS JG & MARGARITIS KG, 2002, *A multipopulation cultural algorithm for the electrical generator scheduling problem*, *Mathematics and Computers in Simulation*, **60(3)**, pp. 293–301.
- [58] DING XD, YAO ZG & CHENG G, 2009, *Lingo language in the logistics distribution center location*, *Logistics Engineering and Management*, **10**, pp. 1–30.
- [59] DOPAZO JF & MERRILL HM, 1975, *Optimal generator maintenance scheduling using integer programming*, *IEEE Transactions on Power Apparatus and Systems*, **94(5)**, pp. 1537–1545.
- [60] DORIGO M & BLUM C, 2005, *Ant colony optimization theory: A survey*, *Theoretical Computer Science*, **344(2)**, pp. 243–278.
- [61] DOWSLAND KA, 1998, *Nurse scheduling with tabu search and strategic oscillation*, *European Journal of Operational Research*, **106(2)**, pp. 393–407.
- [62] DRÉO J, PETROWSKI A, SIARRY P & TAILLARD E, 2006, *Metaheuristics for hard optimization: Methods and case studies*, 1st Edition, Springer Science & Business Media, Berlin.
- [63] D'SA A, 2005, *Integrated resource planning (IRP) and power sector reform in developing countries*, *Energy Policy*, **33**, pp. 1271–1285.
- [64] EDWIN KW & CURTIUS F, 1990, *New maintenance-scheduling method with production cost minimization via integer linear programming*, *International Journal of Electrical Power and Energy Systems*, **12(3)**, pp. 165–170.

- [65] EKPENYONG UE, ZHANG J & XIA X, 2012, *An improved robust model for generator maintenance scheduling*, Electric Power Systems Research, **92**, pp. 29–36.
- [66] EL-AMIN I, DUFFUAA S & ABBAS M, 2000, *A tabu search algorithm for maintenance scheduling of generating units*, Electric Power Systems Research, **54(2)**, pp. 91–99.
- [67] EL-SHARKH MY & EL-KEIB AA, 2003, *An evolutionary programming-based solution methodology for power generation and transmission maintenance scheduling*, Electric Power Systems Research, **65(1)**, pp. 35–40.
- [68] EL-SHARKH MY & EL-KEIB AA, 2003, *Maintenance scheduling of generation and transmission systems using fuzzy evolutionary programming*, IEEE Transactions on Power Systems, **18(2)**, pp. 862–866.
- [69] EL-SHARKH M, EL-KEIB A & CHEN H, 2003, *A fuzzy evolutionary programming-based solution methodology for security-constrained generation maintenance scheduling*, Electric Power Systems Research, **67(1)**, pp. 67–72.
- [70] ELYAS SH, FOROUD AA & CHITSAZ H, 2013, *A novel method for maintenance scheduling of generating units considering the demand side*, International Journal of Electrical Power & Energy Systems, **51**, pp. 201–212.
- [71] ENDRENYI J, ANDERS GJ, ASGARPOOR S, BILLINTON R, DIALYNAS EN, CHOWDHURY N, ALLAN RN, FLETCHER RH, MCCALLEY J & MELIPOULOS S, 2001, *The present status of maintenance strategies and the impact of maintenance on reliability*, IEEE Transactions on Power Systems, **16(4)**, pp. 638–646.
- [72] ENERDATA, 2016, *Electricity consumption — World Power Consumption*, [Online], [Cited October 2nd, 2016], Available from <https://goo.gl/7gnDNO>.
- [73] ESCOBAR LA & MEEKER WQ, 2006, *A review of accelerated test models*, Statistical Science, **21(4)**, pp. 552–577.
- [74] ESHRAGHIA R, SHANECHI MM & MASHHADI HR, 2006, *A new approach for maintenance scheduling of generating units in power market*, Proceedings of the International Conference on Probabilistic Methods Applied to Power Systems (PMAPS 2006), Stockholm, pp. 1–7.
- [75] ESKOM HOLDINGS LIMITED, 2014, *Company information*, [Online], [Cited February 16th, 2015], Available from http://www.eskom.co.za/OurCompany/CompanyInformation/Pages/Company%5C_Information%5C_1.aspx.
- [76] ESKOM HOLDINGS LIMITED, 2017, *Eskom generation plant mix*, [Online], [Cited October 2nd, 2016], Available from <http://www.eskom.co.za/OurCompany/SustainableDevelopment/ClimateChangeCOP17/Documents/GenerationMix.pdf>.
- [77] ESKOM HOLDINGS LIMITED, 2015, *Palmiet pumped storage scheme*, [Online], [Cited March 30th, 2014], Available from http://www.eskom.co.za/AboutElectricity/FactsFigures/Documents/HY_0004PalmietPumpedStorageSchemeCivilWorksRev4.pdf.
- [78] EYGELAAR J, LÖTTER DP & VAN VUUREN JH, 2018, *Generator maintenance scheduling based on the risk of power generating unit failure*, International Journal of Electrical Power & Energy Systems, **95C**, pp. 83–95.
- [79] EYGELAAR J & VAN VUUREN JH, 2017, *Generator maintenance scheduling in pursuit of expected energy generation maximisation*, European Journal of Operations Research, (Submitted).
- [80] FACEBOOK, 2017, *Facebook*, [Online], [Cited May 12th, 2017], Available from <https://www.facebook.com/>.

- [81] FATTAHI M, MAHOOTCHI M, MOSADEGH H & FALLAHI F, 2014, *A new approach for maintenance scheduling of generating units in electrical power systems based on their operational hours*, *Computers and Operations Research*, **50**, pp. 61–79.
- [82] FEDERAL ENERGY REGULATORY COMMISSION (FERC), 2005, *Economic dispatch: Concepts, practices and issues*, Presentation to the Joint Board for the Study of Economic Dispatch, Palm Springs (CA).
- [83] FETANAT A & SHAFIPOUR G, 2011, *Generation maintenance scheduling in power systems using ant colony optimization for continuous domains based 0–1 integer programming*, *Expert Systems with Applications*, **38(8)**, pp. 9729–9735.
- [84] FICO, 2014, *Optimization — Solver your biggest and most complex business problems*, [Online], [Cited October 1st, 2016], Available from <https://goo.gl/Fnef7Q>.
- [85] FINKELSTEIN M, 2008, *Failure rate modelling for reliability and risk*, 1st Edition, Springer, London.
- [86] FOONG WK, MAIER HR & SIMPSON AR, 2005, *Ant colony optimization for power plant maintenance scheduling optimization*, Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation, Washington (DC), pp. 249–256.
- [87] FOONG WK, SIMPSON AR, MAIER HR & STOLP S, 2008, *Ant colony optimization for power plant maintenance scheduling optimization — A five-station hydropower system*, *Annals of Operations Research*, **159(1)**, pp. 433–450.
- [88] FOURCADE F, JOHNSON E, BARA M & CORTEY-DUMONT P, 1997, *Optimizing nuclear power plant refueling with mixed-integer programming*, *European Journal of Operational Research*, **97(2)**, pp. 269–280.
- [89] FRIPP C, 2015, *Eskom: We have not maintained our power plants like we should have*, [Online], [Cited February 16th, 2015], Available from <http://goo.gl/c9QpBW>.
- [90] FROGER A, GENDREAU M, MENDOZA JE, PINSON É & ROUSSEAU L, 2016, *Maintenance scheduling in the electricity industry: A literature review*, *European Journal of Operational Research*, **251(3)**, pp. 695–706.
- [91] FRONTLINE SOLVERS, 2016, *Frontline solvers — Home*, [Online], [Cited October 1st, 2016], Available from <https://goo.gl/ZmFZF>.
- [92] FRONTLINE SOLVERS, 2016, *Standard Excel solver — Dealing with problem size limits*, [Online], [Cited October 1st, 2016], Available from <https://goo.gl/tvQamF>.
- [93] FU Y, SHAHIDEHPOUR M & LI Z, 2007, *Security-constrained optimal coordination of generation and transmission maintenance outage scheduling*, *IEEE Transactions on Power Systems*, **22(3)**, pp. 1302–1313.
- [94] GARVER LL, 1972, *Adjusting maintenance schedules to levelize risk*, *IEEE Transactions on Power Apparatus and Systems*, **(5)**, pp. 2057–2063.
- [95] GEETHA T & SWARUP KS, 2009, *Coordinated preventive maintenance scheduling of GENCO and TRANSCO in restructured power systems*, *International Journal of Electrical Power and Energy Systems*, **31(10)**, pp. 626–638.
- [96] GLOVER F & KOCHENBERGER GA, 2003, *Handbook of metaheuristics*, 1st Edition, Kluwer Academic Publishers, Boston (MA).
- [97] GLOVER F, 1986, *Future paths for integer programming and links to artificial intelligence*, *Computers and Operations Research*, **13(5)**, pp. 533–549.
- [98] GODSKESEN S, JENSEN TS, KJELDTSEN N & LARSEN R, 2013, *Solving a real-life, large-scale energy management problem*, *Journal of Scheduling*, **16(6)**, pp. 567–583.

- [99] GRIGG C, WONG P, ALBRECHT P, ALLAN R, BHAVARAJU M, BILLINTON R, KURUGANTY S, FONG C, HADDAD S & CHEN Q, 1999, *The IEEE reliability test system-1996. A report prepared by the reliability test system task force of the application of probability methods subcommittee*, IEEE Transactions on Power Systems, **14(3)**, pp. 1010–1020.
- [100] GUROBI OPTIMIZATION, 2016, *An optimization solver reimaged*, [Online], [Cited June 6th, 2015], Available from <http://www.gurobi.com/products/features-benefits>.
- [101] GUROBI OPTIMIZATION, 2016, *State of the art mathematical programming solver*, [Online], [Cited June 6th, 2015], Available from <http://www.gurobi.com/products/gurobi-optimizer>.
- [102] GUTHERY SB, 1974, *Partition regression*, Journal of the American Statistical Association, **69(348)**, pp. 945–947.
- [103] HLONGWANE S, 2012, *Hey Eskom, remember 2008?*, [Online], [Cited February 26th, 2015], Available from <http://www.dailymaverick.co.za/article/2012-01-12-hey-eskom-remember-2008/>.
- [104] HOBBS BF, 1995, *Optimization methods for electric utility resource planning*, European Journal of Operational Research, **83**, pp. 1–20.
- [105] HOLLAND JH, 1973, *Genetic algorithms and the optimal allocation of trials*, SIAM Journal on Computing, **2(2)**, pp. 88–105.
- [106] HU X, 2006, *Particle swarm optimization*, [Online], [Cited February 29th, 2016], Available from <http://www.swarmintelligence.org/>.
- [107] HUANG CJ, LIN CE & HUANG CL, 1992, *Fuzzy approach for generator maintenance scheduling*, Electric Power Systems Research, **24(1)**, pp. 31–38.
- [108] HUANG KY & YANG HT, 2002, *Effective algorithm for handling constraints in generator maintenance scheduling*, Generation, Transmission and Distribution, **149(3)**, pp. 274–282.
- [109] HUANG MD, ROMEO F & SANGIOVANNI-VINCENTELLI A, 1986, *An efficient general cooling schedule for simulated annealing*, Proceedings of the Annual Conference on Computer-aided Design (ICCAD-86), Santa Clara (CA), pp. 381–384.
- [110] HUANG SJ, 1998, *A genetic-evolved fuzzy system for maintenance scheduling of generating units*, International Journal of Electrical Power and Energy Systems, **20(3)**, pp. 191–195.
- [111] HUANG SJ, 1997, *Generator maintenance scheduling: A fuzzy system approach with genetic enhancement*, Electric Power Systems Research, **41(3)**, pp. 233–239.
- [112] IBM CORPORATION, 2016, *CPLEX optimizer*, [Online], [Cited May 19th, 2016], Available from <http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/>.
- [113] IBM CORPORATION, 2015, *CPLEX user's manual, Version 12, Release 6*, [Online], [Cited October 2nd, 2016], Available from http://www.ibm.com/support/knowledgecenter/api/content/nl/en-us/SSSA5P_12.6.3/ilog.odms.studio.help/Optimization_Studio/topics/PLUGINS_ROOT/ilog.odms.studio.help/pdf/usrcplex.pdf.
- [114] IBM CORPORATION, 2015, *Getting started with CPLEX, Version 12, Release 6*, [Online], [Cited October 2nd, 2016], Available from http://www.ibm.com/support/knowledgecenter/api/content/nl/en-us/SSSA5P_12.6.3/ilog.odms.studio.help/Optimization_Studio/topics/PLUGINS_ROOT/ilog.odms.studio.help/pdf/gscplex.pdf.
- [115] IBM CORPORATION, 2016, *IBM ILOG CPLEX Optimizer Studio*, [Online], [Cited May 19th, 2016], Available from <https://goo.gl/TOAMdD>.

- [116] IBM CORPORATION, 2017, *Mathematical programming vs. constraint programming*, [Online], [Cited October 18th, 2017], Available from <https://www-01.ibm.com/software/integration/optimization/cplex-cp-optimizer/mp-cp/>.
- [117] IBM CORPORATION, 2017, *Solving scheduling problems with constraint programming*, [Online], [Cited October 18th, 2017], Available from https://www.ibm.com/support/knowledgecenter/en/SSSA5P_12.3.0/ilog.odms.studio.help/Content/Optimization/Documentation/Optimization_Studio/_pubskel/ps_COS_Eclipse61.html.
- [118] IGNIZIO JP, 1982, *Linear programming in single- & multiple-objective systems*, Prentice Hall International, Englewood (NJ).
- [119] INTERNATIONAL UNION OF GEODESY AND GEOPHYSICS (IUGG), 2015, *List of developing countries*, [Online], [Cited October 2nd, 2016], Available from <https://goo.gl/iS0k4j>.
- [120] JACKSON DE & RATNIEKS FL, 2006, *Communication in ants*, *Current Biology*, **16(15)**, pp. R570–R574.
- [121] JOST V & SAVOUREY D, 2013, *A 0–1 integer linear programming approach to schedule outages of nuclear power plants*, *Journal of Scheduling*, **16(6)**, pp. 551–566.
- [122] KANASE-PATIL A, SAINI R & SHARMA M, 2010, *Integrated renewable energy systems for off grid rural electrification of remote area*, *Renewable Energy*, **35(6)**, pp. 1342–1349.
- [123] KENDALL KE & KENDALL JE, 2011, *Systems analysis and design*, Pearson, Upper Saddle River (NJ).
- [124] KENDALL MG, 1938, *A new measure of rank correlation*, *Biometrika*, **30**, pp. 81–93.
- [125] KESSIDES I, BOGETIC Z & MAURER L, 2007, *Current and forthcoming issues in the South African electricity sector*, World Bank Policy Research Working Paper No. 4197.
- [126] KIM DW, KIM KH, JANG W & CHEN FF, 2002, *Unrelated parallel machine scheduling with setup times using simulated annealing*, *Robotics and Computer-Integrated Manufacturing*, **18(3)**, pp. 223–231.
- [127] KIM H, HAYASHI Y & NARA K, 1997, *An algorithm for thermal unit maintenance scheduling through combined use of GA, SA and TS*, *IEEE Transactions on Power Systems*, **12(1)**, pp. 329–335.
- [128] KIM JH, PARK JB, PARK JK & CHUN YH, 2005, *Generating unit maintenance scheduling under competitive market environments*, *International Journal of Electrical Power and Energy Systems*, **27(3)**, pp. 189–194.
- [129] KIRKPATRICK S, GELATT CD & VECCHI MP, 1983, *Optimization by simulated annealing*, *Science*, **220(4598)**, pp. 671–680.
- [130] KNUTH DE, 1974, *Postscript about NP-hard problems*, *ACM SIGACT News*, **6(2)**, pp. 15–16.
- [131] KOAY CA & SRINIVASAN D, 2003, *Particle swarm optimization-based approach for generator maintenance scheduling*, *Proceedings of the 2003 IEEE Symposium on Swarm Intelligence (SIS '03)*, Indianapolis (IN), pp. 167–173.
- [132] KOVÁCS A, ERDŐS G, VIHAROS ZJ & MONOSTORI L, 2011, *A system for the detailed scheduling of wind farm maintenance*, *CIRP Annals — Manufacturing Technology*, **60(1)**, pp. 497–501.
- [133] KRALJ B & RAJAKOVIĆ N, 1994, *Multiobjective programming in power system optimization: New approach to generator maintenance scheduling*, *International Journal of Electrical Power and Energy Systems*, **16(4)**, pp. 211–220.

- [134] KRALJ BL & PETROVIĆ R, 1995, *A multiobjective optimization approach to thermal generating units maintenance scheduling*, European Journal of Operational Research, **84(2)**, pp. 481–493.
- [135] KRALJ BL & PETROVIĆ R, 1988, *Optimal preventive maintenance scheduling of thermal generating units in power systems — A survey of problem formulations and solution methods*, European Journal of Operational Research, **35(1)**, pp. 1–15.
- [136] KUO L & YANG TY, 1996, *Bayesian computation for nonhomogeneous Poisson processes in software reliability*, Journal of the American Statistical Association, **91(434)**, pp. 763–773.
- [137] KUZLE I, PANDŽIĆ H & BREZOVEC M, 2007, *Implementation of the Benders decomposition in hydro generating units maintenance scheduling*, Proceedings of the Hydro Conference 2007: New Approaches for a New Era, Granada, pp. 1–8.
- [138] LAND AH & DOIG AG, 1960, *An automatic method of solving discrete programming problems*, Econometrica: Journal of the Econometric Society, **28(3)**, pp. 497–520.
- [139] LAPORTE G, 1992, *The vehicle routing problem: An overview of exact and approximate algorithms*, European Journal of Operational Research, **59(3)**, pp. 345–358.
- [140] LEE KY & PARK JB, 2006, *Application of particle swarm optimization to economic dispatch problem: Advantages and disadvantages*, Proceedings of the 2006 IEEE Power Systems Conference and Exposition (PSCE '06), Atlanta (GA), pp. 188–192.
- [141] LEOU RC, 2006, *A new method for unit maintenance scheduling considering reliability and operation expense*, International Journal of Electrical Power and Energy Systems, **28(7)**, pp. 471–481.
- [142] LEWIS A, 2013, *Reliability*, [Online], [Cited January 22nd, 2015], Available from <http://cosmologist.info/teaching/STAT/CHAP7.pdf>.
- [143] LEWIS PAW & ROBINSON DW, 1973, *Testing for a monotone trend in a modulated renewal process*, (Unpublished) Technical Report NPS-55Lw73121, Naval Postgraduate School, Monterey (CA).
- [144] LIN CE, HUANG CJ, HUANG CL, LIANG CC & LEE SY, 1992, *An expert system for generator maintenance scheduling using operation index*, IEEE Transactions on Power Systems, **7(3)**, pp. 1141–1148.
- [145] LINDO SYSTEMS INC., 2016, *An overview of LINGO*, [Online], [Cited June 06th, 2016], Available from http://www.lindo.com/index.php?option=com_content&view=article&id=2&Itemid=10.
- [146] LINDSEY JC & RYAN LM, 1998, *Methods for interval-censored data*, Statistics in Medicine, **17(2)**, pp. 219–238.
- [147] LIU G & TOMSOVIC K, 2012, *Quantifying spinning reserve in systems with significant wind power penetration*, IEEE Transactions on Power Systems, **27(4)**, pp. 2385–2393.
- [148] LOOTSMA FA, 2013, *Fuzzy logic for planning and decision making*, 1st Edition, Springer Science & Business Media, Boston (MA).
- [149] LV C, WANG J, YOU S & ZHANG Z, 2015, *Short-term transmission maintenance scheduling based on the Benders decomposition*, International Transactions on Electrical Energy Systems, **25(4)**, pp. 697–712.
- [150] MALHOTRA MK, SHARMA S & NAIR SS, 1999, *Decision making using multiple models*, European Journal of Operational Research, **114(1)**, pp. 1–14.

- [151] MANDEL T, 1997, *The elements of user interface design*, John Wiley & Sons, New York (NY).
- [152] MARWALI MKC & SHAHIDEHPOUR SM, 1998, *Integrated generation and transmission maintenance scheduling with network constraints*, IEEE Transactions on Power Systems, **13(3)**, pp. 1063–1068.
- [153] MARWALI M & SHAHIDEHPOUR S, 1998, *A deterministic approach to generation and transmission maintenance scheduling with network constraints*, Electric Power Systems Research, **47(2)**, pp. 101–113.
- [154] MATHWORKS, 2016, *Optimization software — Optimization toolbox — MATLAB*, [Online], [Cited October 1st, 2016], Available from <https://goo.gl/S58v6g>.
- [155] MAZIDI P, TOHIDI Y, RAMOS A & SANZ-BOBI MA, 2017, *Profit-maximization generation maintenance scheduling through bi-level programming*, European Journal of Operational Research, In Press.
- [156] MCGILCHRIST C, 1995, *Recursive estimation and residuals*, Mathematical and Computer Modelling, **22(10-12)**, pp. 201–206.
- [157] MDUNGE B, 2016, Production Assurance Manager at Eskom, [Personal Communication], Contactable at mdungeba@eskom.co.za.
- [158] MICROSOFT, 2016, *Excel 2016 spreadsheet software*, [Online], [Cited October 1st, 2016], Available from <https://goo.gl/XDIqIR>.
- [159] MILLER RG, GONG G & MUÑOZ A, 2011, *Survival analysis*, 2nd Edition, John Wiley & Sons, New York (NY).
- [160] MIN C, KIM M, PARK J & YOON Y, 2013, *Game-theory-based generation maintenance scheduling in electricity markets*, Energy, **55**, pp. 310–318.
- [161] MOHANTA DK, SADHU PK & CHAKRABARTI R, 2004, *Fuzzy reliability evaluation of captive power plant maintenance scheduling incorporating uncertain forced outage rate and load representation*, Electric Power Systems Research, **72(1)**, pp. 73–84.
- [162] MOHANTA DK, SADHU PK & CHAKRABARTI R, 2007, *Deterministic and stochastic approach for safety and reliability optimization of captive power plant maintenance scheduling using GA/SA-based hybrid techniques: A comparison of results*, Reliability Engineering and System Safety, **92(2)**, pp. 187–199.
- [163] MOLLAHASSANI-POUR M, ABDOLLAHI A & RASHIDINEJAD M, 2014, *Application of a novel cost reduction index to preventive maintenance scheduling*, International Journal of Electrical Power and Energy Systems, **56**, pp. 235–240.
- [164] MOMOH JA, MA XW & TOMSOVIC K, 1995, *Overview and literature survey of fuzzy set theory in power systems*, IEEE Transactions on Power Systems, **10(3)**, pp. 1676–1690.
- [165] MONTGOMERY DC & RUNGER GC, 2010, *Applied statistics and probability for engineers*, 5th Edition, John Wiley & Sons, Inc., New York (NY).
- [166] MORO LM & RAMOS A, 1999, *Goal programming approach to maintenance scheduling of generating units in large scale power systems*, IEEE Transactions on Power Systems, **14(3)**, pp. 1021–1028.
- [167] MOUBRAY J, 2007, *Reliability-centered maintenance*, 1st Edition, Butterworth Heineemann, Oxford.
- [168] MROMLINSKI LR, 1985, *Transportation problem as a model for optimal schedule of maintenance outages in power systems*, International Journal of Electrical Power and Energy Systems, **7(3)**, pp. 161–164.

- [169] MUKERJI R, MERRILL HM, ERICKSON BW, PARKER JH & FRIEDMAN RE, 1991, *Power plant maintenance scheduling: Optimizing economics and reliability*, IEEE Transactions on Power Systems, **6(2)**, pp. 476–483.
- [170] MYTAKIDIS T & VLACHOS A, 2008, *Maintenance scheduling by using the bi-criterion algorithm of preferential anti-pheromone*, Leonardo Journal of Sciences, **12(16)**, pp. 143–164.
- [171] NATRELLA M, 2013, *NIST/SEMATECH e-handbook of statistical methods*, [Online], [Cited March 2nd, 2015], Available from <http://goo.gl/8E1Js>.
- [172] NEGNEVITSKY M & KELAREVA G, 1999, *Genetic algorithms for maintenance scheduling in power systems*, Proceedings of the Australasian Universities Power Engineering Conference and IEAust Electric Energy Conference, Darwin, pp. 184–189.
- [173] O’CONNOR PDT & KLEYNER P, 2012, *Practical reliability engineering*, 1st Edition, John Wiley & Sons, Chichester.
- [174] PYTHON.ORG, 2016, *Welcome to Python.org*, [Online], [Cited October 1st, 2016], Available from <https://goo.gl/WxZLS2>.
- [175] R FOUNDATION, 2016, *The R project for statistical computing*, [Online], [Cited May 19th, 2016], Available from <https://www.r-project.org/>.
- [176] RARDIN RL, 1998, *Optimization in operations research*, 1st Edition, Prentice Hall, Upper Saddle River (NJ).
- [177] RELIASOFT, 2015, *Accelerated life testing reference*, [Online], [Cited March 17th, 2016], Available from <http://goo.gl/4HTjB6>.
- [178] RELIASOFT, 2012, *Confidence bounds*, [Online], [Cited May 7th, 2015], Available from http://reliawiki.org/index.php/Confidence_Bounds#Fisher_Matrix_Confidence_Bounds.
- [179] RELIASOFT, 2014, *Hypothesis test*, [Online], [Cited April 23rd, 2015], Available from http://reliawiki.org/index.php/Hypothesis_Tests.
- [180] RELIASOFT, 2014, *Life data analysis reference*, [Online], [Cited March 4th, 2015], Available from http://www.synthesisplatform.net/references/Life_Data_Analysis_Reference.pdf.
- [181] RELIASOFT, 2014, *System analysis reference: Reliability, availability & optimization*, [Online], [Cited March 3rd, 2015], Available from <http://goo.gl/1kS8LB>.
- [182] RELIASOFT, 2012, *Using the power law model for data analysis in RGA*, [Online], [Cited July 8th, 2015], Available from <http://www.weibull.com/hotwire/issue131/relbasics131.htm>.
- [183] ROSSI F, VAN BEEK P & WALSH T, 2006, *Handbook of constraint programming*, 1st Edition, Elsevier, Amsterdam.
- [184] RSTUDIO, 2016, *RStudio*, [Online], [Cited May 19th, 2016], Available from <https://www.rstudio.com/>.
- [185] SARAIVA JT, PEREIRA ML, MENDES VT & SOUSA JC, 2011, *A simulated annealing based approach to solve the generator maintenance scheduling problem*, Electric Power Systems Research, **81(7)**, pp. 1283–1291.
- [186] SARAVANAN B, DAS S, SIKRI S & KOTHARI D, 2013, *A solution to the unit commitment problem — A review*, Frontiers in Energy, **7(2)**, pp. 223–236.

- [187] SATOH T & NARA K, 1991, *Maintenance scheduling by using simulated annealing method [for power plants]*, IEEE Transactions on Power Systems, **6(2)**, pp. 850–857.
- [188] SCHLÜNZ EB, 2011, *Decision support for generator maintenance scheduling in the energy sector*, MSc Thesis, Stellenbosch University, Stellenbosch.
- [189] SCHLÜNZ EB & VAN VUUREN JH, 2013, *An investigation into the effectiveness of simulated annealing as a solution approach for the generator maintenance scheduling problem*, International Journal of Electrical Power and Energy Systems, **53**, pp. 166–174.
- [190] SCHLÜNZ EB, BOKOV PM & VAN VUUREN JH, 2014, *Research reactor in-core fuel management optimisation using the multiobjective cross-entropy method*, Proceedings of the 2014 International Conference on Reactor Physics (PHYSOR 2014), Kyoto.
- [191] SELVI V & UMARANI DR, 2010, *Comparative analysis of ant colony and particle swarm optimization techniques*, International Journal of Computer Applications, **5(4)**, pp. 1–6.
- [192] SEN S & KOTHARI D, 1998, *Optimal thermal generating unit commitment: A review*, International Journal of Electrical Power & Energy Systems, **20(7)**, pp. 443–451.
- [193] SILVA EL, MOROZOWSKI M, FONSECA LGS, OLIVEIRA GC, MELO ACG & MELLO J, 1995, *Transmission constrained maintenance scheduling of generating units: A stochastic programming approach*, IEEE Transactions on Power Systems, **10(2)**, pp. 695–701.
- [194] STAIR R & REYNOLDS G, 2012, *Statistical learning theory*, Cengage Learning, Boston (MA).
- [195] STEYNBERG R, 2016, *A framework for identifying the most likely successful underprivileged tertiary bursary applicants*, MEng Thesis, Stellenbosch University, Stellenbosch.
- [196] STONE D, JARRETT C, WOODROFFE M & MINOCHA S, 2005, *User interface design and evaluation*, Morgan Kaufmann Publishers, San Francisco (CA).
- [197] SUMAN B & KUMAR P, 2006, *A survey of simulated annealing as a tool for single and multiobjective optimization*, Journal of the Operational Research Society, **57(10)**, pp. 1143–1160.
- [198] SURESH K & KUMARAPPAN N, 2013, *Hybrid improved binary particle swarm optimization approach for generation maintenance scheduling problem*, Swarm and Evolutionary Computation, **9**, pp. 69–89.
- [199] TALBI E, 2009, *Metaheuristics: From design to implementation*, 1st Edition, John Wiley & Sons, Hoboken (NJ).
- [200] TAŞKIN ZC, 1998, *Benders decomposition*, Wiley Encyclopaedia of Operations Research and Management Sciences, John Wiley & Sons, Malden (MA).
- [201] TAVAKKOLI-MOGHADDAM R, JAVADIAN N, JAVADI B & SAFAEI N, 2007, *Design of a facility layout problem in cellular manufacturing systems with stochastic demands*, Applied Mathematics and Computation, **184(2)**, pp. 721–728.
- [202] TECHOPEDIA, 2016, *Job scheduling*, 16th, 2016], Available from <https://www.techopedia.com/definition/7882/job-scheduling>.
- [203] THE NEW YORK TIMES COMPANY, 2017, *The New York Times*, [Online], [Cited May 12th, 2017], Available from <https://www.nytimes.com/>.
- [204] TOBIAS PA & TRINDADE D, 2011, *Applied reliability*, 2nd Edition, CRC Press, New York (NY).
- [205] TRIKI E, COLLETTE Y & SIARRY P, 2005, *A theoretical study on the behavior of simulated annealing leading to a new cooling schedule*, European Journal of Operational Research, **166(1)**, pp. 77–92.

- [206] TSAI WY, JEWELL NP & WANG MC, 1987, *A note on the product-limit estimator under right censoring and left truncation*, *Biometrika*, **74(4)**, pp. 883–886.
- [207] TSENG C, LI C & OREN S, 2000, *Solving the unit commitment problem by a unit decommitment method*, *Journal of Optimization Theory and Applications*, **105(3)**, pp. 707–730.
- [208] TWITTER INC., 2017, *Twitter*, [Online], [Cited May 12th, 2017], Available from <https://twitter.com/>.
- [209] UNITED STATES DEPARTMENT OF DEFENCE, 1990, *Reliability prediction of electronic equipment*, (Unpublished) Technical Report US MIL-HDBK-217F, Washington (DC).
- [210] VAN LEEUWEN J, 1991, *Handbook of theoretical computer science: Algorithms and complexity*, 1st Edition, Elsevier, Amsterdam.
- [211] VAN LAARHOVEN PJ & AARTS EH, 1987, *Simulated annealing: Theory and applications*, Springer Science & Business Media, Dordrecht.
- [212] VAPNIK VN & VAPNIK V, 1998, *Statistical learning theory*, Wiley, New York (NY).
- [213] VOLKANOVSKI A, MAVKO B, BOŠEVSKI T, ČAUŠEVSKI A & ČEPIN M, 2008, *Genetic algorithm optimisation of the maintenance scheduling of generating units in a power system*, *Reliability Engineering and System Safety*, **93(6)**, pp. 779–789.
- [214] WANG X & McDONALD JR, 1994, *Modern power system planning*, 1st Edition, McGraw-Hill, London.
- [215] WANG Y & HANDSCHIN E, 2000, *A new genetic algorithm for preventive unit maintenance scheduling of power systems*, *International Journal of Electrical Power and Energy Systems*, **22(5)**, pp. 343–348.
- [216] WATT A & ENG N, 2014, *Database design*, Pressbooks, Montreal.
- [217] WHITMORE D, 2013, *Heuristics*, [Online], [Cited July 28th, 2015], Available from <http://www.managers-net.com/heuristics.html>.
- [218] WINSTON WL, 2004, *Operations research: Applications and algorithms*, 4th Edition, Brooks/Cole–Thomson Learning, Belmont (CA).
- [219] YAMAYEE Z, SIDENBLAD K & YOSHIMURA M, 1983, *A computationally efficient optimal maintenance scheduling method*, *IEEE Transactions on Power Apparatus and Systems*, **PAS-102(2)**, pp. 330–338.
- [220] YARE Y & VENAYAGAMOORTHY GK, 2010, *Optimal maintenance scheduling of generators using multiple swarms-MDPSO framework*, *Engineering Applications of Artificial Intelligence*, **23(6)**, pp. 895–910.
- [221] YELLEN J, AL-KHAMIS T, VEMURI S & LEMONIDIS L, 1992, *A decomposition approach to unit maintenance scheduling*, *IEEE Transactions on Power Systems*, **7(2)**, pp. 726–733.
- [222] ZADEH LA, 1965, *Fuzzy sets*, *Information and Control*, **8(3)**, pp. 338–353.
- [223] ZEILEIS A, LEISCH F, HORNIK K, KLEIBER C, HANSEN B & MERKLE E, 2015, *Testing, monitoring, and dating structural changes, (R Package)*, [Online], [Cited March 10th, 2017], Available from <https://cran.r-project.org/web/packages/strucchange/strucchange.pdf>.
- [224] ZHAN JP, GUO CX, WU QH, ZHANG LL & FU HJ, 2014, *Generation maintenance scheduling based on multiple objectives and their relationship analysis*, *Journal of Zhejiang University SCIENCE C*, **15(11)**, pp. 1035–1047.

-
- [225] ZHAO Y, VOLOVOI V, WATERS M & MAVRIS D, 2006, *A sequential approach for gas turbine power plant preventative maintenance scheduling*, Journal of Engineering for Gas Turbines and Power, **128(4)**, pp. 796–805.
- [226] ZURN HH & QUINTANA VH, 1975, *Generator maintenance scheduling via successive approximations dynamic programming*, IEEE Transactions on Power Apparatus and Systems, **94(2)**, pp. 665–671.
- [227] ZURN HH & QUINTANA VP, 1977, *Several objective criteria for generator preventive maintenance scheduling*, IEEE Transactions on Power Apparatus and Systems, **PAS-96(3)**, pp. 984–992.

APPENDIX A

Eskom case study: Parameter optimisation experimental results

This appendix serves the purpose of documenting the results obtained during the SA parameter optimisation experiments carried out in §9 within the context of the 157-unit Eskom case study.

A.1 Minimising the probability of unit failure

This section contains the results of the parameter optimisation experiments performed for the minimisation of the probability of unit failure scheduling criterion within the context of the 157-unit Eskom Case study in §9.

A.1.1 Initial acceptance ratio and soft constraint violation severity factor

The mean objective function values associated with the feasible incumbents returned during the first phase of the parameter optimisation experiment are shown in Table A.1.

TABLE A.1: Mean objective function values for all parameter value combinations during the first phase of the parameter optimisation experiment involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ for the 157-unit Eskom case study with minimisation of the probability of unit failure as scheduling criterion.

χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	321.588	321.621	321.596	321.599	321.574
0.5	321.538	321.594	321.562	321.687	321.639
0.6	321.582	321.620	321.600	321.556	321.636
0.7	321.535	321.587	321.556	321.565	321.636
0.8	321.712	321.766	321.790	321.801	321.583

The mean computation times involved in evaluating combinations of these parameter values are shown in Table A.2. These table entries include computation times expended during runs that returned infeasible incumbents.

The numbers of times (out of 30) that an infeasible incumbent was returned during an SA search run are finally shown in Table A.3.

288 APPENDIX A. ESKOM CASE STUDY: PARAMETER OPTIMISATION EXPERIMENTAL RESULTS

TABLE A.2: Mean computation times required (in seconds) for all parameter value combinations during the first phase of the parameter optimisation experiment involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ for the 157-unit Eskom case study with minimisation of the probability of unit failure as scheduling criterion.

χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	43 200	43 200	43 200	43 200	43 200
0.5	43 200	43 200	43 200	43 200	43 200
0.6	43 200	43 200	43 200	43 200	43 200
0.7	43 200	43 200	43 200	43 200	43 200
0.8	43 200	43 200	43 200	43 200	43 200

TABLE A.3: Number of infeasible incumbents (out of 30) returned for all parameter value combinations during the first phase of the parameter optimisation experiment involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ for the 157-unit Eskom case study with minimisation of the probability of unit failure as scheduling criterion.

χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	14 (46.67%)	1 (3.33%)	0 (0.00%)	0 (0.00%)	0 (0.00%)
0.5	12 (40.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)
0.6	8 (26.67%)	3 (10.00%)	0 (0.00%)	0 (0.00%)	0 (0.00%)
0.7	11 (36.67%)	1 (3.33%)	1 (3.33%)	0 (0.00%)	0 (0.00%)
0.8	14 (46.67%)	3 (10.00%)	1 (3.33%)	0 (0.00%)	0 (0.00%)

A.1.2 Cooling parameter, reheating parameter and epoch parameter

The mean objective function values for the feasible incumbents returned during the second phase of the parameter optimisation experiment are shown in Table A.4.

TABLE A.4: Mean objective function values for all the combinations of parameter values during the second phase of the parameter optimisation experiment involving variation of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ for the 157-unit Eskom case study with minimisation of the probability of unit failure as scheduling criterion.

		ψ	ξ			
			0.55	0.75	0.95	
α	0.85	1	322.443	322.561	322.087	
		2	321.774	321.519	321.843	
		4	321.712	321.533	322.361	
		0.9	1	325.195	325.549	325.968
			2	321.844	321.764	321.749
			4	321.682	321.665	322.129
	0.95	1	326.336	326.183	327.586	
		2	326.117	325.556	327.242	
		4	322.028	321.909	321.844	

The mean computation times required for the evaluation of the combinations of these parameter values are shown in Table A.5. These values again include computation times expended during runs that returned infeasible incumbents.

TABLE A.5: Mean computation times required (in seconds) for all the combinations of parameter values during the second phase of the parameter optimisation experiment involving variation of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ for the 157-unit Eskom case study with minimisation of the probability of unit failure as scheduling criterion.

		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	43 200	43 200	43 200
		2	43 200	43 200	25 510
		4	43 200	43 200	7 972
	0.9	1	43 200	43 200	43 200
		2	43 200	43 200	41 897
		4	43 200	43 200	13 718
	0.95	1	43 200	43 200	43 200
		2	43 200	43 200	43 200
		4	43 200	43 200	39 980

The numbers of times (out of 30) that an infeasible incumbent was returned during an SA search run are shown in Table A.6.

TABLE A.6: Number of infeasible incumbents (out of 30) returned for all the combinations of parameter values during the second phase of the parameter optimisation experiment involving variations of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ for the 157-unit Eskom case study with minimisation of the probability of unit failure as scheduling criterion.

		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	4 (13.33%)	4 (13.33%)	5 (16.67%)
		2	6 (20.00%)	6 (20.00%)	6 (20.00%)
		4	5 (16.67%)	5 (16.67%)	5 (16.67%)
	0.9	1	6 (20.00%)	9 (30.00%)	10 (33.33%)
		2	6 (20.00%)	3 (10.00%)	8 (26.67%)
		4	4 (13.33%)	1 (3.33%)	4 (13.33%)
	0.95	1	9 (30.00%)	7 (23.33%)	13 (43.33%)
		2	8 (26.67%)	5 (16.67%)	5 (16.67%)
		4	8 (26.67%)	8 (26.67%)	5 (16.67%)

A.2 Maximising expected energy production

This section contains the results of the parameter optimisation experiments performed for the maximisation of the expected energy production scheduling criterion within the context of the 157-unit Eskom Case study in §9.

A.2.1 Initial acceptance ratio and soft constraint violation severity factor

The mean objective function values associated with the feasible incumbents returned during the first phase of the parameter optimisation experiment are shown in Table A.7.

The mean computation times involved in evaluating combinations of these parameter values are shown in Table A.8. These table entries include computation times expended during runs that returned infeasible incumbents.

290 APPENDIX A. ESKOM CASE STUDY: PARAMETER OPTIMISATION EXPERIMENTAL RESULTS

TABLE A.7: Mean objective function values for all parameter value combinations during the first phase of the parameter optimisation experiment involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ for the 157-unit Eskom case study with maximisation of expected energy production as scheduling criterion.

χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	19 474 630	19 461 907	19 456 327	19 447 782	19 391 103
0.5	19 451 300	19 426 781	19 417 565	19 399 915	19 416 100
0.6	19 459 663	19 447 087	19 434 016	19 423 503	19 417 576
0.7	19 463 779	19 443 069	19 431 552	19 388 152	19 384 670
0.8	19 441 080	19 420 379	19 402 569	19 422 396	19 394 736

TABLE A.8: Mean computation times required (in seconds) for all parameter value combinations during the first phase of the parameter optimisation experiment involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ for the 157-unit Eskom case study with maximisation of expected energy production as scheduling criterion.

χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	43 200	43 200	43 200	43 200	43 200
0.5	43 200	43 200	43 200	43 200	43 200
0.6	43 200	43 200	43 200	43 200	43 200
0.7	43 200	43 200	43 200	43 200	43 200
0.8	43 200	43 200	43 200	43 200	43 200

The numbers of times (out of 30) that an infeasible incumbent was returned during an SA search run are shown in Table A.9.

TABLE A.9: Number of infeasible incumbents (out of 30) returned for all parameter value combinations during the first phase of the parameter optimisation experiment involving the initial acceptance ratio χ_0 and the soft constraint violation severity factor γ for the 157-unit Eskom case study with maximisation of expected energy production as scheduling criterion.

χ_0	γ				
	0.25	0.5	0.75	1	1.25
0.4	2 (6.67%)	0 (0.00%)	1 (3.33%)	1 (3.33%)	2 (6.67%)
0.5	5 (16.67%)	4 (13.33%)	4 (13.33%)	2 (6.67%)	0 (0.00%)
0.6	6 (20.00%)	0 (0.00%)	4 (13.33%)	1 (3.33%)	1 (3.33%)
0.7	5 (16.67%)	1 (3.33%)	2 (6.67%)	1 (3.33%)	4 (13.33%)
0.8	9 (30.00%)	6 (20.00%)	2 (6.67%)	1 (3.33%)	0 (0.00%)

A.2.2 Cooling parameter, reheating parameter and epoch parameter

The mean objective function values for the feasible incumbents returned during the second phase of the parameter optimisation experiment are shown in Table A.10.

The mean computation times required for the evaluation of the combinations of these parameter values are shown in Table A.11. These table entries again include computation times expended during runs that returned infeasible incumbents.

The numbers of times (out of 30) that an infeasible incumbent was returned during an SA search run are shown in Table A.12.

TABLE A.10: Mean objective function values for all the combinations of parameter values during the second phase of the parameter optimisation experiment involving variation of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ for the 157-unit Eskom case study with maximisation of expected energy production as scheduling criterion.

		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	19 440 684	19 442 239	19 430 188
		2	19 481 211	19 481 895	19 482 346
		4	19 480 743	19 475 981	19 462 576
	0.9	1	19 326 448	19 285 738	19 335 538
		2	19 465 054	19 458 809	19 479 233
		4	19 480 083	19 475 712	19 473 333
	0.95	1	18 760 866	18 828 574	18 855 865
		2	19 300 199	19 301 215	19 280 346
		4	19 464 763	19 470 082	19 469 139

TABLE A.11: Mean computation times required (in seconds) for all the combinations of parameter values during the second phase of the parameter optimisation experiment involving variation of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ for the 157-unit Eskom case study with maximisation of expected energy production as scheduling criterion.

		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	43 200	43 200	43 200
		2	43 200	43 200	38 784
		4	43 200	43 200	15 631
	0.9	1	43 200	43 200	43 200
		2	43 200	43 200	43 200
		4	43 200	43 200	24 475
	0.95	1	43 200	43 200	43 200
		2	43 200	43 200	43 200
		4	43 200	43 200	43 200

TABLE A.12: Number of infeasible incumbents (out of 30) returned for all the combinations of parameter values during the second phase of the parameter optimisation experiment involving variations of the cooling parameter α , the reheating parameter ξ and the epoch parameter ψ for the 157-unit Eskom case study with maximisation of expected energy production as scheduling criterion.

		ψ	ξ		
			0.55	0.75	0.95
α	0.85	1	4 (13.33 %)	1 (3.33 %)	6 (20.00 %)
		2	1 (3.33 %)	0 (0.00 %)	3 (10.00 %)
		4	4 (13.33 %)	3 (10.00 %)	2 (6.67 %)
	0.9	1	13 (43.33 %)	10 (33.33 %)	10 (33.33 %)
		2	2 (6.67 %)	2 (6.67 %)	3 (10.00 %)
		4	2 (6.67 %)	1 (3.33 %)	3 (10.00 %)
	0.95	1	24 (80.00 %)	27 (90.00 %)	25 (83.33 %)
		2	5 (16.67 %)	8 (26.67 %)	10 (33.33 %)
		4	4 (13.33 %)	0 (0.00 %)	2 (6.67 %)

292 APPENDIX A. ESKOM CASE STUDY: PARAMETER OPTIMISATION EXPERIMENTAL RESULTS

APPENDIX B

Contents of the accompanying disc

The compact disc included in this dissertation contains the DSS described in §11 as an **R** file. The latest versions of **R** and **Rstudio** are required to run the DSS. The disc also contains the required specifications (the demand specification and PGU specifications) for both the IEEE-RTS benchmark system and the 157-unit Eskom case study. These files are in the correct input format as specified in Figures 11.2 and 11.3. Empty templates for both the demand specification and PGU specifications are also included on the disc which may be copied and completed for ease of use of the proposed DSS. An electronic copy of the dissertation (in **.pdf** format) is finally also included on the disc.