

A framework for optimising real-estate development incentivisation in priority areas

by
Martinus Loots



Thesis presented in partial fulfilment of the requirements for the degree of
Master of (Industrial) Engineering
in the Faculty of Engineering at Stellenbosch University

Supervisor: Prof JH van Vuuren
Co-supervisor: Quintin van Heerden

March 2021

Declaration

By submitting this thesis 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: November 19, 2020

Abstract

South African cities are experiencing rapid growth as the country becomes more urbanised and people search for a better quality of life. This rapid population growth has exacerbated the urban spatial contrasts that South Africa has been grappling with for centuries. The historical spatial separation experienced in South Africa has been reinforced due to the most marginalised in society settling on the peripheries of cities where the delivery of municipal services and quality of life is a stark contrast to those of more central locations in these cities. These problems have prompted the South African government to create an urban vision of pursuing more inclusive, integrated and compact cities in the future. For the modern South African city to align with this vision, spatial transformation of the urban environment must take place.

Urban spatial transformation may be encouraged by enacting municipal plans and frameworks. A strategy for encouraging spatial transformation involves the incentivisation of real-estate development in certain strategically located land areas within the boundaries of a city. Such development allows for densification of the population in specific areas where municipal services may be more efficiently employed, allowing for more sustainable growth of cities. Municipalities often employ tailored urban policies which encourage densification in strategic areas by making available subsidies and grants for this purpose.

A novel generic framework is proposed in this thesis for optimising the implementation of urban policies aimed at incentivising the development of residential real-estate in strategically prioritised areas. The framework comprises of two generic components which facilitate the iterative optimisation of potential urban policy implementation. The components of the framework employ statistical methods to predict possible future urban compositions and employ a meta-heuristic approach to optimise the anticipated future consequences of strategically implementing these urban policies. The framework is intended as a decision support tool for policy makers and city planners in aid of developing incentivisation urban densification policies.

The framework is instantiated on a computer and implemented virtually as a proof of concept in a real-world case study. This case study is based on data for the City of Ekurhuleni, located in the South African province of Gauteng. It is demonstrated in the case study how the framework is capable of achieving significantly more efficient urban densification policies than would occur naturally.

Opsomming

Suid-Afrikaanse stede ervaar snelle groei namate die land meer verstedelik en mense na 'n beter lewenskwaliteit soek. Hierdie vinnige bevolkingsaanwas het die ruimtelike kontraste in stede waarmee Suid-Afrika al vir eeue worstel, vererger. Die historiese ruimtelike skeiding wat in Suid-Afrika ervaar word, is versterk as gevolg van die mees gemarginaliseerdes in die samelewing wat hulle in die buitewyke van stede vestig, waar die lewering van munisipale dienste en lewenskwaliteit 'n skrilte kontras is met dié van meer sentrale liggings in hierdie stede. Hierdie probleme het die Suid-Afrikaanse regering aangespoor om 'n stedelike visie daar te stel om meer inklusiewe, geïntegreerde en kompakte stede in die toekoms na te streef. Om die moderne Suid-Afrikaanse stad met hierdie visie te bely, moet ruimtelike transformasie van die stedelike omgewing plaasvind.

Stedelike ruimtelike transformasie kan aangemoedig word deur munisipale planne en raamwerke daar te stel. 'n Strategie vir die aanmoediging van ruimtelike transformasie behels die aansporing van die ontwikkeling van vaste eiendom in sekere strategiese geleë gebiede binne die grense van 'n stad. Sodanige ontwikkeling maak voorsiening vir verdigting van die bevolking in spesifieke gebiede waar munisipale dienste meer doeltreffend gelewer kan word, wat meer volhoubare groei in stede moontlik maak. Munisipaliteite gebruik dikwels pasgemaakte stedelike beleide wat verdigting in strategiese gebiede aanmoedig deur subsidies en toelaes hiervoor beskikbaar te stel.

'n Nuwe generiese raamwerk word in hierdie tesis voorgestel vir die optimering van stedelike beleidsimplementering wat daarop gemik is om die ontwikkeling van residensiële vaste eiendom in strategiese prioriteitsareas aan te spoor. Die raamwerk bestaan uit twee generiese komponente wat die iteratiewe optimering van potensiële stedelike beleidsimplementering vergemaklik. Die komponente van die raamwerk gebruik statistiese metodes om moontlike toekomstige stedelike samestellings te voorspel en volg 'n metaheuristiese benadering om die verwagte toekomstige gevolge van die strategiese implementering van hierdie stedelike beleide te optimeer. Die raamwerk is as 'n besluitnemingshulpmiddel vir beleidsmakers en stadsbeplanners bedoel om die bevordering van aansporingsbeleide vir stedelike verdigting te bewerkstellig.

Die raamwerk word rekenaarmatig geïnstansieer en virtueel as 'n bewys van konsep in 'n werklike gevallestudie geïmplementeer. Hierdie gevallestudie is gebaseer op data vir die Stad Ekurhuleni, wat in die Suid-Afrikaanse provinsie Gauteng geleë is. In die gevallestudie word aangetoon hoe die raamwerk daartoe in staat is om aansienlik meer doeltreffende aansporingsbeleide vir stedelike verdigting te bewerkstellig as wat natuurlik sou voorkom.

Acknowledgements

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

- My supervisor, Prof JH van Vuuren, for his support and guidance. It was an absolute pleasure working with him and I have been blessed to have learnt so much from him, both professionally and personally. He is truly a great role model.
- My co-supervisor, Mr Q van Heerden from the CSIR, for his enthusiastic assistance and sharing of knowledge. He provided me with much needed technical support, exhibiting grace and patience.
- The *Stellenbosch Unit for Operations Research and Engineering* (SUnORE) and the Industrial Engineering department at Stellenbosch University for creating an inductive environment for producing high quality work whilst constantly striving for excellence.
- My friends and colleagues at SUnORE for the support, friendship, assistance and feeling of camaraderie throughout the past three years.
- My friend and thesis buddy, Pierre Cilliers, for his relentless support and assistance throughout my studies and for always being there when I needed him.
- My friends and family for their constant support and love, especially my father André and brother Tiaan.
- A special thanks to my late mother, Salomé, for her constant support and love and major contribution toward shaping the person I am today. All I have achieved is thanks to the remarkable role model she was and numerous lessons I learnt from her.

Table of Contents

Abstract	iii
Opsomming	v
Acknowledgements	vii
List of Acronyms	xiii
List of Figures	xv
List of Tables	xvii
1 Introduction	1
1.1 Background	1
1.2 Informal problem description	6
1.3 Objectives	6
1.4 Thesis scope	7
1.5 Research methodology	7
1.6 Thesis organisation	8
2 Mathematical prerequisites	9
2.1 Coefficient estimation	10
2.1.1 Maximum likelihood estimation	10
2.1.2 Newton-Raphson method	11
2.1.3 Probability distributions	11
2.2 Monte Carlo simulation	14
2.3 Combinatorial optimisation problem solution methodologies	15
2.3.1 Exact solution methodologies	16
2.3.2 Heuristic solution methodologies	17
2.3.3 Metaheuristic solution methodologies	18

2.4	The genetic algorithm	20
2.4.1	Initial population generation	20
2.4.2	Selection strategies	22
2.4.3	Crossover strategies	24
2.4.4	Mutation strategies	25
2.4.5	Replacement strategies	26
2.4.6	Stopping criteria	27
2.5	Chapter summary	27
3	Spatial planning tools	29
3.1	Urban simulation	29
3.1.1	Steps in a typical urban simulation study	30
3.1.2	Validation and verification of a typical urban simulation study	31
3.2	Integrated transport and land use models	32
3.2.1	The Integrated Transportation and Land Use Package	32
3.2.2	MEPLAN	33
3.2.3	Modelo de Uso de Suelo de SANTIAGO	33
3.3	The UrbanSim simulation software suite	34
3.3.1	UrbanSim design considerations	35
3.3.2	UrbanSim software architecture	37
3.3.3	The UrbanSim model structure	38
3.3.4	Location choice models	43
3.3.5	UrbanSim validation	50
3.4	Chapter summary	50
4	Urban policy scenario optimisation framework	51
4.1	Framework overview	52
4.2	The preprocessing component	53
4.2.1	Scenario area identification	54
4.2.2	Coefficient estimation	56
4.3	Scenario creation and analysis	59
4.3.1	Scenario creation	59
4.3.2	Scenario analysis	62
4.3.3	Fitness scoring	64
4.4	The optimisation component	64
4.4.1	Population initialisation	64

Table of Contents	xi
4.4.2 Selection	65
4.4.3 Crossover	66
4.4.4 Mutation	66
4.4.5 Replacement	67
4.4.6 Stopping criteria	67
4.5 Results produced by the framework	67
4.6 Chapter summary	69
5 Case study	71
5.1 Background	71
5.1.1 City of Ekurhuleni	72
5.1.2 Case study dataset	73
5.2 Case study preprocessing	74
5.2.1 Case study scenario areas	75
5.2.2 Case study coefficient estimation	75
5.3 Policy scenario construction	79
5.3.1 The policy scenario plan	80
5.3.2 Policy scenario implementation	80
5.3.3 Policy scenario analysis	81
5.4 Results and discussion	83
5.5 Chapter summary	87
6 Conclusion	89
6.1 Thesis summary	89
6.2 Appraisal of thesis contributions	90
6.3 Suggestions for future work	92
References	95

List of Acronyms

- ACO** Any colony optimisation
- CDF** Cumulative distribution function
- CSIR** Council for Scientific and Industrial Research
- DFD** Data flow diagram
- DRAM** Disaggregate Residential Allocation Model
- EMPAL** Employment Allocation Model
- GA** Genetic algorithm
- GIS** Geographic information system
- ITLUM** Integrated transport and land use model
- ITLUP** Integrated Transportation and Land Use Package
- MLE** Maximum likelihood estimation
- MPO** Metropolitan planning organisation
- MUSSA** Modelo de Uso de Suelo de SAntiago
- PDF** Probability density function
- PMF** Probability mass function
- PSO** Particle swarm optimisation
- RUM** Random utility maximisation
- SA** Simulated annealing
- SSI** Simple sequential inhibition
- TS** Tabu search

List of Figures

1.1	Early European settlement in South Africa	2
1.2	Apartheid in South Africa	3
1.3	Main mode of travel to work in South Africa (2003–2013)	4
2.1	Two examples of Poisson distribution PMFs	12
2.2	Two examples of binomial distribution PMFs	13
2.3	An example of the multinomial distribution PMF	13
2.4	Two examples of gumbel distribution PDFs	14
2.5	Monte Carlo sampling for a discrete distribution	15
2.6	Monte Carlo sampling for a continuous distribution	16
2.7	Working of a typical GA implementation	21
2.8	Illustration of the role of a neighbourhood in the SSI procedure	22
2.9	Solution space partitioning according to parallel diversification initialisation	23
2.10	An illustration of the tournament selection strategy in a GA	24
2.11	1-Point crossover and 2-point crossover in a GA	25
2.12	Uniform crossover in a GA	25
2.13	An illustration of integer swap mutation in a GA	26
2.14	A variation on the strategy of elitism replacement in a GA	27
3.1	Elements of urban modelling in UrbanSim	35
3.2	Graphical representation of the UrbanSim architecture	38
3.3	UrbanSim software data flow	39
3.4	UrbanSim location choice model specification	44
3.5	UrbanSim location choice model computational process	44
4.1	High-level DFD overview of the UPSOM framework architecture	53
4.2	The preprocessing component of the UPSOM framework	54
4.3	The process of selecting scenario areas in the UPSOM framework	55

4.4	The nature of estimation and simulation data in the UPSOM framework	56
4.5	A visual representation of a policy scenario plan	60
4.6	A visual representation of policy implementation	61
4.7	A visual presentation of how each policy scenario is analysed	62
4.8	A tabular representation of optimal zones output by the UPSUM framework . . .	68
4.9	A visual representation of optimal zones output by the UPSUM framework . . .	68
5.1	Geographic priority areas in the City of Ekurhuleni	73
5.2	Potential scenario areas for real-estate price or grant provision policies	75
5.3	Potential scenario areas for job accessibility policies	76
5.4	Policy scenario plan for the Ekurhuleni case study	81
5.5	Policy scenarios exhibiting variation in policy element concentration	82
5.6	UPSOM framework incumbent fitness for $b = 2\,000$	83
5.7	UPSOM framework incumbent fitness for $b = 3\,000$	84
5.8	Assignment of incumbent solution policy elements for $b = 2\,000$	84
5.9	Assignment of incumbent solution policy elements for $b = 3\,000$	85
5.10	Policy scenario with variation in concentration for $b = 2\,000$	85
5.11	Policy scenario with variation in concentration for $b = 3\,000$	86

List of Tables

2.1	Evaluation of GA initial population generation strategies	21
5.1	Attractiveness of locations for real-estate development of type 201	78
5.2	Attractiveness of locations for real-estate development of type 202	79
5.3	Summary of potential policies that may be implemented in the case study	80
5.4	UPSOM framework improvements over random policies for $b = 2\,000$	87
5.5	UPSOM framework improvements over random policies for $b = 3\,000$	87

CHAPTER 1

Introduction

Contents

1.1	Background	1
1.2	Informal problem description	6
1.3	Objectives	6
1.4	Thesis scope	7
1.5	Research methodology	7
1.6	Thesis organisation	8

Following twenty-seven years of imprisonment, Nelson Rolihlahla Mandela walked out of prison as a free man on 11 February 1990 [3]. This initiated a major wave of change that swept over South Africa, providing hope to millions of people who suffered under the yoke of oppression [3]. Mandela’s release soon became the cause for celebration as peace and justice were expected to return to a country torn apart by violence and oppression during the Apartheid regime. The political change set in motion by Mandela’s release brew a certain optimism among the South African people that future generations would have a better life [6]. The footprint left on the new-found free South African society by the oppressive Apartheid regime was, however, preceded by years of racial tension dating as far back as the arrival of the first European settlers in the Cape peninsula.

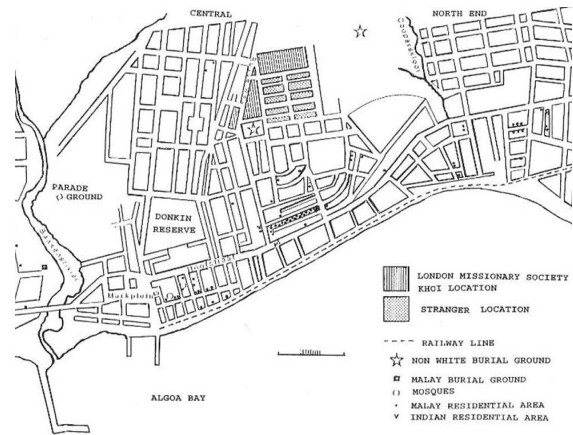
1.1 Background

It all started when the Dutch vessel, Haerlem, wrecked in Table Bay in 1647. Upon returning to the Netherlands, its survivors produced glowing reports of the region which piqued the interest of the *Vereenigde Oost-Indische Compagnie* (VOC)¹. Following these reports, the company sent its representative, Jan van Riebeeck, to establish a station there tasked with resupplying ships sailing past the Cape of Good Hope. Soon after Van Riebeeck set foot on South African shores on April 6, 1652, as depicted in Figure Figure 1.1(a), the VOC selected sites for the creation of a fort and a vegetable garden, marking the birth of the first South African city, Cape Town. As

¹The VOC was a trading company created in the modern-day Netherlands in 1602. The reason for the creation of the company was to facilitate Dutch interests in trade within nations surrounding the Indian Ocean, as well as to assist the Dutch in their quest for independence from the Spanish at the time [22].



(a) Painting by Charles Bell (1813–1882) of Jan van Riebeeck arriving in Table Bay in April 1652 [71]



(b) Port Elizabeth's Native Strangers Location during the 1850s [54]

FIGURE 1.1: Early European settlement in South Africa.

the VOC released men² from employment to settle and farm the land from 1657 onwards, the small station started to grow. Moreover, as slaves were imported, political exiles introduced, and marriage and cohabitation with the native Khoikhoi occurred, the population increased and the station turned into a small town around the southern-most tip of Africa [5].

Soon after the fort and the surrounding settlements had been established at the Cape, patterns of exclusion started to emerge as the settlers erected boundaries between them and the indigenous people of the Cape. One of the first ways in which barriers were erected was when Van Riebeeck instructed his men to plant bitter almond trees, various brambles and thorn bushes fulfilling the role of deterrents along farm boundaries, including on his own farm in Wynberg. This boundary was meant to deter the local Khoikhoi from attacking the settlers and to prevent their cattle from wandering onto the settlers' land³ [72]. By 1676, the Khoikhoi herders were completely repelled from the lands immediately surrounding the early settlement area [71]. The Dutch then continued to systematically lay physical and imagined boundaries separating specific areas in which Khoikhoi were allowed to build houses and let their cattle graze from others in which they were forbidden to do so. By the end of the 1700s, the Khoikhoi people were devastated by an outbreak of smallpox and the loss of grazing pastures due to annexations by colonial farmers and *vrye burghers*.

The boundaries that were created to exclude the Khoikhoi laid the foundation for what would later become a segregated town, marking the beginning of the segregated city model that would much later be implemented all across the country [71]. Within this early Cape society, spatial demarcation based on the position that an individual held in society became increasingly prevalent [71]. One instance of this was when the VOC's slaves were allocated specific slave lodges whilst military and administrative personnel were accommodated within the castle [37]. This was followed by the introduction of laws aimed at regulating the lifestyle of slaves and the Khoikhoi, marking the first instance in which legislation played a role in creating the later segregated cities of South Africa [71].

²These men were granted land in the Cape of Good Hope by the VOC and were called "vrye burghers." The first *vrye burgher* settled at the Cape on 21 February 1657, marking the beginning of settlers making South Africa their permanent residence [9].

³"The belt will be so densely overgrown that it will be impossible for cattle and sheep to be driven through and it will take the form of a protective fence..." wrote Jan van Riebeeck in his diary on 23–25 February 1660 [93].



(a) A sign used during the Apartheid era to establish an area for white people only [43]



(b) Sophiatown during the Group Areas Act [4]

FIGURE 1.2: Apartheid in South Africa.

As the Cape Colony's population grew under the command of the British during the period 1806–1910 [56], the birth and development of other major cities took place, following the establishment of Cape Town. Many of these cities have their own early segregated past. During the 1850s, for example, Port Elizabeth created a so-called *Native Strangers Location* where non-Europeans were expected to live, as shown in Figure 1.1(b), if they visited the city, resulting in non-Europeans very rarely entering urban areas. In the Colony of Natal, the city of Durban was more focussed on the Indian population⁴ and in 1871 the City Council created specific areas for Indians to move to [71]. The Boer republics of *Suid-Afrikaanse Republiek* and *Oranje Vrystaat* also adopted certain segregationist policies. In Johannesburg during the 1890s, the government demarcated separate African and Malay living locations, whilst in the Free State town councils also implemented segregation laws in its towns [53].

The aforementioned segregation created during the colonial era would be further exacerbated by the oppression system, known as Apartheid, imposed on non-Europeans by the South African government from 1948 onward [38]. This system was based on partitioning groups of people on the basis of race. Certain areas were earmarked specifically for use by certain racial groups, as depicted in Figure 1.2(a). This system saw the mass removal of 3.5 million black South Africans to these demarcated areas during the period of 1948–1983, resulting in one the largest mass removal attempts in modern history [71]. Large-scale removals were initiated during the 1950s and 1960s when the Group Areas Act was enacted [71]. This act called for the removal of black Africans, Indians and other people of colour from areas occupied by people of European origin throughout the country, further segregating cities. The government did this at a time of mass resistance against the Apartheid regime. Two of the most famous areas to be destroyed by the government were Sophiatown in Johannesburg, shown in Figure 1.2(b), and District Six in Cape Town. These were two vibrant multi-cultural, multi-racial communities that were disenfranchised and subsequently declared as white areas [61].

The legacy of Apartheid and of colonialism continues to play an integral role in the current make-up of South African cities. Following the end of restrictive apartheid laws forbidding

⁴The first Indian people arrived in Durban on 16 November 1860 to work as indentured labourers on the Natal sugar cane plantations. They arrived from India, another British colony at the time, to work in the Colony of Natal [94].

the freedom of movement of non-Europeans, South African cities have grown at an alarming pace [45]. This growth is the result of a combination of natural and inbound-migration from rural areas and neighbouring states [45]. A major proportion of the urban population growth has occurred in townships and informal settlements that had already been established in the segregated past. In 2014, South Africa faced the unique challenge that 38% of the working-age urban population resides in these townships and informal settlements whilst also accounting for 60% of its unemployed population [51]. This is because these areas are the “first recipients of rural migrants searching for work” [45]. These areas have mostly been established on the peripheries of cities [45]. The aforementioned population growth has therefore resulted in the poor continuing to be marginalised on the peripheries of South African society. These townships on city outskirts often have very poor access to employment areas and public amenities as the latter are often focussed in and around city centres. This results in commuter transport being a major expense incurred by the poorest of the South African workforce. Data from the 2013 *National Travel Surveys* (NTS) show that commuters using taxis, buses and trains spent, on average, more than 15% of their gross income on their commuter travelling expenses [41]. As can be seen in Figure 1.3, a large proportion of the population travelled in this manner. Not only do poor South Africans lose large proportions of their expendable income instantly due to incurring significant transportation costs, they also endure long travel times, which means that they lose many potential working hours per day [45]. Furthermore, access to good schooling from townships and informal settlements is also restricted, further perpetuating the poverty cycle as the next generation cannot readily receive a high-quality education [74]. These inequality issues have prompted the South African government to create an urban vision of pursuing more inclusive, integrated and compact cities in the future [45]. Since the announcement that

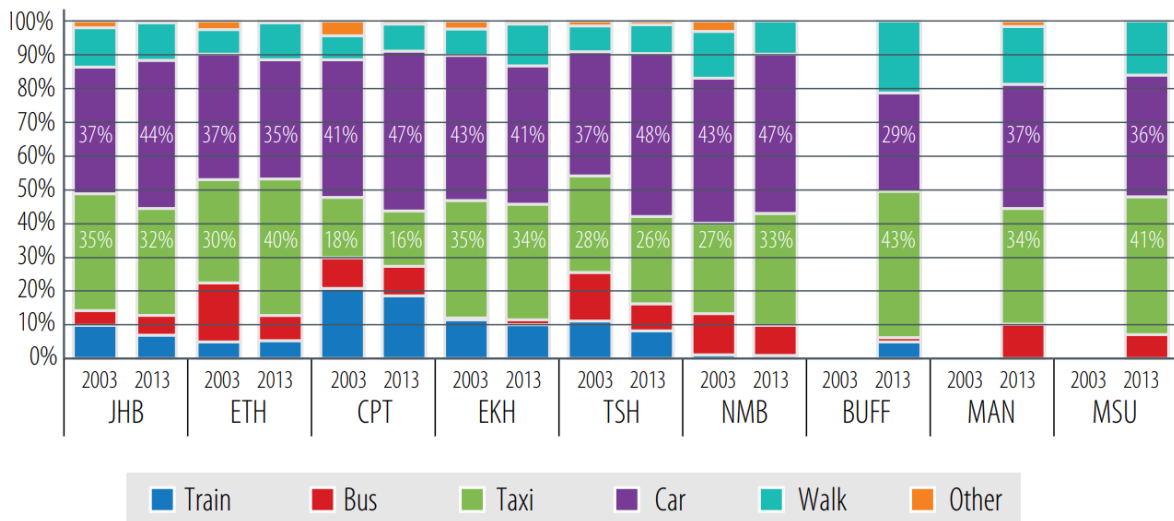


FIGURE 1.3: Main modes of travel to work in South Africa (2003–2013) [45].

Apartheid would be abolished during the early 1990s, the government has, in fact, set out plans to create more inclusive, integrated and compact cities. The first stage of this plan was launched pre-1994 and was known as the *Government Building Policy* which resulted in the establishment of housing rights for the previously marginalised portion of the population [45]. The next phase emerged during the period 1994–2003 when the focus shifted to building houses. During this time, the government focussed on building affordable, subsidised houses to alleviate major housing shortages within cities and encouraged banks to extend bonds to low-income persons so that they could buy houses [45]. During the period 2004–2014 the focus shifted again — this time to building and extending human settlements. During this time, the

goal was to develop sustainable human settlements and improve spatial integration and housing assets [45]. Informal settlements were either eradicated or upgraded and accredited. The *South African Cities Network* (SACN) then adopted a strategy from 2014 onwards to take a long-term perspective. The main theme for the future of the spatial transformation of South African cities has shifted to spatial integration, sustainability, efficiency and balance by the creation of integrated urban settlements [45]. The current state of South African cities, accompanied by the increasing complexity associated with the design of cities globally, has presented a major challenge to city planners in respect of expanding and creating cities in the most effective and efficient manner possible.

As urban technologies, infrastructure, economies, political and social structures, and norms and transportation have evolved, urban systems have become more complex. Moreover, as resources around the world have become scarcer, more emphasis has been placed on efficiency. The urban environment has become more interconnected which has resulted in a situation where it has become both irresponsible and infeasible to deal with choices related to major land area usage and transportation infrastructure as isolated choices to be made by planners and bureaucrats [89]. For a long time theoretical and mathematical models, such as the monocentric model of a city, were used to decode the complexities involved in city planning and to allow policy makers to acquire a clear and concise view of developments of cities in order to facilitate the aforementioned important decisions [1]. These models are important tools enabling a fundamental understanding of the underlying principles of urban development. Despite this, much of the current understanding of city expansion is based on oversimplified assumptions underlying these models. Agencies and policy makers require more accurate information to facilitate their decision making and this calls for more complex models of urban development [89].

In order to address this planning and policy challenge, computerised models which emulate urbanised land use and commuter transportation were initially created during the 1960s in the United States [89]. The first such model was called the *Urban Transportation Planning System* and was used to forecast travel demand and the locations of jobs and households within a city [92]. From there on the requirements for models of urban expansion have become increasingly complex, requiring a diverse range of land use and transportation models. This has led to the creation of more advanced urban planning models such as *UrbanSim* [32] during the 1990s, aimed at addressing these more complex planning requirements [89]. UrbanSim is an urban simulation model capable of simulating the spatial make-up and growth within city areas over time. This simulation tool is able to accommodate a variety of policy scenarios aimed at addressing the impact they would have on city development over time. The results returned by this simulation may be used by city planners to justify the use of certain policies [32].

For any South African city to achieve its goal of a more integrated, sustainable, efficient and balanced spatial make-up in view of its spatially divisive past, remedial policies have to be introduced. One possible approach is to encourage economic development and growth in key areas within an urban make-up. These areas are known as *priority areas*. Such areas have been identified by the South African government as areas that, if sufficiently developed, are expected to benefit cities from a spatial layout perspective [12]. For economic growth to occur in an area, real-estate is a critical factor in facilitating the building of residential homes and locations where jobs may be created. A particular challenge faced by policy makers is to make it fundamentally feasible and financially attractive for real-estate developers to develop within these prioritised areas at the lowest possible cost to the government. With the availability of new state-of-the-art urban simulation software, such as that described above, a significant opportunity arises in respect of assisting the planning of development in South African cities in the best manner possible by promoting optimised development within these prioritised areas.

1.2 Informal problem description

The research aim of this thesis is to design a framework for the optimisation of incentivisation strategies, in the form of urban policies, aimed at encouraging real-estate developers to focus their residential development efforts in prioritised urban areas. These urban areas have been identified as areas in which densification of residential real-estate development should benefit urban integration within the region, thereby contributing to spatial development planning by creating more integrated, sustainable, efficient and balanced urban areas. The optimisation of these strategies is aimed at maximising the beneficial effects of resource expenditure during their implementation. The incentivisation strategies and their implementation should be feasible in terms of the resources available and should be flexible in the sense that they should be capable of accommodating the heterogeneous nature of different types of residential real-estate developments and what the developers of these real-estate development types may deem attractive. The framework should be generic in nature so that it is applicable to a variety of urban areas.

1.3 Objectives

The following seven objectives are pursued in this thesis:

- I To *conduct* a thorough survey of the relevant literature:
 - (a) To *review* mathematical techniques and statistical distributions employed to quantify the attractiveness of spatial features and simulate decision making within a stochastic model environment,
 - (b) To *analyse* and *identify* combinatorial optimisation techniques for incentivisation implementation in spatial urban planning models when various trade-offs and large solution spaces are considered,
 - (c) To *research* and *compare* urban simulation models such as UrbanSim, and
 - (d) To *report* on location choice models and variations of these models as part of urban simulation models.
- II To *conceptualise* a generic framework that may be used to optimise the implementation of strategies in the form of urban policies employed in industry to encourage real-estate developers to develop residential real-estate in the areas prioritised for densification within an urban and regional planning space.
- III To *design* the components in the framework in Objective II by employing relevant techniques researched from the literature as stated in Objective I. More specifically:
 - (a) To *develop* appropriate procedures for a preprocessing component capable of accurately assessing spatial features within a specified urban area, assist an optimisation component aimed at optimally implementing urban policies and mitigate the computational challenges involved,
 - (b) To *establish* a suitable course of action with respect to the design of urban policies employed to encourage real-estate development in priority areas, the implementation of the policies designed and the evaluation of the quality of the implementation of the policies designed, and
 - (c) To *devise* an appropriate combinatorial optimisation mechanism capable of producing high-quality implementation strategies for the urban policies designed, which is compatible with the UrbanSim simulation software suite.

- IV To *optimise* and *apply* an urban policy implementation in Objective II within the framework designed in pursuit of Objective III in the context of a relevant real-world case study.
- V To *interpret* the results obtained in pursuit of Objective IV and compare them with unoptimised implementation of urban policies in order to motivate the need for decision support regarding the implementation of urban planning policies.
- VI To *verify* and validate the optimised implementation of the policies in Objective V.
- VII To *suggest* sensible follow-up work and improvements that may stem from the work documented in this thesis.

1.4 Thesis scope

Due to the intricacies of urban development and the complexities associated with modelling growth in urban areas, this thesis is restricted by the following scope assumptions and limitations:

Spatial features. The number of spatial features not altered when a policy scenario is simulated, is assumed to remain unchanged over a simulation year. No additional spatial features are, therefore, added or removed during the execution of the model framework.

Policy element assignment. The policy elements assigned, such as the number of jobs created or the distance of highway roads newly built, when a policy is implemented, is assumed to be assignable during a single year.

Control totals. The numbers of residential real-estate developments per year within specific regions are assumed from control totals in the literature.

1.5 Research methodology

The approach adopted during the research conducted in this thesis is partitioned into four stages. The first stage comprises a thorough review of the literature relevant to this thesis. An overview of relevant mathematical prerequisites required to facilitate an understanding of the work presented in this thesis is initially performed in the spirit of self-containment, as far as this is possible. This stage of the research stands in fulfilment of Objective I(a) of §1.3 as the components of the urban simulation model and the optimisation framework employed in this thesis are based on the prerequisite mathematical notions reviewed. The literature on methods for solving combinatorial optimisation problems is also reviewed briefly in fulfilment of Objective I(b). The purpose of this review is to identify optimisation techniques that may be applied within the optimisation component of the framework presented in this thesis. The literature review then turns toward research on urban spatial planning tools, such as UrbanSim, in fulfilment of Objective I(c). This part of the review is aimed at developing a fundamental understanding of spatial planning tools and models prior to the optimisation of the constituent elements of these models. The final part of the review includes a description of location choice models embedded within the UrbanSim environment, in fulfilment of Objective I(d). A fundamental understanding of these models is required as specific versions of these models are implemented within the framework presented in this thesis.

The second stage of this research is devoted to the conceptualisation of a model optimisation framework that may be used to streamline the implementation of incentivisation strategies in

the form of urban policies, as described in the preceding sections. This framework should facilitate an encouragement of residential real-estate development in priority areas as opposed to competing areas within the same region. The framework conceptualisation is carried out in pursuit of Objective II. A detailed design of the framework follows thereafter, starting with the establishment of adequate data preprocessing procedures. This is followed by proposing a mechanism for urban policy design, implementation and evaluation. The framework finally includes a component responsible for the optimisation of urban policy development. The aforementioned framework components are designed in accordance with Objectives III(a)–(c) in an attempt to assist policy makers and city planners in the form of policy implementation decision support.

The third stage involves the computer-implementation of an instantiation of the generic framework conceptualised and designed during the previous stage and applying this instantiation to optimise urban policy development in the context of a real-world case study in pursuit of Objective IV. During this stage a spatial plan is employed according to which the densification of priority areas should take place in order to assist with integration within an urban environment and the practicability of the framework is demonstrated in the context of the case study. Thereafter, the optimised policy development is interpreted, compared with unoptimised policy implementations, and the framework instantiation is verified and validated, in fulfilment of Objectives V and VI, respectively.

Finally during the fourth stage, sensible follow-up work on, and potential improvements to, the work documented in this thesis is suggested, in fulfilment of Objective VII.

1.6 Thesis organisation

This thesis consists of the current introductory chapter followed by five subsequent chapters. Chapters 2 and 3 are dedicated to a review of the literature on relevant techniques, tools and prerequisite concepts applicable to the topic of this thesis. Chapter 2, in particular, contains a discussion on a number of mathematical prerequisites, including coefficient estimation, Monte Carlo simulation and combinatorial optimisation solution methodologies. The focus of the literature review shifts towards urban simulation modelling and its evolution over time, in Chapter 3. This chapter includes thorough reviews of the UrbanSim software suite in particular, as well as the location choice models embedded within UrbanSim. Understanding these components is essential as they directly form part of the model framework proposed in this thesis.

Following on the literature review, Chapter 4 contains an introduction to and thorough description of the generic framework for optimising incentivisation strategies within an urban environment planning context put forward in this thesis. Pragmatic design considerations for the preprocessing and optimisation components of the framework are described in detail, as is the procedure employed for the design, implementation and analysis of urban incentivisation strategies. The framework design described in this chapter draws from knowledge acquired in Chapters 2 and 3.

A computer-implemented instantiation of the generic framework is applied to a real-world case study in Chapter 5. A detailed description of the case study is provided, and this is followed by a discussion on how the framework was applied and the critical reasoning behind decisions made during the application of the framework in the case study.

The thesis then closes in Chapter 6. This chapter is the conclusion of the thesis and contains a summary and appraisal of the contributions made, as well as various recommendations for future work which may follow on the contributions of this thesis.

CHAPTER 2

Mathematical prerequisites

Contents

2.1	Coefficient estimation	10
	2.1.1 <i>Maximum likelihood estimation</i>	10
	2.1.2 <i>Newton-Raphson method</i>	11
	2.1.3 <i>Probability distributions</i>	11
2.2	Monte Carlo simulation	14
2.3	Combinatorial optimisation problem solution methodologies	15
	2.3.1 <i>Exact solution methodologies</i>	16
	2.3.2 <i>Heuristic solution methodologies</i>	17
	2.3.3 <i>Metaheuristic solution methodologies</i>	18
2.4	The genetic algorithm	20
	2.4.1 <i>Initial population generation</i>	20
	2.4.2 <i>Selection strategies</i>	22
	2.4.3 <i>Crossover strategies</i>	24
	2.4.4 <i>Mutation strategies</i>	25
	2.4.5 <i>Replacement strategies</i>	26
	2.4.6 <i>Stopping criteria</i>	27
2.5	Chapter summary	27

A framework for the optimisation of complex statistical models is proposed in this thesis. A number of mathematical preliminary concepts are, however, required in order to gain an understanding of the framework components. These preliminaries are described in this chapter, which opens with a brief review of coefficient estimation with a particular focus on the method of *maximum likelihood estimation* and the accompanying *Newton-Raphson* method. A discussion follows thereafter on a number of noteworthy probability distributions, after which a discussion is provided on the *Monte Carlo simulation* sampling technique. The discussion then turns to solution methodologies for combinatorial optimisation problems, touching on exact solution methodologies, heuristic methodologies, and (both trajectory-based and population-based) metaheuristic methodologies. A more detailed discussion follows on the genetic algorithm, the population-based metaheuristic applied in this thesis. The chapter finally closes with a brief summary of its contents.

2.1 Coefficient estimation

Coefficients are often required to measure characteristics of a data set, process or phenomenon, on given certain conditions. Regression coefficients, in particular, are employed to estimate the unknown parameters of a population and usually describe the relationship between predictor variables (or independent variables) and their effect (the dependent variable) [26]. In regression, these coefficients are real values that are multiplied by the independent variables. A coefficient value is a representation of the mean change in the dependent variable, given a single unit increase of the corresponding independent variable. Numerous ways for estimating these coefficients are documented in the literature. Perhaps the best-known technique among these is the method of *maximum likelihood estimation* (MLE).

2.1.1 Maximum likelihood estimation

Predictive statistical models typically attempt to forecast results by emulating observed data. This is usually achieved by assessing the data and assigning an adequate probability distribution to the data set which best describes the observed data [10]. Probability distribution functions describe each possible value of a random variable together with its corresponding probability of occurrence. There are two types of probability distribution functions, namely *probability mass functions* (PMFs) for discrete random variables and *probability density functions* (PDFs) for continuous random variables. MLE is a method for computing the parameter values of probability distribution functions [10]. These parameter values are computed to maximise the likelihood that the resulting distribution function produces the data observed. MLE therefore provides a statistical framework for assessing information available among data by maximising the joint PDF or joint PMF, better known as the likelihood function [21]. This function is expressed mathematically as

$$L(\theta) = \prod_{i=1}^N f(y_i | \theta) \quad (2.1)$$

in the case where there are N independent data points, denoted by y_1, \dots, y_N . The set of relevant coefficients is denoted by θ . The MLE process may be computationally expensive, and so the alternative of opting to maximising the natural logarithm of the likelihood function (2.1) is usually adopted instead of maximising the likelihood function itself. This alternative is a valid equivalent because the natural logarithm of the likelihood function is an increasing function of the likelihood function, and is computationally cheaper, because it avoids the costly evaluation of products in favour of evaluating less computationally expensive summations. The log likelihood function is expressed mathematically as

$$\ell(\theta) = \ln \left[\prod_{i=1}^N f(y_i | \theta) \right] = \sum_{i=1}^N \ln[f(y_i | \theta)]. \quad (2.2)$$

A theoretical statistical distribution is assumed when the MLE is employed. The PDF or PMF of the assumed distribution is inserted into the function $f(y_i | \theta)$ with the result that the log likelihood function may be computed for the observed data points y_1, \dots, y_N and different sets of parameter values θ .

In order to find values for the parameters that maximise the log likelihood function, the method of MLE involves computing the partial derivatives of (2.2) with respect to all the parameters in θ . Local maxima of the log likelihood function are obtained by equating these derivatives to zero (rendering the hypersurface gradients zero), and solving the resulting set of equations.

Thereafter, the solutions thus obtained must be verified to yield maxima as opposed to minima of the log likelihood function before they are accepted. This is achieved by computing the second-order derivative of (2.2) with respect to some parameter of interest. This value determines whether the log likelihood function is concave up or down around the solution. If the second-order derivative is negative, it means the log likelihood function is concave down and the solution is accepted.

2.1.2 Newton-Raphson method

When employing the MLE method for functions which have many parameters, the complexity of the method increases. This is due the log likelihood function containing numerous variables. When the function derivatives are set equal to zero, there might not be an analytical method for solving the system in order to obtain values for all the parameters [70]. A numerical approach, called the Newton-Raphson method, may be employed in such cases.

The Newton-Raphson method is an efficient method for approximating the roots of a function, based on linear approximation. For a function of one variable, the method initiates with a random approximation of what a root value might be, denoted by x_0 [90]. The function value $f(x_0)$ at the initial approximation is then calculated, as is the corresponding gradient of the function $f'(x_0)$ at that point. The function point and corresponding gradient are then employed to calculate a subsequent root approximation x_1 , and this process is repeated, yielding a sequence of root approximations, until convergence occurs (*i.e.* the difference between two successive root approximations is smaller than a predetermined tolerance). The $(n+1)$ -st value in this sequence is given by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \quad n = 0, 1, 2, \dots \quad (2.3)$$

Upon thus approximating the first root, the entire process is repeated by starting at a new random approximation, until all the roots of the function have been calculated.

Although the above description of the Newton-Raphson method was for a function of one variable, the method is easily generalisable to multivariate functions, where the notion of a function derivative is replaced by its Jacobian matrix.

2.1.3 Probability distributions

There are a number of important theoretical statistical distributions in the literature, each with its own PMF or PDF. A number of distributions relevant to the topic of this thesis are reviewed in this section.

The *Poisson distribution* governs a discrete random variable typically used to enumerate events for which the probability of occurrence of each within a definite time or space is small [66]. The PMF of the Poisson distribution is

$$f(x) = \frac{\lambda^x e^{-\lambda}}{x!}, \quad (2.4)$$

where x denotes the number of occurrences of the event, e denotes Euler's number (the base of the natural logarithm) and λ is a parameter representing the expected value of the a discrete random variable. The shape of the distribution is illustrated in Figure 2.1 for two values of the parameter λ .

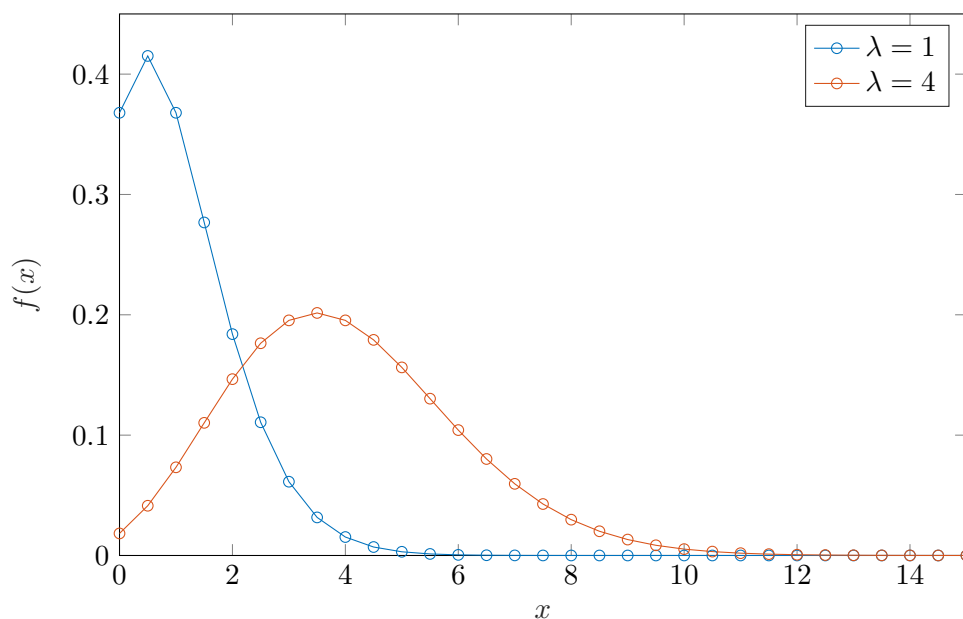


FIGURE 2.1: Two examples of the Poisson distribution PMF.

The *binomial distribution*, on the other hand, governs a discrete random variable used to enumerate the number of occurrences of each of two states among the outcomes of a number of trials. These states are typically called a *success* and a *failure* [8]. The PMF of a binomial distribution is

$$f(x) = \frac{n!}{x!(n-x)!} p^x (1-p)^{n-x}, \quad (2.5)$$

where x denotes the number of occurrences of successes, n is a parameter denoting the number of trials and $p \in [0, 1]$ is another parameter denoting the probability of observing a single success, independently of the outcomes of the other trials. The shape of the distribution is illustrated in Figure 2.2 for two sets of values of the parameters n and p .

The binomial distribution is a special case of the more general *multinomial distribution* which is used to enumerate the outcomes of trials involving more than two states [25]. The PMF of the multinomial distribution is

$$f(x_1, \dots, x_k) = \begin{cases} \frac{n!}{\prod_{i=1}^k x_i!} \prod_{i=1}^k p_i^{x_i} & \text{when } \sum_{i=1}^k x_i = n, \\ 0 & \text{otherwise,} \end{cases} \quad (2.6)$$

where x_i denotes the number of occurrences of state i , k denotes the number of states and n is a parameter denoting the number of trials. The probability of state i occurring (independently of the outcomes of the other trials) is the parameter $p_i \in [0, 1]$. It should hold that $\sum_{i=1}^k p_i = 1$.

An example of a multinomial PMF is illustrated graphically in Figure 2.3. In this illustration, there are three states, enumerated by x_1 , x_2 and x_3 , respectively.

The gumbel distribution is a well-known asymmetric, continuous distribution typically employed to model the maximum or minimum of a number of samples of various distributions [27]. The PDF for a gumbel distribution is

$$f(x) = \frac{1}{\beta} e^{(-\frac{x-\mu}{\beta})} e^{-e^{(-\frac{x-\mu}{\beta})}}, \quad (2.7)$$

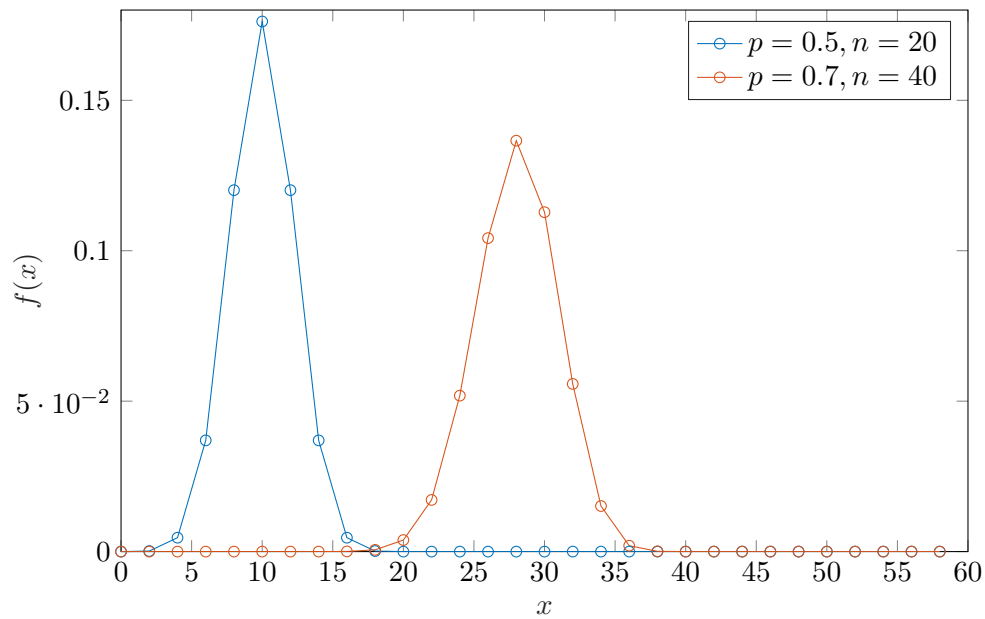


FIGURE 2.2: Two examples of binomial distribution PMFs.

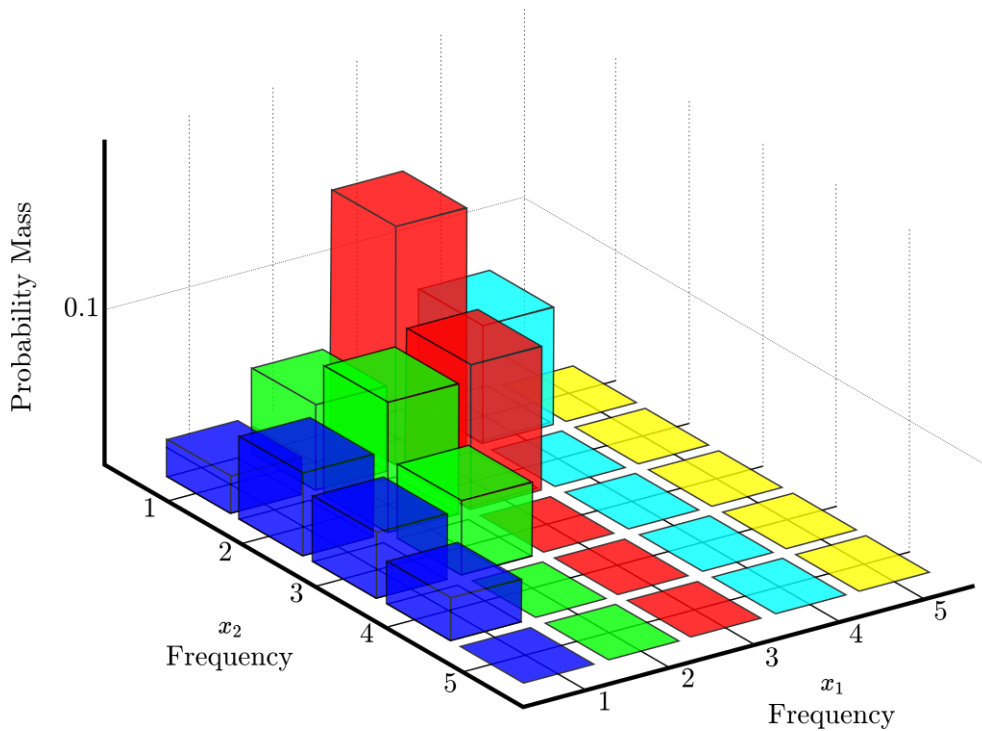


FIGURE 2.3: An example of the multinomial distribution PMF where the parameters of the distribution are $n = 5$, $k = 8$, $p_1 = 0.5$, $p_2 = 0.33$ and $p_3 = 0.16$. The graph only illustrates the PMF for the random variables x_1 and x_2 , whilst $x_3 = n - x_1 - x_2$.

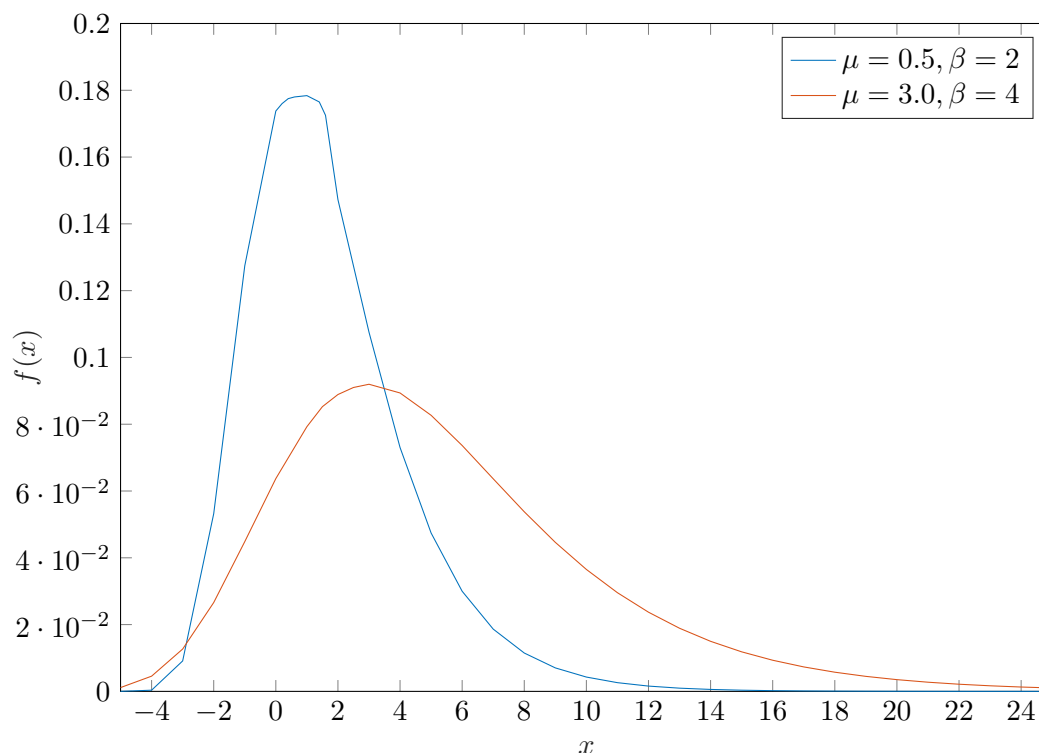


FIGURE 2.4: Two examples of the gumbel distribution PDF.

where μ is a so-called *location parameter* and β is a so-called *scale parameter*, both for a random real variable x . The shape of the gumbel distribution is illustrated graphically in Figure 2.4 for two sets of values of the parameters μ and β .

2.2 Monte Carlo simulation

Statistical models often require a sampling process in order to select a subset from a large population according to some specified distribution. The literature mentions numerous sampling techniques capable of achieving this. One such sampling technique is *Monte Carlo simulation*. This technique employs repeated random sampling to compute numerical estimations according to a probability distribution [63]. Sampling by means of this technique allows for a wide range of possible outcomes whilst considering the probabilities of the various outcomes being estimated [2]. There are numerous variations on the theme of Monte Carlo simulation for solving deterministic or probabilistic problems through a process of random number generation. This section is devoted to a brief overview of direct sampling by means of Monte Carlo simulation.

As stated above, Monte Carlo simulation requires a probability distribution according to which the sampling process is performed. Similarly to MLE in §2.1.1, Monte Carlo simulation may be applied to either discrete or continuous distribution functions. For direct sampling by means of Monte Carlo simulation, the *cumulative distribution function* (CDF) of the probability distribution is required [82]. The CDF of a discrete probability function, for which the probabilities of outcomes are known, may be expressed mathematically as

$$F(x) = \sum_{x_i \leq x} p(x_i), \quad (2.8)$$

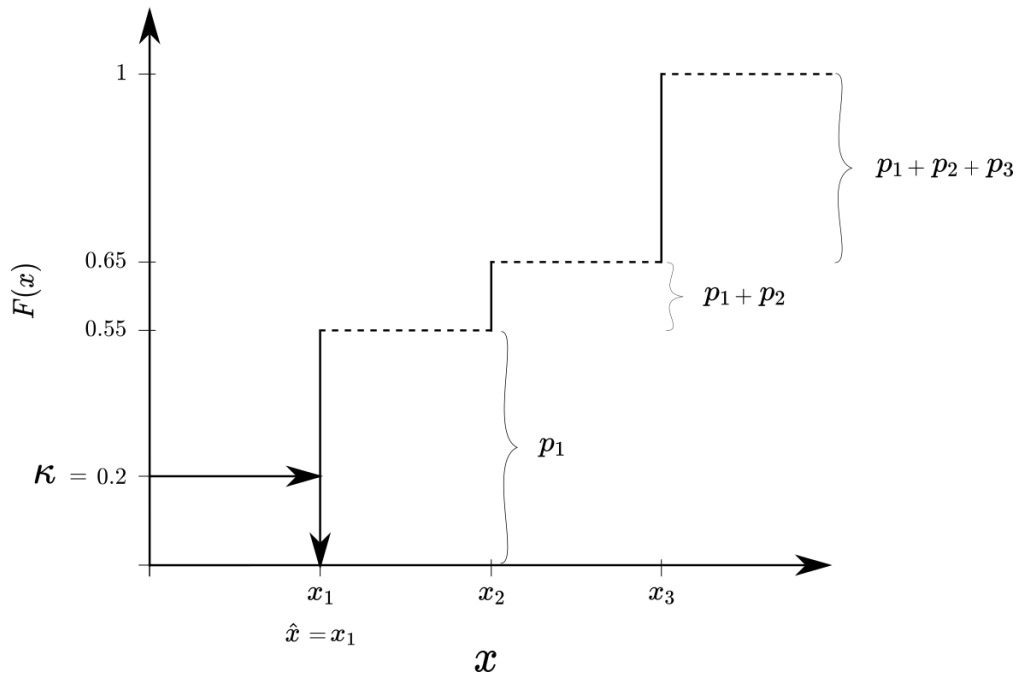


FIGURE 2.5: Monte Carlo sampling for a discrete distribution [82].

where x denotes outcome being considered and x_i denotes other possible outcomes. The probability of each outcome x_i is denoted by $p(x_i)$. A continuous CDF, on the other hand, typically requires a PDF f to be specified and is expressed mathematically as

$$F(x) = \int_{-\infty}^x f(u) du. \quad (2.9)$$

The version of the Monte Carlo simulation relevant to the topic of this thesis involves generating values for a continuous random variable $\kappa \in [0, 1]$. The CDF of the probability distribution is set equal to the generated number, that is $F(x) = \kappa$ [82]. A variable value may then be generated by calculating the corresponding x -value of the CDF, denoted by \hat{x} . This process is iteratively repeated until the required number of variable values have been sampled according to their respective probabilities of occurrence. An illustration of how the Monte Carlo simulation sampling is performed for discrete and continuous probability distributions may be found in Figures 2.5 and 2.6, respectively.

2.3 Combinatorial optimisation problem solution methodologies

There is a wide variety of different types of solution approaches in the literature for solving combinatorial optimisation problems (optimisation problems in which the decision variables are discrete). The three main classes of solution approaches are *exact solution methodologies*, *heuristic solution methodologies* and *metaheuristic solution methodologies*. A further distinction is made in the class of heuristic solution methodologies between *iterative heuristics*, *constructive heuristics* and *local search heuristics*, while a distinction is made in the class of metaheuristic solution methodologies between *trajectory-based* and *population-based* metaheuristics. Each of these classes of solution approaches is reviewed briefly in this section.

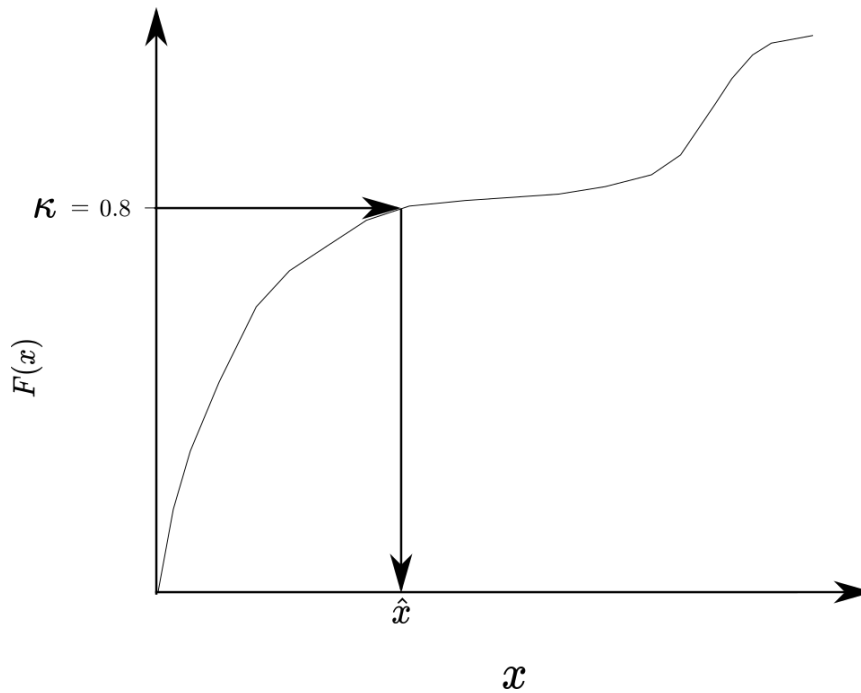


FIGURE 2.6: Monte Carlo sampling for a continuous distribution [82].

2.3.1 Exact solution methodologies

The aim of an exact solution approach is to find a globally optimal solution to a combinatorial optimisation problem by means of an extensive search, which may be either explicit or implicit. This type of solution approach is problem-specific and holds the advantage of producing an exact solution. If the dimensions of the optimisation problem instance are small, such an exact approach may be used to solve the instance within a short time frame. For larger problem instances, however, such an approach may require vast amounts of computational resources (time and/or memory).

The most obvious explicit approach towards solving a combinatorial optimisation problem exactly is the method of *total enumeration* [59]. This method simply involves performing a complete enumeration of all the possible solutions to the given problem iteratively whilst maintaining the best solution encountered during the process in memory. The method of total enumeration may be impractical for large problem instances as it may be computationally too expensive to consider all candidate solutions.

When larger problem instances are considered, a less computationally expensive, preferably implicit, approach may be required. One such exact solution approach is the well-known *branch-and-bound algorithm* due to Doig and Land [46]. This method systematically, yet implicitly, enumerates groupings of possible candidate solutions in order to obtain a globally optimal solution by a process of successively discarding subsets of provably inferior candidate solutions [7]. While this allows for a more efficient solution process than total enumeration, it may also be intractable for very large problem instances.

Consider a combinatorial maximisation problem in which the goal is to maximise an objective function $h(\mathbf{x})$, where \mathbf{x} denotes a vector of discrete decision variables. Suppose the search space (the set of all possible candidate solutions) to this problem is denoted by \mathcal{Q} . As the name suggests, two procedures form the main operations during the branch-and-bound process, namely *branching* and *bounding*. From a given subset of candidate solutions, $\mathcal{Q}' \subseteq \mathcal{Q}$, the branching

procedure returns two or more smaller, disjoint sets $\mathcal{Q}'_1, \mathcal{Q}'_2, \dots$ of which the union is \mathcal{Q}' . As the branch and bound algorithm proceeds, a *search tree* is built up to capture its progress. Each node in this tree-like structure represents the unique subsets of \mathcal{Q}' as the algorithm iterates. From a set of objective function values $\{h(\mathbf{x}_1), h(\mathbf{x}_2), \dots\}$ the maximum value of $h(\mathbf{x})$ is calculated over the subset \mathcal{Q}' . In this way, a largest objective function value $\{h(\mathbf{x}_i)\}$ is associated with some decision variable value \mathbf{x}_i in \mathcal{Q}'_i .

The bounding procedure is required to produce upper and lower bounds on the maximum value of $h(\mathbf{x})$ for each subset \mathcal{Q}' . The literature describes various methods that may be used for this purpose. There is no formalised, universal bounding procedure; instead, bounding procedures are problem-specific. If an upper bound on the objective function for a specific node \mathcal{Q}'_i of the branch-and-bound tree is less than the lower bound of another node \mathcal{Q}'_j (with $i \neq j$), the node \mathcal{Q}'_i is rejected from the search (*i.e.* subsequently ignored). The algorithm keeps track of the largest lower bound encountered throughout the search, captured in a global variable denoted by b_ℓ . Therefore, nodes with upper bound values smaller than b_ℓ are discarded. This process of eliminating subsets is called *pruning*. A search termination criterion is required. The algorithm terminates when the current set \mathcal{Q} only constitutes a single element or the upper bound of \mathcal{Q} corresponds to the lower bound of any objective function. When this occurs, the objective function has attained a maximum in the set \mathcal{Q} .

2.3.2 Heuristic solution methodologies

In a heuristic solution approach, the aim is to find a high-quality or near-optimal solution to a combinatorial optimisation problem which would suffice if an exact solution cannot be found within an acceptable time frame [33]. This type of solution approach therefore does not produce an exact solution. There are three broad classes of heuristic solution approaches in the operations research literature, namely *iterative heuristics*, *constructive heuristics* and *local search heuristics*.

Iterative heuristics are rigid procedures that, as the name suggests, are carried out in an iterative manner [33]. Each iteration produces a new solution to the combinatorial optimisation problem instance at hand in the hope that it is better than the previous solutions encountered. This is achieved by producing these solutions based on intuition and expert knowledge as rules of thumb. The process is repeated until a termination criterion is satisfied. During the execution of the algorithm, the best solution encountered is stored, compared with each newly produced solution and updated if a higher-quality solution is uncovered [47]. When the algorithm terminates, this best solution found is returned as approximate solution to the combinatorial optimisation problem instance.

Constructive heuristics, on the other hand, function slightly differently. They are greedy algorithms which iteratively select solution components that are each considered to be the best possible immediate options available. These iterations do not take into account any form of future consequences of the greedy choices made. A pre-defined set of rules is employed by the heuristic when populating an approximate solution to the combinatorial optimisation problem at hand, starting from an empty solution and continuing until a complete solution has been constructed [65].

Finally, local search heuristics start by generating an initial random solution to the combinatorial optimisation problem at hand which is a complete candidate solution [60]. This is often achieved in a random fashion. A single element or component of the initial solution is then changed in an attempt to produce an improved solution. This process is repeated, yielding a sequence of improving solutions. In cases where a newly generated solution is worse than the current

solution, another change is made to the current solution in an attempt to produce an improved solution. This type of heuristic typically proceeds until no further improving solutions can be made to the current solution. Local search heuristics often converge to local optima instead of global optima.

All of the aforementioned heuristic solution approaches are referred to in the literature as *ad hoc* approaches, because they are typically implemented in a way that caters specifically for a certain type of combinatorial optimisation problem instead of being generic in the sense that they can be applied to other types of optimisation problems as well [33]. Therefore, a new heuristic solution approach typically has to be developed every time a new type of combinatorial optimisation problem is considered. Since this thesis is aimed at developing a generic *framework*, a generic solution methodology should rather be embedded within it for the purpose of solving combinatorial optimisation problems. Moreover, the typically greedy nature of heuristics often cause them to become trapped at local optima, thereby producing potentially substantially sub-optimal solutions. This drawback is particularly prevalent for large, non-linear combinatorial optimisation problems. For these reasons, heuristics are not incorporated into the framework proposed in this thesis.

2.3.3 Metaheuristic solution methodologies

As defined by Hillier [33], metaheuristics are “a general kind of solution methodology that orchestrates the interaction between local improvement procedures and higher-level strategies to create a process that is capable of escaping from local optima and performing a robust search of a feasible region.” These approaches provide generic guidelines and frameworks for a solution approach that can be tailored to solve various types of combinatorial optimisation problems. As mentioned, metaheuristics may be classified into two broad classes, namely *trajectory-based metaheuristics* and *population-based metaheuristics*.

Trajectory-based metaheuristics function by iteratively updating single candidate solutions (not necessarily greedily), with some provisional mechanism for escaping from locally optimal solutions. Arguably, the two most notable metaheuristics in this class include the method of *simulated annealing* (SA), proposed by Kirkpatrick *et al.* [42] in 1983, and the method of *tabu search* (TS), proposed by Glover [28] in 1986.

SA mimics the process of annealing in metallurgy during which metals are heated and then allowed to cool slowly in stages, called epochs. The molecules of the heated metal vibrate in an excited fashion between various high-energy states (resembling an exploration of candidate solutions in the context of optimisation). As the temperature is lowered, these vibrations are less vigorous, leading to stable low-energy states (resembling an exploitation of the neighbourhood of good candidate solutions in the context of optimisation). The energy level of a molecular state represents the objective function value of a candidate solution in SA, while the cooling process controls the degree of trade-off between exploration and exploitation. During each iteration of SA, a neighbouring solution of the current solution is generated randomly. If this solution is superior to the current solution, then it becomes the new current solution with certainty, while if it is inferior to the current solution, it only becomes the new current solution probabilistically, guided by the so-called Metropolis rule [57]. The ability to accept non-improving solutions during the iterative search process allows the method to escape from local optima, while the gradual lowering of a control parameter, called the temperature, avoids cycles during the search process.

Like SA, the method of TS is also able to escape from local optima by accepting non-improving neighbouring solutions of the current solution. The algorithm is, however, entirely deterministic,

whereas SA contains stochastic elements. The method of TS employs a set of memory structures, namely short-term, medium-term and long-term memories, in order to prevent cycling during the search, exploiting promising areas of the solution space and exploring unvisited areas of the solution space, respectively.

In population-based methods, on the other hand, a population of multiple candidate solutions is iteratively modified in parallel with the objective of locating a globally optimal solution. There are two main classes of solution modification methodologies, namely genetically inspired modification (based on Darwins [14] theory of evolution by natural selection) and modification based on swarm intelligence (mimicking the foraging swarm behaviour of animals such as ants or birds). The methods of *ant colony optimisation* (ACO), proposed by Dorigo [17] in 1992, and the method of *particle swarm optimisation* (PSO), proposed by Eberhart *et al.* [40] in 1995, are arguably the most popular population-based metaheuristics in the latter class, while the *genetic algorithm* (GA), proposed in 1975 by Holland [34], is arguably the best-known population-based metaheuristic in the former class.

ACO is a multi-agent system in which the behaviour of each agent is inspired by the behaviour of a real ant. The main interest in real ant behaviour stems from the fact that ants exhibit collective behaviour when they perform complex tasks, such as the transportation of food to their nest along shortest paths between food sources and the nest. Ants cannot see well and so they communicate with each other by depositing a trail of pheromone (an olfactory and volatile substance) during their foraging trips. Other ants can sense this pheromone, which evaporates over time. The role of the deposited pheromone trails is to guide other ants toward target points (typically a food source or their nest). The larger the amount of pheromone present along a particular path, the larger the probability that other ants will follow that path. As more ants follow paths with a strong pheromone scent, they continue to deposit more pheromone while following the path, thereby strengthening the pheromone trail along that path. If a long path and a short path are traversed by the same number of ants, then the long path will possess a weaker pheromone trail than the short path, because it takes longer for ants travelling to and from along it to strengthen the pheromone presence while it is evaporating. Since pheromone trails therefore tend to be stronger along shorter paths, the ants can use these trails to discover short paths between food sources and their nest. ACO has traditionally been applied successfully to combinatorial optimisation problems involving distance optimisations (such as routing and facility location problems).

PSO mimics the social behaviour of natural organisms such as bird flocking and fish schooling when sourcing food. In these swarms, a coordinated behaviour typically emerges from local individual member movements without any central control being exerted on them. The swarm is modelled as particles flying around in a high-dimensional search space. The position of each particle represents a candidate solution to the optimisation problem under consideration. The success of some particles in the swarm and the momentum of a particle (in terms of discovering superior areas within the solution space) influences the behaviour of that particle and its peers. While PSO was originally designed to solve continuous optimisation problems, discrete versions of the metaheuristic have subsequently been proposed for solving combinatorial optimisation problems.

Finally, a GA mimics the iterative evolution of a biological population over time by applying stochastic operators, such as crossover representing sexual reproduction and mutation representing the introduction of imperfections into the copying process of genetic material during biological inheritance of the population offspring.

It is well known that trajectory-based metaheuristics are more suited to the exploitation of good regions of the feasible region of a combinatorial optimisation problem at hand, whereas

population-based metaheuristics are better suited to the exploration of hitherto unvisited regions of this feasible region [36]. Since the combinatorial optimisation problem considered later in this thesis has a very large solution space, a metaheuristic in the class of genetically inspired population-based metaheuristics is embedded in the framework proposed later in this thesis. More specifically, the oldest and arguably most famous metaheuristic in this class, namely the GA, is incorporated into the framework. The following section is therefore devoted to a more detailed description of the working of this algorithm.

2.4 The genetic algorithm

As mentioned, the GA has its roots in Darwin's theory of evolution by natural selection. For this reason, GA terminology is traditionally borrowed from the realm of biology. The algorithm initialises by generating an initial population of candidate solutions known as *chromosomes*. Thereafter, the algorithm iteratively allows for the evolution of these chromosomes over time in a controlled environment by applying three stochastic operators, namely *selection*, *crossover* and *mutation*. The quality of each solution is measured by employing a fitness function which assigns a fitness value to every candidate solution generated. During each iteration, pairs of solutions are selected from the population, known as the *parent solutions*. The literature contains a variety of selection operators that may be employed for the purpose of selecting these parent pairs. This selection process is generally based on the various fitness values of the solutions selected from the population [67]. Thereafter, the parent solutions mate in their pairs, each pair creating two new solutions known as *offspring solutions*. These solutions have similar characteristics to those of their parent solutions and may form part of the population during the next generation. The mating process involves application of a *crossover* operator (although in some cases, the algorithm does not perform any crossover, instead selecting the two parent solutions to become offspring solutions themselves). After mating has occurred, the algorithm allows for a small subset of offspring solutions to be changed slightly by employing the *mutation* operator. The purpose of this operator is to promote diversity in the population seeing that offspring solutions inherit characteristics from their parent solutions. This is required to ensure that unvisited areas of the solution space are explored and avoid premature convergence of the algorithm towards solutions that all represent poor local optima. The probability of mutation being performed on a solution is typically very small, and so the main source of variation in the GA is its crossover operator. Following the application of the aforementioned three operators, the offspring solutions form part of the population during the next generation through a process called *replacement*. This entire procedure is iteratively executed until a stopping criterion is met. The working of a typical GA implementation is illustrated in flowchart form in Figure 2.7.

2.4.1 Initial population generation

Population-based metaheuristics tend to be explorative in nature. This leads to many researchers not paying careful attention to the process of initial population generation. The initial population may in some cases, however, play a critical role in the performance of the algorithm [50]. The main objective of an effective procedure for initial population generation is to promote diversification. This is required to prevent premature convergence of the algorithm towards a poor local optimum [79]. The various initial population generation strategies available in the literature may be classified into four primary categories, namely *random generation*, *sequential diversification*, *parallel diversification* and *heuristic initialization*. Each of these strategies exhibits various attributes and may be particularly applicable to different types of combinatorial

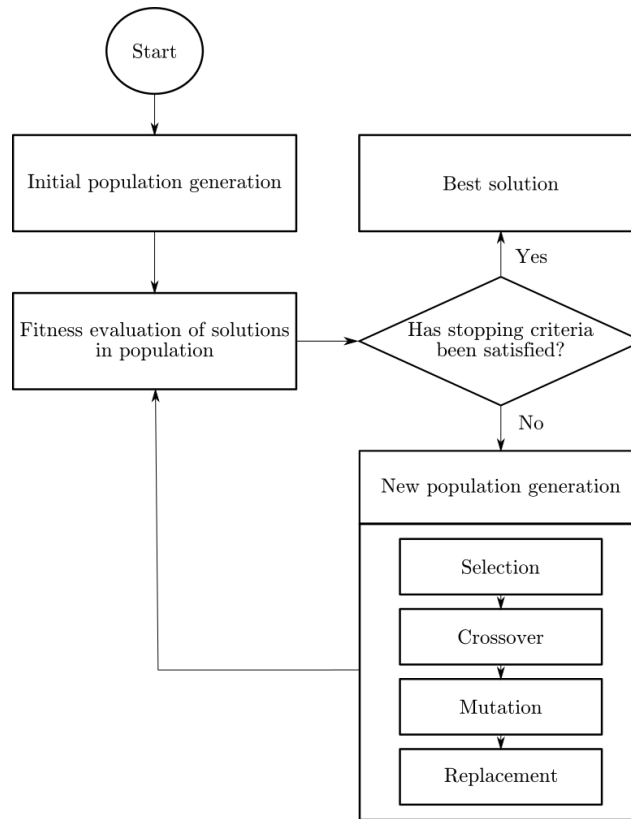


FIGURE 2.7: The working of a typical GA implementation [11].

TABLE 2.1: Evaluation of GA initial population generation strategies on a scale which ranges between 1 and 4 (inclusive), with 1 being the worst and 4 being the best [79].

Strategies	Attributes		
	Diversity	Computational cost	Quality of initial solution
Random	2	3	1
Sequential diversification	4	2	1
Parallel diversification	4	3	1
Heuristic	1	1	3

optimisation problems. A comparison of the relative strengths and weaknesses of these strategies may be found in Table 2.1.

As the name suggests, the random generation strategy generates an initial population in a stochastic manner according to a statistical distribution (often the uniform distribution). Solutions are generated independently from one another and are required to be feasible [79].

Similarly, sequential diversification is also a stochastic strategy accompanied by a number of additional rules. The solutions of this strategy are generated in a sequence as to optimise diversity [79]. The generation of a solution is therefore not independent of that of other solutions that already exist in the population. After the new solution has been generated it will also have an effect on the generation of the subsequent solutions.

A well-known form of this strategy is the *simple sequential inhibition* (SSI) process [15]. A current subpopulation, denoted by C , initially contains only one randomly generated solution. Each new solution is stochastically generated but may not be within a threshold distance Δ

of any existing solution in the subpopulation C . Therefore, any newly generated solution is repeatedly regenerated until this criterion is satisfied. The entire process is repeated until the population contains a predetermined number of solutions [79]. This method allows the initial population to cover large portions of the solution space and avoids high concentrations of initial solutions in certain areas [11], thereby prioritising diversification. Any distance measure may be adopted in the SSI process, such as, for example, the Euclidean distance if candidate solutions are encoded as integer vectors. All solutions within a pre-specified distance from a solution forms part of the neighbourhood of that solution. A schematic two-dimensional illustration of how the neighbourhood of each solution plays a role during the SSI process is depicted in Figure 2.8.

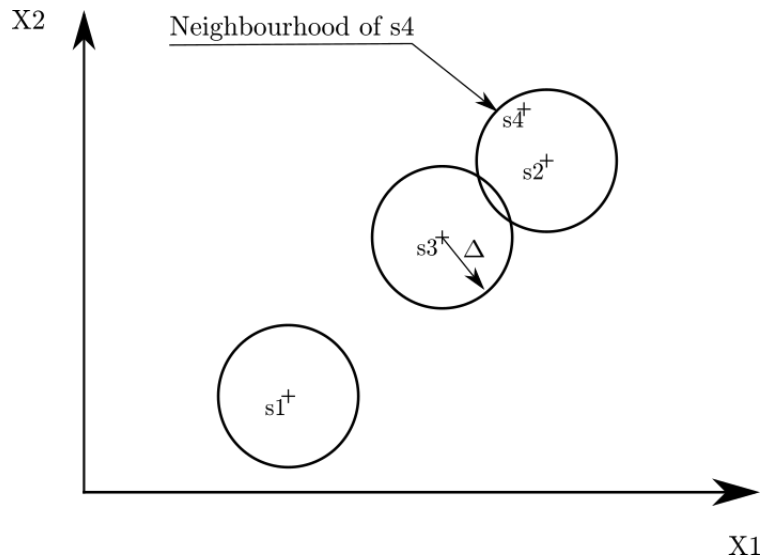


FIGURE 2.8: Schematic illustration of the role of a neighbourhood (represented by a circle) being applied during the SSI procedure, where solution s_4 is not accepted because it falls within the neighbourhood or threshold distance of s_2 [11].

Parallel diversification, on the other hand, involves generating solutions in a parallel fashion. This is typically achieved by partitioning the search space into various sub-spaces. A solution is then generated randomly in each individual sub-space [79]. This ensures that there are no excessively large spaces between the solutions of the initial population and also promotes diversification in the initial population. A two-dimensional illustration of how the solution space may be partitioned into squares for this purpose is illustrated in Figure 2.9.

The final initial population generation strategy is heuristic initialisation. This strategy employs a heuristic, such as a local search, for example, to initialise the population. This strategy prioritises diversity the least of the strategies reviewed here and may result in premature convergence. It typically produces high-quality initial populations [79], but the effectiveness of the subsequent algorithmic evolution will depend on the nature of the fitness landscape of the combinatorial optimisation problem under consideration (*i.e.* on how many local optima there are and whether the local optimal are close to global optima). In certain instances heuristic initialisation can be more effective than a stochastic initial population generation strategy.

2.4.2 Selection strategies

The first step during each iteration of a GA is the application of its selection operator to the population of candidate solutions. This stochastic operator is responsible for selecting which pairs of solutions will partake in reproduction to produce offspring candidate solutions. As men-

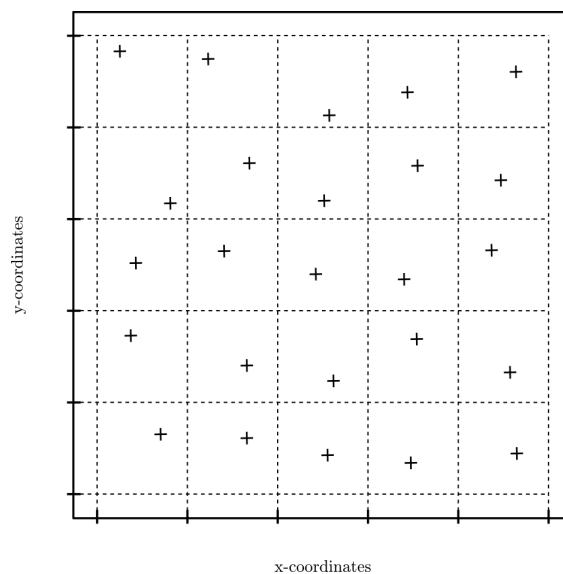


FIGURE 2.9: A partitioning of a two-dimensional solution space as per initialisation by parallel diversification [79].

tioned, these strategies typically take the fitness values of the various solutions in the population into account when selecting parent solutions [67]. The main objective is to select fitter solutions to partake in crossover (*i.e.* to have a higher probability of passing on the characteristics of high-quality solutions to the next generation) [79]. The extent to which parent solutions pass on their characteristics is called the *selection pressure*. This typically allows for subsequent populations to converge towards fitter and fitter solutions over time. There may, however, also be useful genetic material in some of the worst solutions in a population and for this reason they must also be considered for selection, with a much lower probability, when the selection operator is employed.

The literature describes two main classes of fitness assignment which subsequently result in two different types of selection strategies. The first class is *proportional fitness assignment* according to which absolute fitness values are assigned to the candidate solutions in the population. In this case, selection strategies such as *tournament selection*, *roulette wheel selection* and *stochastic universal sampling* are applicable. The other class of fitness assignment methodologies is referred to as *rank-based fitness assignment*, which involves assigning relative fitness values to the candidate solutions after which all the solutions are ranked in ascending order of fitness. This class would typically employ a *rank-based selection* strategy.

The roulette wheel selection strategy involves selecting a parent candidate solution from the population with a probability that is directly proportional to the fitness of the solution. Therefore, solutions associated with larger fitness values have a higher probability of being selected as parent solutions. The working of this strategy is akin to that of a roulette wheel in a casino, but with slots weighed by the respective fitness values of the solutions in proportion with each other [67]. A linear search is then employed and the probability of solution i being selected as a parent is

$$p(i) = \frac{f(i)}{\sum_{j=1}^N f(j)}, \quad (2.10)$$

where the fitness of solution i is denoted by $f(i)$ and the total size of the population is denoted by N [47]. The roulette wheel strategy is the simplest selection strategy, but may occasionally lead to premature convergence, resulting in suboptimal solutions [67].

The tournament selection strategy is initialised by selecting a random sample of k candidate solutions from the population to partake in a tournament. The parameter k is user-specified and is known as the *tournament size*. Again, the probability of a candidate solution being selected is proportionate to its fitness. Following the establishment of the tournament, the winner of the tournament is selected deterministically as the solution with the largest fitness value and becomes a parent solution [79]. Use of this strategy is illustrated schematically in Figure 2.10.

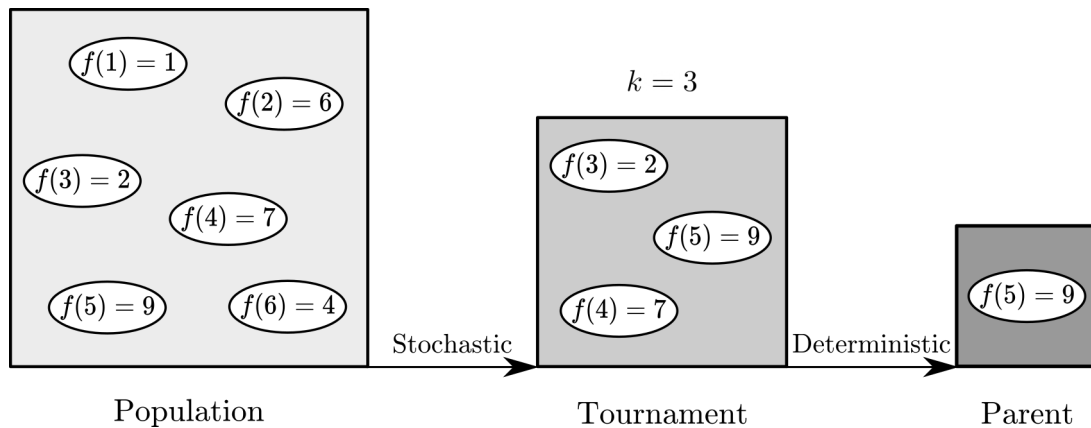


FIGURE 2.10: A schematic illustration of the working of the tournament selection strategy applied to a population of $N = 6$ candidate solutions, with a tournament of size $k = 3$ being selected [79].

As mentioned above, a rank-based selection strategy requires the population to be ranked according to their fitness values. Following the sorting of the population, a unique rank is assigned to each solution. The fittest solution is assigned rank μ and the least fit solution rank 1 [67]. The probability of solution i being selected according to this strategy is

$$p(i) = \frac{2 - s}{\mu} + \frac{2(s - 1)r(i)}{\mu(\mu - 1)}, \quad (2.11)$$

where the total population size is again denoted by N and the rank of solution i is denoted by $r(i)$. The expression in (2.11) also employs a parameter $s \in [1, 2]$, known as the selection pressure [79].

2.4.3 Crossover strategies

After having selected pairs of parent solutions for crossover, mating must occur in order to produce offspring solutions. The crossover operator is employed for this purpose. The role of crossover is to ensure that the next generation of candidate solutions inherits characteristics from the parent solutions [79].

Crossover is also a stochastic operator and is associated with a probability, denoted by p_c . This probability is typically relatively large due to the fact that crossover is the primary source of variability within the GA [33]. The literature describes numerous crossover techniques which may be applied. These operators depend on the method of solution encoding and are therefore problem context-specific. Since solutions are encoded as integer vectors later in this thesis, only crossover operators pertaining to this type of vector encoding are reviewed in this section. The most popular crossover operators in this context include 1-point and 2-point crossovers, which may be generalised to n -point crossover. Other well-known crossover techniques include *uniform crossover* and *intermediate crossover*.

When employing 1-point crossover, a position within the parent solutions is randomly selected at which segmentation takes place. The resulting segments of the two parent solutions (the partial solution vector encodings before and after the selected position) are then interchanged in order to produce two offspring solutions [47]. The notion of n -point crossover works similarly, although in this case n positions are selected randomly instead of only one. Figure 2.11 contains illustrations of the working of 1-point crossover and 2-point crossover. Uniform crossover is performed in a similar fashion, although the sizes of the segments do not have to be taken into account. This type of crossover is illustrated in Figure 2.12.

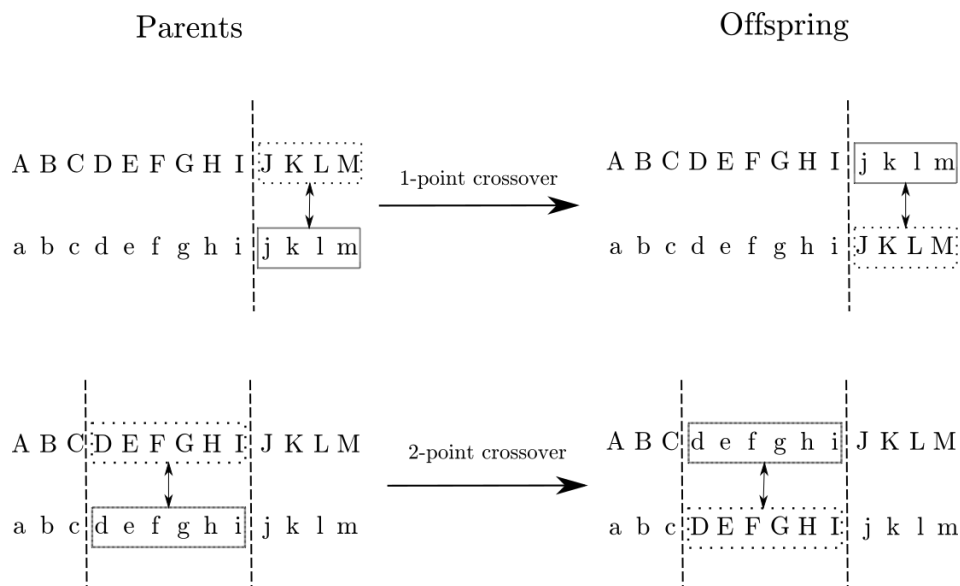


FIGURE 2.11: 1-Point crossover and 2-point crossover in a GA [79].

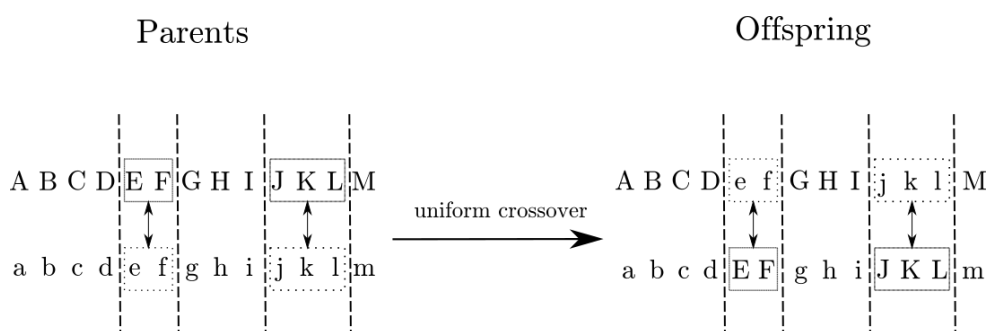


FIGURE 2.12: Uniform crossover in a GA [79].

2.4.4 Mutation strategies

After the crossover operator has been applied to pairs of parent solutions selected from the population, the final stochastic operator that allows for the evolution of solutions, known as mutation, may be performed. In contrast to the crossover operator, the mutation operator is performed on selected elements in an offspring solution encoded as an integer vector instead of on entire solution segments. The operator performs small alterations or perturbations to a small selection of solutions in the population [11]. Mutation is associated with a probability denoted by p_m . This probability is typically very small so as to prevent too much diversification

and a loss of the good characteristics of parent solutions built up iteratively over time. Three important factors have to be considered when employing a mutation strategy [79]:

Ergodicity. Every solution in the solution space should be reachable.

Validity. Feasible solutions should be produced and so the constraints of the combinatorial optimisation problem under consideration have to be taken into account.

Locality. The mutation size should be small and produce minimal changes. When small changes are made to solutions, the effects on these fitness values are not expected to be large. If this notion of locality does not hold, the search may tend toward a random search.

Mutation in integer vector solution representations typically consists of integer swap mutations, which involves swapping two randomly selected elements in a candidate solution, as illustrated in Figure 2.13.

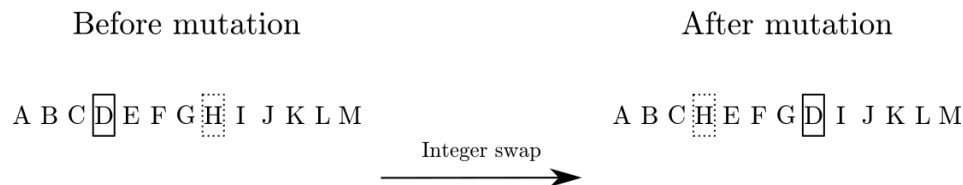


FIGURE 2.13: An illustration of integer swap mutation.

2.4.5 Replacement strategies

The crossover and mutation operators are required in the GA to promote variation between generations of population. Following the creation of offspring solutions, a replacement strategy is followed during which the subsequent generation is created. The replacement strategy is concerned with selecting which offspring and/or parent solutions may survive an iteration [79]. Replacement strategies may be categorised into two primary classes, namely *generational replacement* and *steady-state replacement*:

Generational replacement. The original replacement strategy proposed by Holland [34] consists of replacing the entire population of parent solutions with the offspring solutions generated. The offspring solutions then become the parent population during the subsequent iteration.

Steady-state replacement. Only one or two offspring solutions per generation are generated [49]. The offspring solutions then replace selected parent solutions from the population during the next generation.

There are numerous variations on the replacement techniques mentioned above in the literature, with *elitism* perhaps being the most notable [79]. This involves allowing the best solutions in the combined parent and offspring populations to survive to the next generation while keeping the population size constant. A variation of the elitism replacement strategy [11] is where the offspring population constitutes the entire population of the subsequent generation with the addition that if the incumbent (*i.e.* best solution) of the parent population is superior to that of the offspring population, then it replaces the worst solution in the offspring population. This allows for the best solution characteristics always to propagate in the population. A flow chart representation of this strategy is proffered in Figure 2.14.

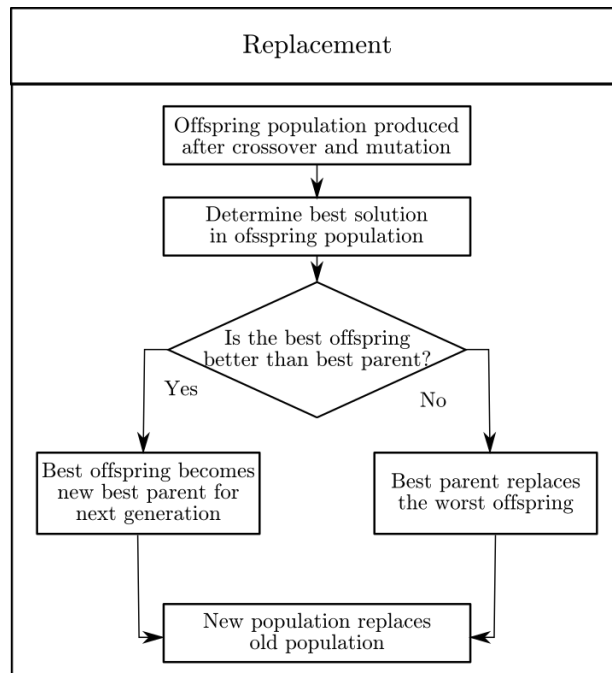


FIGURE 2.14: A variation on the elitism replacement strategy [11].

2.4.6 Stopping criteria

As mentioned, the GA is an iterative procedure, employing the aforementioned operations until a pre-determined stopping criterion is satisfied. Stopping criteria proposed in the literature may be categorised into two primary classes, namely *static criteria* and *adaptive criteria* [79]. When employing a static criterion, the end of the search is known *a priori*. Specifying a fixed number of generational iterations (*i.e.* a maximum number of algorithmic iterations) is an example of a static termination criterion. In the case of an adaptive criterion, the end of the search is not known *a priori*. Specifying a fixed number of generational iterations without improvement of the incumbent is an example of an adaptive termination criterion. Criteria for algorithmic termination are typically concerned with the diversity of the population and are therefore aimed at search termination when the level of diversity falls below a predetermined threshold.

2.5 Chapter summary

This chapter was devoted to a discussion on a number of preliminary concepts aimed at familiarising an uninitiated reader with the mathematical notions that underpin the material presented in the remainder of this thesis. The estimation of regression coefficients, described in §2.1, plays a vital role in understanding statistical models. Stochastic sampling by Monte Carlo simulation, described in §2.2, is also a critical component of the combinatorial optimisation model proposed later in this thesis. Various methods for the solution of such models were also described in §2.3, including exact solution methodologies (in §2.3.1), heuristic solution methodologies (in §2.3.2) and metaheuristic solution methodologies (in §2.3.3). The particular metaheuristic solution methodology employed later in this thesis, the GA, was finally reviewed in some detail in §2.4.

CHAPTER 3

Spatial planning tools

Contents

3.1	Urban simulation	29
	3.1.1 <i>Steps in a typical urban simulation study</i>	30
	3.1.2 <i>Validation and verification of a typical urban simulation study</i>	31
3.2	Integrated transport and land use models	32
	3.2.1 <i>The Integrated Transportation and Land Use Package</i>	32
	3.2.2 <i>MEPLAN</i>	33
	3.2.3 <i>Modelo de Uso de Suelo de SAntiago</i>	33
3.3	The UrbanSim simulation software suite	34
	3.3.1 <i>UrbanSim design considerations</i>	35
	3.3.2 <i>UrbanSim software architecture</i>	37
	3.3.3 <i>The UrbanSim model structure</i>	38
	3.3.4 <i>Location choice models</i>	43
	3.3.5 <i>UrbanSim validation</i>	50
3.4	Chapter summary	50

This chapter is devoted to a review of urban simulation modelling and how it has developed over time. The chapter opens with an overview discussion on urban simulation, the steps involved in a typical urban simulation study and a brief description of validation and verification processes for urban simulation models. The discussion then turns to a background on and the use of *integrated transport and land use models* (ITLUMs) as an illustration of how urban simulation models have evolved. A thorough description of the original UrbanSim simulation software suite, the model on which the optimisation process proposed later in this thesis is based, is then provided and this description is supplemented by a discussion on location choice models and their use within the UrbanSim environment. This chapter finally closes with a brief summary of all that has been discussed.

3.1 Urban simulation

The evolution of urban economies, transport systems, political and social structures, technologies, and infrastructure has resulted in an ever-increasing complexity of urban systems. Efficiency in these systems has become paramount due to the importance of providing the citizens

populating urban systems with accessibility to scarce resources, which typically involve various trade-offs between conflicting values and priorities of a variety of stakeholders. It is, therefore, not helpful to consider major transport and land use investments and urban development policies in isolation [89]. The use of theoretical and mathematical models, such as the well-known work on the Monocentric Model of a city [1], have long been recognised as important tools for facilitating a better understanding and reduction of the complexity of urban systems.

While theoretical models have indeed facilitated a broad understanding of urban systems, the models employed for this purpose typically lack the ability to assist in identifying the operational requirements of decision making related to policy and planning [89]. In order to address this problem, computerised models for simulating dynamic interactions and processes of urban transport and development (better known as *urban simulation*) have been developed since the 1960s.

3.1.1 Steps in a typical urban simulation study

The recommended steps taken during a typical urban simulation study are continually evolving as a result of the increased complexity of urban systems and rapid development of technology. The objective, however, remains fairly constant and pertains to the convergence of agreement on a specific set of goals and the policies that have to be implemented in order to achieve these goals. A model development process applicable to the urban simulation models considered in this thesis may be summarised in the following twelve steps [89]:

1. *Assess the institutional, political and technical context*, which refers to answering questions such as: Who will be using the model? Who will be affected by the model? What are the technical requirements and limitations of the decision problem at hand? and What are the mandates and limitations of the stakeholders?
2. *Assess stakeholders, value conflicts and public policy objectives*, which typically involves mitigating conflicting values of stakeholders involved in creating a coordinated urban vision for the future of the areas related to specific objectives.
3. *Develop measurable benchmarks for objectives*, which are important for evaluating progress toward objectives established in the previous step.
4. *Identify inventory policies to be tested*, which refers to typical policy scenarios that must be considered during the design of the model, such as expansion of roadways, congestion pricing and incentives for infill and redevelopment.
5. *Map policy inputs to outcomes*, which involves conceiving a conceptual mapping of input policies and their effects on potential outcomes.
6. *Assess model requirements*, which refers to the identification of typical considerations to which a model should adhere, such as being sensitive to pricing policies on travel behaviour or being able to assess policy effects over a certain number of years.
7. *Prepare input data*, which refers to the collection and preparation of adequate and relevant data for the purpose of constructing a simulation model.
8. *Develop a model specification*, which entails the separation of model components within an urban modelling environment into reasonable distinct components. This is necessary

because different types of agents and processes have different parameter estimations. Typical model components include household allocation, employment allocation, real-estate development and land price simulation.

9. *Estimate model parameters*, which involves estimating the individual coefficients of the model components. Typical estimation methods include the method of maximum likelihood, the method of least squares and probabilistic or Monte Carlo simulation.
10. *Calibrate the model system*, which refers to the calibration of the entire urban modelling system after having established the aforementioned coefficients.
11. *Validate the model system*, which involves running the model on data which have not yet been used for estimation or calibration purposes.
12. *Operational use*, which refers to the actual operation of the model during which a model user inputs a baseline scenario and compares alternative scenarios with the baseline scenario in order to assess the effectiveness and desirability of various policy scenarios.

The aforementioned steps provide an adequate framework for the development of urban simulation models and the use of these models. No existing model can perfectly predict the future, however, because the complexities and nuances of real urban environments render urban simulation and its ability to predict particularly challenging. Therefore, these models are more useful as indicators of likely directions and trend magnitudes of a variety of policy scenarios when compared to a baseline scenario.

3.1.2 Validation and verification of a typical urban simulation study

When simulation models are used for decision making it is imperative that the models are operationally accurate and provide an adequate representation of what would occur in reality. In order to address this concern, a simulation model has to be verified and validated. The definition of model verification adopted here is “ensuring that the computer program of the computerized model and its implementation are correct” [69]. The definition of model validation, on the other hand, is “substantiation that a computerized model within its domain of applicability possesses a satisfactory range of accuracy consistent with the intended application of the model” [69].

During a typical urban simulation validation procedure, data are required from a time period which acts as the initial conditions of the model and are used for coefficient estimation. A second set of data from a later time period is then also required which facilitates a comparison with the predictions of the model based on the data from the initial conditions. If this comparison is satisfactory, the model is deemed to be validated and the predictive power of the model is acknowledged [89]. This process is called historical validation and is one of the most informative validation techniques for urban simulation models. The urban simulation model UrbanSim, for example, was validated historically when it was applied to Eugene-Springfield, Oregon [84]. Certain practical constraints, such as a lack of historical data, may render such a historical validation approach impossible, in which case alternative validation techniques from the literature have to be considered [69]. Some of these alternative validation techniques include the comparison of model results with the results of other models, face validation (involving asking field specialists whether the behaviour of the model may be considered reasonable) and internal validation (involving several runs of a stochastic model in order to determine internal variability which may have an effect on the reasonability and representativeness of the output of the model).

Verification of a simulation model involves computerised model verification. This ensures that the computer programming and implementation of a model are correct and is typically achieved by performing structured walk-throughs and traces of the model implementation code [69].

3.2 Integrated transport and land use models

The 1950s saw a substantial increase in the use of automobiles in the United States of America, which resulted in a major decline in the efficiency of public transport systems. This led to an increase in a demand for research related to transport planning and travel demand modelling which, in turn, resulted in the development of the first generation of travel demand models [73]. Throughout the course of development of these models, it became apparent that transport systems and land use are interdependent. Land-use models are typically concerned with the forecasting of changes in land development, employment and households. It was observed that changes in transport systems directly affected land development patterns [29]. Employment and household locations, on the other hand, are major contributors toward trip patterns within an urban setting, therefore impacting transport system changes. The interdependency of these factors resulted in the development of ITLUMs.

For decades, the implementation of ITLUMs has been neglected in practice, with a few exceptions of the largest *metropolitan planning organisations* (MPOs) in the United States of America [19]. During the early 1990s, the implementation of ITLUMs, however, became more common due to the introduction of two United States federal policies, namely the Clean Air Act Amendments of 1990 and the Intermodal Surface Transportation Efficiency Act of 1991. Numerous ITLUMs have since been developed and implemented in aid of urban planning in metropolitan areas around the globe.

A number of ITLUMs are available in the literature, exhibiting varying levels of accessibility, usability and completeness. The remainder of the section is dedicated to providing the reader with a high-level background on some of the most noteworthy ITLUMs in the literature.

3.2.1 The Integrated Transportation and Land Use Package

In 1983, Putnam [64] developed one of the most celebrated ITLUMs, known as the *Integrated Transportation and Land Use Package* (ITLUP). This model consists of a number of sub-models, including a multinomial logit modal split sub-model and a trip assignment sub-model which support numerous assignment algorithms. The most popular models in the literature, however, are the *Employment Allocation Model* (EMPAL) and the *Disaggregate Residential Allocation Model* (DRAM). These models are based on a Lowry-derivative [48] form for allocations of employment and households. The DRAM allocates households, typically classified into four varieties based on income, to an array of spatial zones. The zonal attractiveness is the main driver of these allocations and is based on the capacity (derived from developable land and vacancy), current residential development and other socio-economic characteristics on a zonal level. Whilst generating and distributing household allocations, the DRAM simultaneously computes the corresponding trip generation. The EMPAL, on the other hand, allocates employment, also typically classified into four varieties, in a similar manner. These two models require exogenous input of employment forecasts, population forecasts, trip forecasts, activity rates and household types. In practice, the DRAM and EMPAL are, however, often applied separately from one another and concurrently with other commercial travel demand forecasting models.

The ITLUP is a less flexible and more complex ITLUM and is a non-market-based land allocation model [19]. The ITLUP exhibits relative parsimonious data requirements in comparison with other frameworks in the literature. An important advantage of the ITLUP is that the required data are typically freely available, such as household, population and employment data [73]. A disadvantage of these data requirements, however, is that the framework does not account for land market clearing processes. In an attempt to improve linkages with *geographic information system* (GIS) databases and to improve system modularity, a software-based version of ITLUP has since been developed, known as METROPILUS. The literature contains detailed documentation on this framework, provided by Putnam [64].

3.2.2 MEPLAN

Another ITLUM was developed in 1985 in the United Kingdom by the private consulting firm Marcial Echenique and Partners Ltd [20]. The MEPLAN framework is an aggregate model in which a spatial domain is partitioned into zones. The model then allocates economic activities (called sectors or factors) and households to these zones. Interactions amongst factors in these zones act as drivers for the flow of transport demand in the area under consideration. The core of the framework is a spatially disaggregated input-output matrix referred to as the *social accounting matrix*. The framework facilitates the modelling of economic activities and households as producing and consuming activities which are represented by so-called technical coefficients. The social accounting matrix contains these variable technical coefficients for a variety of space sectors and labour sectors. The framework allocates production arising to satisfy consumption among the zones through discrete choice models reacting to the production prices, thereby maintaining spatial disaggregation. The results of the allocations then determine the demand for travel in the area under consideration.

Sequential points in time are considered to simulate temporal change. The framework considers space “on-transportable” and it is consumed in the zone in which it is produced, while the supply of space for each zone remains fixed at a given point in time [35]. The consumption of space, represented by the technical coefficients, reacts according to the price of space. The price of space is established endogenously so as to ensure that demand corresponds with supply in each zone so as to conserve equilibrium at each point in time. The output prices for other sectors also follow the production-consumption principle and are established endogenously.

Logit functions, which represent route and mode choices, form part of a multimodal network that allocates travel demands for a given point in time, whilst considering congestion. Inspired by the Lowry basic sector [48], the addition of exogenous demand provides the initial circumstances for economic activity. Changes in the exogenous demand and the amount of space available per zone over successive time periods are the primary drivers for economic change which, in turn, are allocated to the zones.

3.2.3 Modelo de Uso de Suelo de Santiago

An operational floor space and urban land market model was developed by *Martnez* [52] for Santiago, Chile. The *Modelo de Uso de Suelo de SAntiago* (MUSSA) is fully connected with a detailed four-stage model known as ESTRAUS. The combined model, referred to as 5-LUT, predicts equilibrium travel demand and generates land use forecasts for Santiago. The model has been implemented to compare the relative effectiveness of numerous land use and transport policies.

MUSSA is based on the extreme application of micro-economic theory aimed at upholding consistency [35]. The model treats building stock demand and supply by applying an equilibrium approach. Demand for households or firms is based on the willingness to pay. Buyers constantly attempt to maximise their surplus whilst sellers try to maximise the selling price of houses and firms. Developers supply building stock in a manner aimed at maximising profit whilst accounting for demand. The prices of buildings are endogenously determined within the equilibrium process. The model upholds a static equilibrium in the year forecast by adjusting building stock supply in response. Consumers' expectations, denoted by the expected utility obtained from their housing, are also adjusted as a demand response. Both these responses contribute towards maintaining equilibrium. The model is a highly disaggregated ITLUM and employs traffic analysis zones as its spatial units [52]. It has the ability to differentiate between 65 household types in Santiago, and is continually being extended to accommodate further environmental impact calculations.

3.3 The UrbanSim simulation software suite

The UrbanSim simulation model is an open-source tool for simulating and predicting urban growth and its spatial distribution at different time stages in the future. It has been discovered that it may be used to assist in anticipating and accommodating spatial change in major metropolitan regions and may be applicable to South African cities as well [32]. This is typically achieved by simulating an array of policy scenarios and then using the results thus obtained to assist planners and policy makers in analysing the potential impacts of various policies on spatial development.

UrbanSim consists of various components that have been designed so that choices by businesses, developers, households and the government, the key role players in the urban development process, may be simulated by the model [81]. This means that the UrbanSim model is an analytical tool that can be used by decision-makers to evaluate the predicted consequences of policies and infrastructure investment decisions.

As urban systems have become more complex, predicting urban evolution has become more difficult. The demand for prediction tools in pursuit of this endeavour, such as mathematical and theoretical models, has therefore increased. UrbanSim is one such model which includes components for accommodating economies, space, social and political structures, transport and other infrastructure systems, and technologies. UrbanSim thus strives to be a model that produces results which can be considered as economically, socially and spatially sustainable [32]. The UrbanSim modelling approach involves incorporating mathematical descriptions as representations of parts of the real world into the model system and then iteratively running the model a number of times, simulating outcomes of policy scenarios into the future. These results are then used to evaluate predetermined indicators of land use change, thus measuring the outcome of each scenario. A schematic overview of the UrbanSim approach towards urban simulation is provided in Figure 3.1.

Within the UrbanSim environment, certain terms are used in a very specific way. The two main terms are the notion of a *policy scenario* and that of an *indicator* [32]. These terms are defined as follows:

Policy scenario: A unique combination of assumptions and policy alternatives used to achieve a certain spatial goal. An example is a study aimed at estimating the growth/decline of a population within a region, where policy alternatives could involve different zoning strategies, altering growth boundaries, establishing building density restrictions and improving local transport infrastructure.

Indicator: A variable calculated from one data field or many data fields within the UrbanSim database. A single indicator or a combination of indicators may be used to measure or represent the relative success of each policy scenario in respect of obtaining the spatial goal it set out to achieve or an indicator may simply be defined and used to evaluate simulation results.

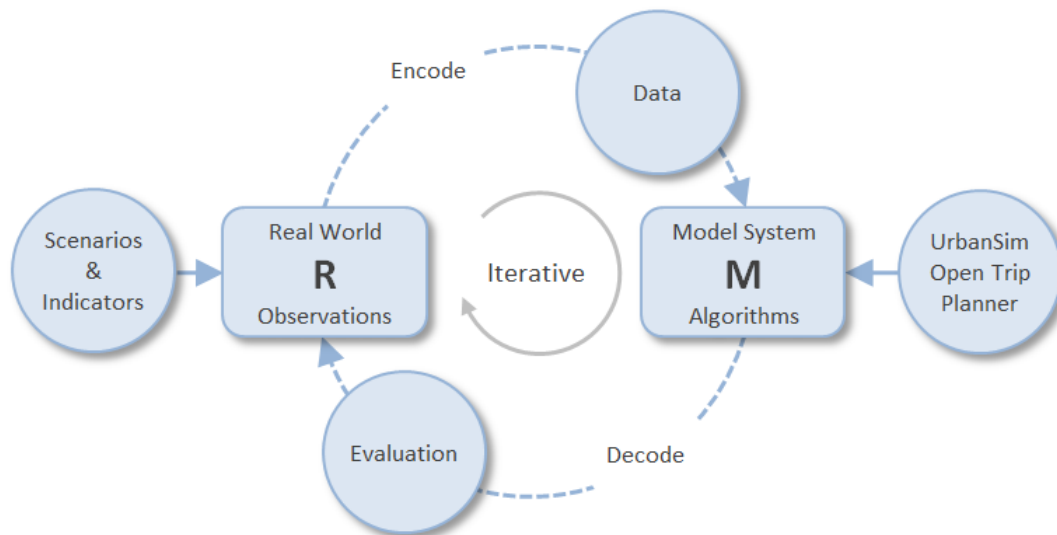


FIGURE 3.1: *Elements of urban modelling in UrbanSim [32].*

3.3.1 UrbanSim design considerations

Metropolitan planning entities have routinely used transport models for decades [87]. Land use planning has, however, traditionally not been integrated appropriately within transport planning, despite their prominent correlation. Furthermore, land use models have historically been much less advanced than transport models [87]. Many metropolitan areas simply did not use land use models in their planning or perhaps used a simple aggregate model instead which does not import policy decisions related to urban growth, zoning, boundaries, incentives and taxes at an acceptable level. This has resulted in models becoming redundant when comparing alternative scenarios involving alternative policy decisions.

Substantial progress has been made to address this problem in recent years, leading, for example, to the development of the reusable land modelling system UrbanSim, which has been integrated with numerous transport models [81]. The remainder of this section is devoted to a discussion on the the original design considerations of the UrbanSim team when developing version 1.0 of model, which was released as open-source software [81].

UrbanSim is different from other operational urban models in several ways. Three major design considerations gave rise to these differences. First, UrbanSim opted for a dynamic approach towards urban simulation as opposed to an equilibrium-based approach [35]. Equilibrium models are based on the assumption that interdependent variables in a model, such as supply, demand and price, would adjust to equilibrium with zero delay [32]. This assumption is taken from the economic theory of equilibria in which the objective is to gain general understanding when

drawing comparisons between two steady-state conditions that exist in perfectly competitive markets and which only differ in response to an external shock to the system [80]. The equilibrium approach to economics assumes that markets are perfectly competitive, that the acts of an individual do not affect prices, that resources have perfect mobility, that all competitors in a market produce homogeneous products, and that future costs and prices are clearly known to all participants in the market [80]. This approach also assumes that all sellers and buyers in the market coordinate simultaneously. These assumptions are, of course, an over-simplification of the economy when considering the many complex interactions within urban labour, transport and housing markets. Well-known models that employ equilibrium models are MEPLAN, TRANUS and DRAM/EMPAL [35].

The equilibrium approach raises serious concerns within three specific time scales that are relevant to the system interaction of land use and transport. The first is that travel behaviour may change within the scope of a day if changes occur in the transport system — this is a short-term concern [18]. Secondly, within the medium-term time frame, households and business would take slightly longer to adjust to changes within a transport system and it therefore cannot be assumed that local housing demand would adjust to changes to a transport system in the short term. Thirdly, real-estate developers are speculative and will respond more cautiously to changes in a transport system, (although responses to changes in local demand would occur immediately). This response may take several years when considering all the steps that take place during the real-estate development process [44] — it is considered over a long-term time frame [18]. Clashes between these time frames of adjusting to change is where the root problem in equilibrium modelling lies.

Some would argue that time scales are not relevant if the same outcome is obtained in the long run, seeing that adjustments over time are accounted for [18]. This is, however, not the case. Consider, for example, that during the time frame over which a developer partakes in constructing real-estate, the above-mentioned short- and medium-term changes may be occurring in a manner that renders the initial decision to construct the real-estate a sub-optimal one. This often happens and results in over or under building, a common phenomenon in urban real-estate markets [87]. Furthermore, development, even if suboptimal, is durable and has an influence on real-estate development availability and the prices of competing developers, indicating that path-dependence is an important part of the reaction to any transport system change. Also, there are many constraints to relocation which means that a change in the local transport system is unlikely to be large enough for every household and business to relocate to their so-called optimal locations when considering their transport cost in comparison with their other costs of living, which is what would occur according to a full equilibrium modelling approach [87]. This time-scale problem suggests that it would be unwise to expect real-estate demand and supply to alter in the short term if changes were to be affected to an environment as is done in the full equilibrium modelling approach.

The developers of UrbanSim have opted rather to adopt a dynamic approach towards modelling. Dynamic modelling is based on the assumption that changes in demand may be quicker than the response of the supply and that these differences in the rate of adjustment are so compelling that an urban system could be prevented from reaching equilibrium if this is not accounted for [91]. Therefore, the designers of UrbanSim have suggested that it might be more accurate to identify the relevant time scales of various short-, medium- and long-term processes, appraising the degree to which each partial equilibration takes place and using this information as the basis of the rate of adjustment [87]. This disaggregated approach is a necessity for predicting the choices and movement of the South African economy due to the heterogeneity of the South African population [32].

Another difference in the design of UrbanSim is that its developers have elected to adopt an extremely disaggregated modelling approach. In particular, UrbanSim models individual jobs, households and real-estate developments separately. It also models individual location choices separately, originally combining these into grid cells of 150×150 metres [62]. The capabilities of GIS systems have, however, since evolved and have become very flexible. The user of the software now has a choice between using grid cells, zones or cadastral parcels. Moreover, zones are customised by user discretion and within the South African context could describe a traffic analysis zone, a subplace or any other administrative/planning unit [32]. The model also employs various input data, including address-level business establishment data, parcel-level land use and real-estate inventories to micro-simulate the annual change that occurs in households, jobs and real-estate within each grid cell, zone or parcel due to real-estate developers' actions. This was a novel urban simulation modelling approach when UrbanSim was originally developed [87].

Finally, UrbanSim is an open-source tool aimed at accommodating the disaggregated simulation approach to which it is committed. The management of data and model components can thus be adapted for anyone to access source code freely, modify/improve it and then distribute it. This approach was adopted to encourage openness, collaboration and transparency of the model, whilst also increasing the speed of evolution and robustness of the software [87]. This feature is a very important consideration from a South African urban system modelling point of view as the models incorporated into UrbanSim have all been developed within the context of the developed world. This open-source approach allowed the CSIR to adapt the software for use in the South African context [32].

3.3.2 UrbanSim software architecture

The UrbanSim architecture consists of four main parts [58]:

1. *Models* are responsible for encoding the attributes of agents who participate in the simulation, such as real-estate developers and households. These models also determine the behaviour of objects on which the agents operate, such as buildings and land parcels.
2. *A model coordinator* schedules the above-mentioned models when running and informs the models if relevant data have changed.
3. *An object store* contains the combined representations of the agents and entities within the simulation.
4. *A translation and aggregation layer* is responsible for managing an array of data conversions between the models and the object store.

The various models incorporated into UrbanSim encompass a variety of processes and actors related to the urban environment. Apart from determining the behaviour of these processes and actors, each model also creates a range of object types, as well as the fields of these objects, on which they operate. A model may also choose to share fields created by other models, which represents one technique for data-level integration and coupling through the object store [58]. In addition, models have the capacity to create new object types that contain domain-specific data which did not exist in the past. A water quality model may, for example, declare a nutrient load value. Models may also monitor sets of object types and fields throughout the simulation [87]. All models are responsible for indicating how often they have to be executed.

Models do not have the ability to communicate directly with each other. They rather interact through shared data stored in the object store that has been mediated by the translation

and aggregation layer [58]. The architecture encourages system evolution, simplifies the process of replacement of models by updated versions, and makes it easy to create and add new models. The object store allows models to define and share common objects that are used by all models, thus creating a convenient platform for synchronising all the model actions. The architecture also allows for an automatic large-scale conversion of an array of data through the translation/aggregation layer which promotes model integration [87].

The ultimate driver for this type of architecture was the pursuit to remove software complexity from the flexible individual models and rather to transfer complexity into the more rigid supporting architecture. This is advantageous because the supporting architecture only has to be written once and can be developed by a highly skilled expert programming team, while allowing for the models to be changing frequently [58]. A visual representation of the UrbanSim architecture is presented in Figure 3.2.

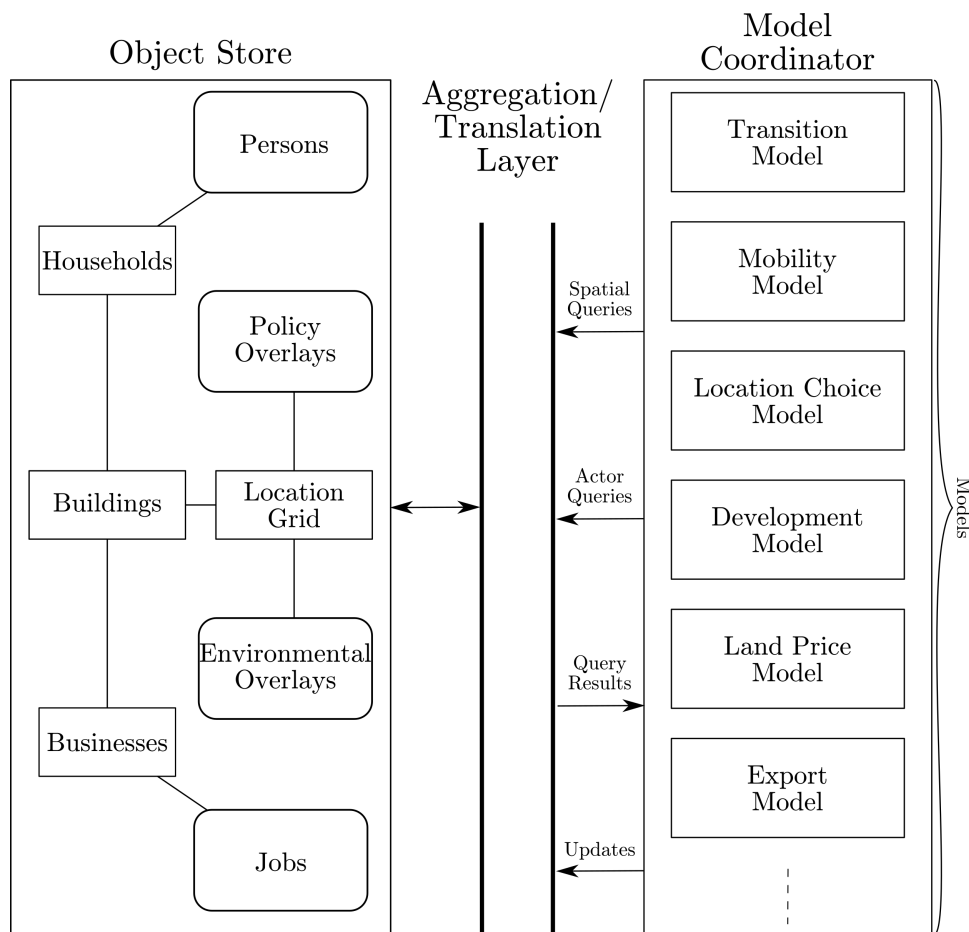


FIGURE 3.2: Graphical representation of the UrbanSim architecture [58].

3.3.3 The UrbanSim model structure

UrbanSim makes use of two key exogenous inputs. The first is a set of predicted future macroeconomic circumstances, including indicators such as population and employment per sector, produced by an external macroeconomic model. The second is a set of predicted travel conditions, including composite utilities and congestion times of travel between different zones, obtained from an external travel demand model [84]. The travel demand model is relatively

coupled with UrbanSim as land use predictions are used as input to the external travel demand model. This means that the travel conditions are subsequent to the annual iterations of the UrbanSim land use model [87].

Normally UrbanSim would schedule its models to be executed once per simulated year, as may be seen in the data flow diagram in Figure 3.3 [87]. The data store stores the current conditions of all the objects within the system, archiving the necessary models or requesting appropriate information files from the user.

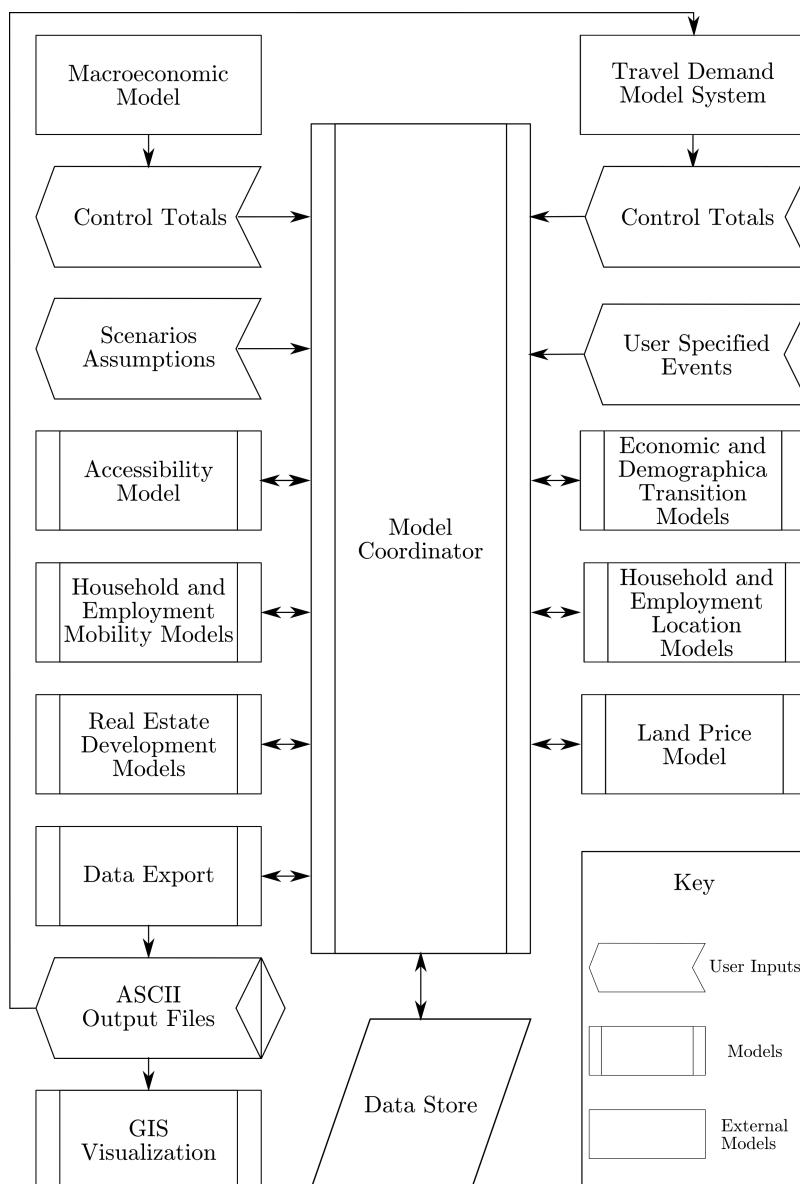


FIGURE 3.3: *UrbanSim software data flow* [84].

Apart from reading in exogenous input data obtained from external models, UrbanSim also takes input information from user-specified inputs. This may range from assumptions reflecting land use policies that regulate real-estate development to any user specification that represents changes in employment, real-estate development or land policy the user wishes to incorporate into the simulation [84]. The base models within the UrbanSim architecture include employment and household mobility models, employment and household location choice models, a land price model, a real-estate development model, an accessibility model, and economic and demographic

transition models [58]. The output of the simulation is then provided in user-specified formatted output files so that it can be scrutinised and further analysed [87]. An example of an output file format may be a GIS shape file or a text file containing time-stamped travel data.

The accessibility model

Monocentric, spatial integration models usually choose workplaces in an exogenously specified manner and residential locations on the basis of commuting to city amenities or these predetermined workplaces. The accessibility model of UrbanSim, however, does not work in this manner, but deals with accessibility according to a more general framework [83]. It views accessibility as a normal attribute like any other positive attribute of housing that a household would take into account when considering choosing a location in which to stay. It is therefore expected that consumers find value in access to workplaces and shopping amenities in addition to many other attributes, although households do not homogeneously value accessibility [87]. A retired person may, for example, be influenced less by accessibility to job opportunities when considering a location to move to.

UrbanSim quantifies accessibility of a location as the distribution of opportunities weighted by the composite utility of all transport modes to that location, defined as a logsum obtained from the travel mode choice model for each original-destination pair [83]. The measure of access for location i is therefore

$$A_i = \sum_{j=1}^J D_j e^{L_{aij}}, \quad (3.1)$$

where the D_j denotes a measure of activity in location j and L_{aij} is the composite utility for households that own vehicles of level a , moving from location i to j .

The accessibility model receives as input an utility matrix from the travel model as well as the land use distribution for a particular year. It then generates accessibility indices for subsequent use by the household and business location choice models [87]. The ultimate goal of the model is therefore to summarise the accessibility from each zone to various activities and amenities for which access is considered a priority in the household and business location choices.

Whereas UrbanSim operates on an annual basis, its travel model is likely to update less frequently because travel utilities remain relatively constant and are only replaced when a new simulation run's results replace the previous utilities. Despite travel utilities remaining constant for the years between travel model execution, the activity distribution in the accessibility indices is updated annually so that these indices change between years to simulate the changing spatial distribution of activities. The literature states that there is little basis for assuming that the effects of major transport development projects, such as railway expansions, would instantaneously be reflected in the surrounding land use [87]. In reality, the response of land use would occur over several years. As a result, UrbanSim only applies the travel models when major transport system changes occur since it is a computationally heavy model and its changes occur slower than other values in the simulation. The accessibility model, however, updates the activity levels often by being executed annually [58].

The economic transition model

The economic transition model forecasts the creation and loss of jobs [84]. The state of employment is user-specified and is typically grouped into ten to twenty sectors based on a number of underlying local economic sectors. External forecasts are employed in respect of economic

activity and sector employment in UrbanSim. These forecasts are often obtained from state forecasts or commercial in-house sources. The economic transition model integrates these forecasts into UrbanSim by computing the sector growth or decline from previous years. This is obtained by removing jobs from the UrbanSim database in declining sectors and adding jobs for growing sectors. In declining sectors, the probability that a job will be removed is proportional to the spatial distribution for the particular jobs sector. In this case, the removed jobs vacate occupied space and the space becomes available, joining a pool of other vacant spaces that may later be occupied by other jobs in the location choice component. This allows for the accounting of occupants, land and structure to remain up to date [87]. In cases of job creation, the new jobs are added to a database, assigned a null location initially and are subsequently allocated locations by the employment location choice model [58]. The economic transition model works as follows. First, the number of jobs to be added or removed is calculated. During this step, the total number of job changes in sector s from year $t - 1$ to year t is

$$\Delta J_{st} = C_{st} - |J_{s(t-1)}|, \quad (3.2)$$

where C_{st} denotes external data on the total employment in sector s during year t , whilst $J_{s(t-1)}$ denotes the set of jobs in sector s during year $t - 1$ [87]. Therefore, J_{st} is calculated either by removing jobs from the previous year's job set, or alternatively by adding jobs from the previous year's job set. Hence the job set $J_{st} \subset J_A$ in sector s during the year t is

$$J_{st} = \begin{cases} J_{s(t-1)} \cup F_{st} & \text{if } \Delta J_{st} > 0, \\ J_{s(t-1)} & \text{if } \Delta J_{st} = 0, \\ J_{s(t-1)} \setminus F_{st} & \text{if } \Delta J_{st} < 0, \end{cases} \quad (3.3)$$

where J_A denotes the set of all jobs in the job universe and F_{st} is the flux of jobs in sector s during year t . The flux of jobs is the set of all the jobs that are removed or added to a particular sector during a certain time-stage. During job creation, the new jobs are taken from the job universe and added to the set of jobs in the model at the time-stage in question. The flux is therefore

$$F_{st} = \begin{cases} \{j \in J_A \mid j \notin J_{st} \text{ and } j \text{ is in sector } s\} & \text{if } \Delta J_{st} > 0, \\ \emptyset & \text{if } \Delta J_{st} = 0, \\ \{j \in J_{st}\} & \text{if } \Delta J_{st} < 0, \end{cases} \quad (3.4)$$

subject to the constraint

$$|F_{st}| = |\Delta J_{st}|. \quad (3.5)$$

The quantity in (3.5) constrains the cardinality of the job flux to be the same as the absolute value of the number of changes in jobs available.

The set of jobs that have not been allocated at time-stage t is denoted by U_t , which is initialised to the empty set for the base year. During the years that follow, U_t is populated by all the unallocated jobs from the previous year. That is,

$$U_t = \begin{cases} \emptyset & \text{if } t = 0, \\ U_{t-1} & \text{otherwise, with updates on allocated jobs included.} \end{cases} \quad (3.6)$$

Jobs are created without a location and are added to a set of unallocated jobs. This set forms part of the employment location choice model. For each sector s ,

$$U_t \leftarrow U_t \cup F_{st} \text{ if } \Delta J_{sr} > 0 \text{ with updates on allocated jobs included} \quad (3.7)$$

or else

$$P_t \leftarrow P_t \setminus \{(j, \ell) \in P_t \mid j \in F_{st}\} \text{ if } \Delta J_{sr} < 0 \text{ with updates on allocated jobs included,} \quad (3.8)$$

where P_t denotes the set of pairs (j, ℓ) of the form job j at location ℓ during time-stage t . Furthermore, the newly vacant locations are placed in a set

$$V_t = \{\ell \in L_t^J \mid \forall j \in J_t, (j, \ell) \notin P_t\} \text{ with updates on allocated jobs included} \quad (3.9)$$

of vacant locations at time-stage t .

The demographic transition model

The demographic transition model is responsible for simulating births and deaths in household populations over time [84]. The model employs an algorithm akin to the employment transition model explained above. Changing demographics require complex modelling and in reality is a result of a combination of an array of factors such as ageing, divorce, household formation, household dissolution, mortality, migration, changes in housing size and changes in income, to name but a few [87]. These data are not always readily available, and so simulating this phenomenon requires exogenous control totals [58]. User inputs according to local demographic analyses are used to construct these control values according to the distributions of income, household size and age of household heads [87]. The control totals are used to approximate the net changes in the number of households after a simulated year.

Similar to the economic transition model, newly created households are placed in limbo by adding them to a movers list. The household location choice model then allocates the households to various vacant residential units [58]. Newly deleted households are, however, reflected by removing households from the housing stock and by accounting for vacancies in residential units created by them. This model is mathematically similar to the above-mentioned economic transition model [84].

The employment mobility model

The employment mobility model predicts the probability of each job per type moving to a new location during a particular year [58]. The model reflects changes in the job market such as lay-offs, business relocations, business foreclosures and job turnover by employees. As in the above-mentioned economic transition model, when job losses occur in dwindling sectors, the model assumes that the probability of moving is proportional to the spatial distribution of jobs per sector [87]. Moreover, when job losses occur, the model removes jobs per sector from the buildings they currently occupy, in addition to indicating that these buildings are now vacant. The jobs are then added to an unallocated pool of new jobs per sector which is calculated in the economic transition model [58]. The new and moving jobs serve as a combined pool that is allocated in the employment location choice model.

The choice of mobility rate is treated as an independent choice and the probabilities are predicted through annual mobility rates observed over a recent period per sector [87]. These rates are calculated from historical data on longitudinally linked business establishment files [58, 87].

A set of jobs in sector s during year t , denoted by M_{st} , is selected to be moved based on a Monte Carlo sampling process denoted by $P(j, t)$ for job j during year t using the annual mobility rate of sector s [87]. A random number ranging between 0 and 1 is generated and compared with the cumulative probability of each possible outcome. The outcome that contains the random number within its cumulative probability bin is then selected [87]. In cases of only two possible outcomes, an outcome is selected based on whether the random number generated is above or below the mobility probability.

The Monte Carlo sampling process $P(j, t)$ is applied to determine whether job j will move during time-stage t . The jobs that move are then added to the unallocated job pool by means of the assignment

$$U_t \leftarrow U_t \cup M_{st} \text{ with updates on allocated jobs included} \quad (3.10)$$

for each sector s and are excluded from the job location pool by the assignment

$$P_t \leftarrow P_t \setminus \{(j, i) \in P_t \mid j \in M_{st}\} \text{ with updates on allocated jobs included} \quad (3.11)$$

for each sector s . Subsequently, the previously occupied job locations are added to the vacant job location pool by means of the assignment

$$V_t \leftarrow \{\ell \in L_t^J \mid j \in J_t, (j, \ell) \notin P_t\} \text{ with updates on allocated jobs included.} \quad (3.12)$$

The household mobility model

The household mobility model is responsible for simulating the movement of households from their current residential units [58]. The model uses the same algorithm as the above-mentioned employment mobility model, but the model employs coefficients calculated for each household type. Census data are used to estimate the mobility probabilities [87]. The model reflects differential mobility rates for owners and renters during different life stages.

The model is applied by first removing the mover-households per type from the household stock in each zone. The mover-households are then added to a pool containing all new households per household type, which is populated by the demographic transition model. The households in the newly combined pool are then allocated to vacant residential units by a household location choice model [87]. In addition, the vacancy of the residential unit is updated, thereby rendering the residential unit available for occupation by the household location choice model.

3.3.4 Location choice models

Location choices occur often within an urban environment. UrbanSim models location choices for households, employment and real-estate development as discrete choice models. The design of its location choice model draws inspiration from the *Random Utility Maximisation* (RUM) framework and a mathematical model, called the *multinomial logit model*, both initially introduced by McFadden [55].

UrbanSim draws from the RUM framework in order to assess the utility of various location alternatives. According to the framework, each agent chooses from a set J of alternatives, where each alternative $j \in J$ is associated with a utility

$$U(j) = u_j + \epsilon_j \quad (3.13)$$

comprising a systematic element u_j and a random element ϵ_j . The systemic element is a function of the form $u_j = \beta \cdot \mathbf{X}_j$, where β denotes a vector of estimable coefficients and \mathbf{X}_j denotes a vector of externally observed variables related to location alternative j that the agent takes into account and interacts with when making a decision [55]. The unobserved random element ϵ_j assumes a Gumbel distribution.

The location choice model in UrbanSim further conforms to the *multinomial logit model*. The latter requires an associated utility for each alternative, and so the systemic element u_j of the RUM framework is incorporated into a function of the form

$$P(j) = \frac{e^{u_j}}{\sum_{j \in J} e^{u_j}}, \quad (3.14)$$

where $P(j)$ denotes the probability of location alternative j being selected [86]. The vector of coefficients β is calculated by applying the maximum likelihood method which, in turn, is employed to calculate the utility u_j within the multinomial logit model in (3.14).

UrbanSim, however, applies a slightly altered version of the RUM framework within its location choice model. In particular, a set K of sub-models is adopted for each variation of the location choice model (household, employment or real-estate development location choice models). Each sub-model $k \in K$ employs of a set N of decision makers (agents). Each agent $n \in N$ considers each alternative location j when making a location choice. While UrbanSim calculates utility values similarly to the RUM framework, the coefficients β exhibit an additional dimension j . More specifically, UrbanSim considers a set I of features when calculating the utility of each location alternative j . Each feature $i \in I$ has a unique coefficient and variable value in relation to each location alternative [86]. Therefore, both the coefficients and the variables have the added dimension i , resulting in two matrices β_{ji} and X_{ji} containing the coefficients and the variables of each location j with respect to each feature i .

Utility operations are implemented as function evaluators of the form $U(j)$ within UrbanSim's location choice model [85]. This mathematical process is illustrated in Figure 3.4, whereas the associated computational process is illustrated in Figure 3.5.

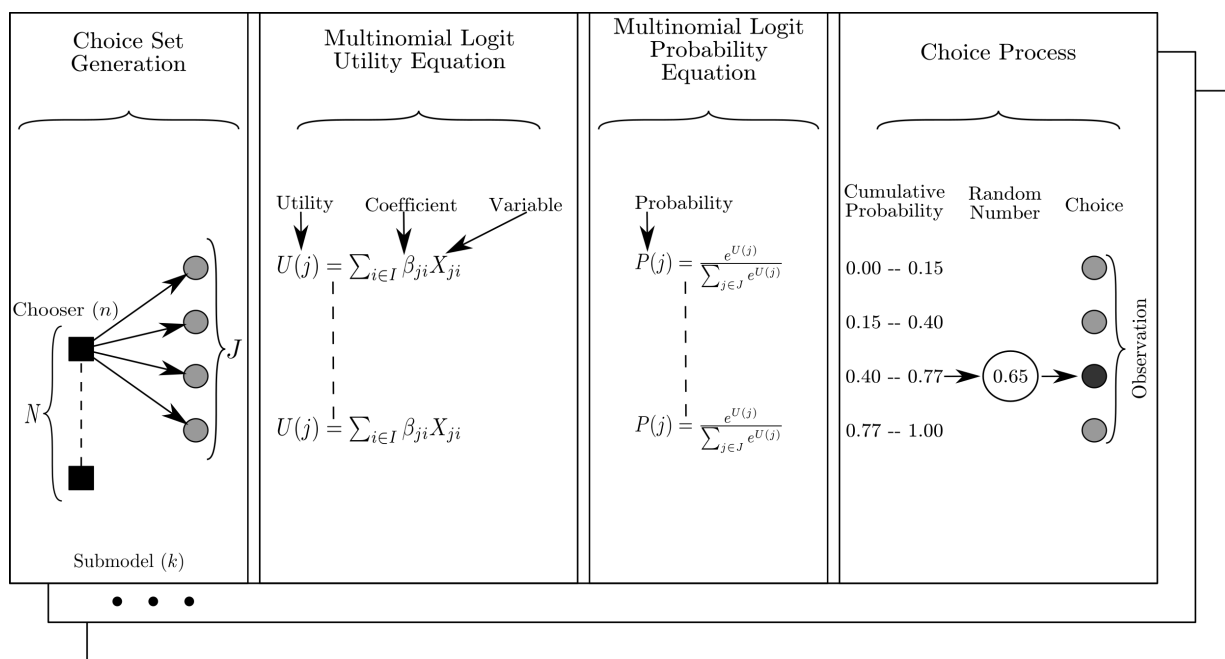


FIGURE 3.4: *UrbanSim location choice model specification [85, 88].*

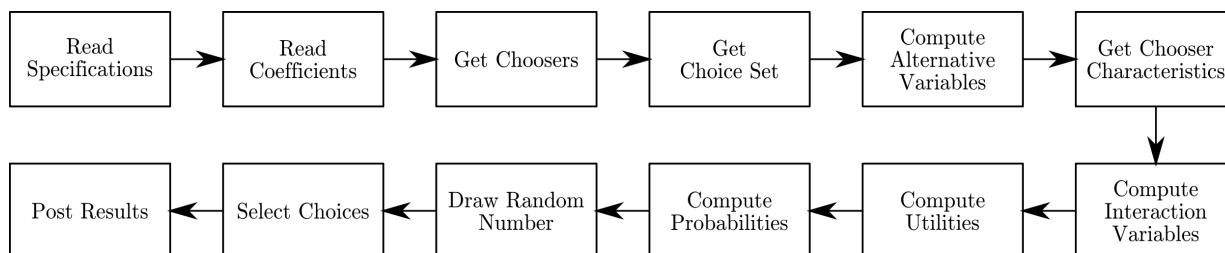


FIGURE 3.5: *UrbanSim location choice model computational process [85, 88].*

The model initially receives relevant specifications and data as input. These data are used to estimate the coefficients β_{ji} by applying the maximum likelihood method, as mentioned above. The set N is established and is managed as a mover queue in a random order, after which the choice set J is constructed to contain location alternatives for each agent $n \in N$. The set N is constructed through a random sampling of alternative options according to the Monte Carlo method [85]. The samples are weighted by the number of location opportunities available, such as vacant houses in the household location choice model or available jobs in the employment location choice model. The number of location alternatives considered in the simulation is user-specified.

When an agent is selected and a set of alternatives is sampled during choice set generation, as illustrated in Figure 3.4, the relevant variables, \mathbf{X}_{ji} , are computed [85]. The variables are utilised within the logit calculations in order to calculate the utility $U(j)$ for each alternative j [86].

Hereafter, the probability $P(j)$ is calculated according to the multinomial logit utility expression in (3.14), as illustrated in Figure 3.4. These probabilities are binned within a cumulative probability range which is used to make the final decision [85, 87]. As illustrated in Figure 3.4, a random number is generated between 0 and 1 during the choice process, and a final choice is selected if the random number falls within the cumulative probability bin allocated to the specific choice alternative.

The employment location choice model

The *employment location model* is responsible for simulating the process of job creation and job movement by assigning unallocated jobs created by the *economic transition model* and *employment mobility model* to certain locations [58]. This allocation of jobs is modelled by employing a location choice model. The number of job opportunities per area represents location opportunities when weighting location alternatives during the construction of the choice set J for each chooser $n \in N$. This is estimated by considering the total area of the available non-residential space within the area considered. The model allows for the possibility that certain job are more compactly located, such as in office blocks, by applying a specific jobs per square metre rate for each employment type [87]. The possibility of home-based employment is also accommodated in the model by randomly selecting a sample set of spaces in residential units as possible home-employment locations [84]. The model estimates the number of job opportunities per area as

$$|L_t^J| = \frac{s_\ell}{r_{sd}} + \frac{h_\ell}{r_{hd}}, \quad (3.15)$$

where the total non-residential area in location ℓ is denoted by s_ℓ and the total number of housing locations in location ℓ is denoted by h_ℓ . Furthermore, the rate of job opportunities per area is denoted by r_{sd} in (3.15) for development type d in the form of area per job, and finally the rate of home-employment is denoted by r_{hd} , which is the minimum units required per job for development type d .

Each sub-model $k \in K$ simulates the location choice for a different type of job, thereby accounting for different variables, depending on the job type when assessing the utilities of location alternatives. UrbanSim draws from urban economics when deciding which variables to consider during the computation of the utility $U(j)$ of each location alternative. The model would, for example, make the assumption that increased accessibility to higher income customers would result in a higher probability of retail and service jobs being created [87]. The two major influences the model accounts for in location utilities are localisation economies and inter-industry

linkages. The former refers to positive external factors that attract other firms in the same industry. The underlying assumption is essentially that anything in the area that allows for lower cost of production results in a higher concentration of similar industry firms. The latter refers to the concentration of strategic businesses from different industries that allow for better access to beneficial business partners. The model also accounts for many other factors that influence the agent in respect of employment location. Some of these generic factors include [87]:

Age: The model accounts for the age depreciation of buildings and makes the assumption that businesses prefer newer buildings, therefore discounting bids for older buildings. This factor represents deterioration of buildings, as well as changes in preferences and architectural style.

Density: Certain industries tend to migrate towards more or less densely populated areas, depending on the nature of the business. Manufacturing would, for example, discount an option that is located in a high-density area, accounting for the fact that more land is typically required. Retail businesses, on the other hand, tend to prefer higher density areas in a bid to increase accessibility to customers. Of all the sectors, the service industry holds higher density in the highest regard and tends to outbid other industries for these options.

Amenities: Certain sectors tend to value locations that are closer to major roads higher than others. The retail sector, especially, would typically outbid other sectors for major road accessibility. This is usually measured by distance in metres as well as travel time to the CBD.

All the independent variables considered in the employment location choice model may be grouped into four major categories [84]:

1. *Real-estate characteristics* allude to factors that describe the actual real-estate, such as price, development type, land use mix and density.
2. *Regional accessibility* refers to accessibility to the population and important areas such as the CBD or airports.
3. *Urban design-scale* has a larger focus on local accessibility to important amenities and means of transport, such as highways and arterials.
4. *Local agglomeration economies within and between sectors* indicate possible synergies between different sectors with a view to boost business.

After the employment location choice model has allocated a job to an employment location, the vacancy status of the employment location is updated and it is no longer available for future job allocations. The assigned job is also subsequently removed from the unallocated jobs list [87]. This simulation update is typically executed on an annual basis.

The household location choice model

The *household location choice model* assigns unallocated households to vacant locations. The unallocated households are drawn from a *demographic transition model* simulating the creation and demise of households, and a *household mobility model*, simulating the movement of households. Similar to the employment location model, this model is a location choice model. Each

unallocated household considers a sample of random locations when constructing the choice set J for each chooser [58]. Vacant housing is used as location opportunities to weight locations when the set is created. An urban society typically consists of many different types of households, each valuing certain property location aspects more than others. The model accommodates this by being stratified into a set K of sub-models, differentiating between household types. There are various ways to classify different household types, depending on the urban society to which the model is applied. Households may be classified according to income, household size or workers, to name but a few [86]. The model proceeds to estimate the desirability of each alternative for a certain household by calculating a location utility $U(j)$ [87]. The desirability of each alternative j is used in the multinomial logit model to simulate the decisions made. In the book, *Urban Economics and Real Estate Markets* [16], it is argued that housing has a positive income elasticity on demand. Therefore, as income increases, households tend to spend a part of the increase on more expensive housing. This results in new housing, typically having more amenities available [16]. UrbanSim uses this hypothesis to justify the assumption that households typically prefer moving to higher income neighbourhoods. Subsequently, UrbanSim tends to simulate new housing being built for more affluent households, resulting in the wealthy moving to the new housing and leaving vacancies behind them which are, in turn, occupied by less wealthy households [87]. This chain continues until the least attractive dwellings are abandoned.

The household location choice model employs variables identified by theories in urban geography, urban economics and urban sociology to determine the utility of a location. Generic factors influencing location decisions include [87]:

Accessibility: The model utilises a trade-off between transport access and land cost. The model accounts for travel time to the CBD, employment opportunities and shopping services.

Density: A net density measure is used to determine the input-substitution effect of land and capital. Therefore, land with high accessibility will exhibit increased prices, upon which builders substitute capital for land and develop denser accommodation. This results in households valuing land over accessibility and subsequently choosing larger plots with less accessibility.

Age: The model accounts for the age depreciation of buildings as the lifespan of a building is finite. Furthermore, changes in preferences are accounted for based on the assumption that wealthier households prefer newer houses (exceptions to this assumption are also taken into account where historically significant housing is preferred by the wealthy).

All the independent variables considered in the household location choice model may be grouped into three major categories [84]:

1. *Housing characteristics* refer to factors describing the residential units and include features such as price, development type (land use mix, density) and age of real-estate.
2. *Regional accessibility* refers to accessibility to employment opportunities and amenities.
3. *Urban design-scale* alludes to characteristics describing the neighbourhood, including features such as neighbourhood land use mix and density neighbourhood employment.

Following the allocation of an unassigned household to a residential unit, the vacancy status of the residential unit is updated and it is no longer available for future household allocations [87]. That is, the assigned household is removed from the unallocated household list.

The real-estate development model

The *real-estate development model* simulates the construction of new real-estate and the redevelopment of existing real-estate units [84]. The model classifies all parcels within a zone on which development is allowed according to the real-estate composition of the parcel [58]. The major driver of this classification is the number of housing units in a parcel. Parcels with a majority of non-residential area are categorised as non-residential and *vice versa*. Additionally, the model accommodates a mixed-use classification of parcels if the prominence of both residential and non-residential areas are substantial enough.

Annually, the model iterates over all the categorised real-estate parcels in the set N and creates a set J of possible real-estate transition alternatives for each parcel $n \in N$ to which it can be transformed (simulating real-estate development), including the option of not developing [58, 88]. As opposed to the employment and household location choice models, which treat households and employees as agents, the *original version* of this model has opted to treat the parcels of land as agents. The desirability of each real-estate alternative is estimated and employed within the multinomial logit calculations carried out in the location choice model. The input data considered by the model for coefficient estimation has a strong focus on building age and is preprocessed on a parcel level [62]. The model identifies historical construction events within a user-specified time stage and in which zones these constructions take place. The events are identified by analysing changes in building ages within a zone. Values of specified independent variables in the zones associated with historical development events are looked up and used to estimate the coefficients β_{ji} [88]. Thereby, a set of development alternatives, representing transitions between two development types during a simulated year, is created and the relevant coefficients are estimated.

The generic multinomial logit process is again followed to calculate the utility and, subsequently, the probability of each development alternative. Thereafter, a Monte Carlo sampling process is employed to simulate the agent's commitment to each development [88]. UrbanSim uses a development template to simulate development. This template is used to identify which would be the most likely characteristics of real-estate development, such as the number of housing units, the construction schedule and improvement value, to name but a few [87, 88]. The development commitment is then placed in a queue of development events to take place.

Moreover, similar to the household location choice model, the real-estate development model estimates the utility of development alternatives by carrying out multinomial logit computations based on an array of variables. The variables that influence the decisions made by developers can be grouped into the following categories [84]:

1. *Site characteristics* describe the characteristics of the environment of a location. This includes existing development characteristics, land use plans and environmental constraints, to name but a few.
2. *Urban design-scale* alludes to urban factors considered, such as proximity to highways and arterials, proximity to existing development, neighbourhood land use mix, property values and recent developments in the neighbourhood.
3. *Regional accessibility* indicates basic accessibility, such as access to population and employment, as well as travel time to CBD or airport.
4. *Market conditions*, which primarily relate to vacancy rates.

The land price model

The final generic model is the *land price model*. This model estimates the evolution over time of land prices for each zone or parcel as the characteristics of the land change [58]. The simulation assumes that the value of a location is capitalised into the price of land according to urban economic theory [84]. Historical data are used in a hedonic regression to incorporate the effect of various characteristics on the price of a piece of land.

Land prices match supply of and demand for land in respect of different real-estate development types. The model computes relative market valuations for development attributes, location and non-residential spaces. Development prices influence location choices with respect to jobs and housing. An adjustment in these prices therefore has an impact on location choice alternatives [87]. Generally, higher priced options will be favoured by decision makers with lower price elasticity of demand. A change in prices also impacts the attractiveness of a location to real-estate developers. UrbanSim simplifies computations in the land price model by making the following assumptions [58, 87]:

- Developers, households and businesses accept the price presented to them as price takers. Also, market adjustments are made in response to supply-demand relationship changes, taking into account price information from previous periods.
- Supply-demand imbalances and location attractiveness is priced into the value of land. The value of buildings only represents building replacement costs and variation in development costs, taking into account terrain and the nature of different development types.
- A long-term structured vacancy rate for each development type is assumed, as well as an adjustment of current vacancy rates to these long-term rates, and this influences the price. Therefore, if the current vacancy rate of a unit falls below the long-term vacancy rate, the price level would rise.

The price of land incorporates a variety of attributes, some of which are the same as those of the real-estate development model [84]. Land-use mix, density and proximity to arterial roads are some of the many characteristics taken into account [87]. The hedonic regression incorporates all the characteristics to determine the price of land per area as

$$P_{ilt} = \alpha + \delta \left(\frac{V_i^s - V_{it}^c}{V_i^s} \right) + \beta X_{ilt} \quad (3.16)$$

for development type i at location ℓ during time stage t . Here V_{it}^c denotes the vacancy rate during time stage t for development type i , where local and regional vacancy is weighted. The long-term vacancy for development type i is denoted by V_i^s . Furthermore, the regression incorporates a set of location and site attributes X_{ilt} and parameters α, δ and β , all of which are estimated. The prices are updated annually according to market and construction activity [87], and the year-end prices are then used as references for market activities during the following year. The variables used in the model may be classified into four general categories [58, 87]:

1. *Site characteristics* describe the actual environment of the location considered, such as environmental constraints, development types and land use plans.
2. *Urban design-scale* alludes to the urban make-up of the location considered, including factors such as density, land use mix and proximity to highway and arterials.

3. *Regional accessibility* encompasses general accessibility characteristics, mainly focussing on access to employment and the population.
4. *Market conditions*, which mainly constitute vacancy rates.

3.3.5 UrbanSim validation

UrbanSim is continually being updated and refined, and is currently a much-improved version upon its initial release, designed by *Waddell* in 1998 [81]. The software was first applied to Eugene-Springfield, Oregon in the United States and has since been applied to numerous other metropolitan areas around the globe, such as Honolulu, Houston, Paris and Brussels, to name but a few [84]. It has been validated longitudinally by comparing 15 years of path-dependent simulations to observed outcomes in these metropolises [85]. The software has been used by researchers from over 70 countries to assist in policy scenario decision making.

3.4 Chapter summary

The objective of this chapter was to provide the reader with a basic insight into the urban simulation environment and the tools available as stated in §3.1. The evolution of urban simulation tools was illustrated by discussing a number of ITLUMs from the literature in §3.2. This was followed by an in-depth review, in §3.3, of an ITLUM relevant to the topic of this thesis, namely UrbanSim. Further elaboration on UrbanSim was provided in the form of a thorough discussion on location choice models within the UrbanSim environment in the same section.

CHAPTER 4

Urban policy scenario optimisation framework

Contents

4.1	Framework overview	52
4.2	The preprocessing component	53
4.2.1	<i>Scenario area identification</i>	54
4.2.2	<i>Coefficient estimation</i>	56
4.3	Scenario creation and analysis	59
4.3.1	<i>Scenario creation</i>	59
4.3.2	<i>Scenario analysis</i>	62
4.3.3	<i>Fitness scoring</i>	64
4.4	The optimisation component	64
4.4.1	<i>Population initialisation</i>	64
4.4.2	<i>Selection</i>	65
4.4.3	<i>Crossover</i>	66
4.4.4	<i>Mutation</i>	66
4.4.5	<i>Replacement</i>	67
4.4.6	<i>Stopping criteria</i>	67
4.5	Results produced by the framework	67
4.6	Chapter summary	69

A novel framework for the optimisation of incentivisation strategies for real-estate development in prioritised urban zones, called the *Urban policy scenario optimisation model* (UPSOM) framework, is presented in this chapter. The chapter opens with a high-level overview of the framework in which the constituent components of the framework are described briefly. Thereafter, an in-depth description follows on the first framework component, known as the *preprocessing component*, during which the various steps and criteria related to the creation and analysis of policy scenarios are elucidated. In addition, a number of central concepts, such as the notions of a *policy scenario plan* and of a *policy element*, are described. The focus of discussion then turns to the *optimisation component* of the framework as well as the GA operators employed within this component. The nature of the results typically produced by the framework is then discussed, after which the chapter closes with a brief summary of its contents.

4.1 Framework overview

From the literature review of Chapter §4.2.2 it is known that ITLUMs are typically used to test the desirability of various possible urban policies. To the best of the author's knowledge, however, ITLUMs have not yet been optimised with a view to produce the best possible implementation versions of the policies considered. This capability is therefore pursued in the UPSOM framework proposed in this thesis in the sense that the framework attempts to optimise the implementation of urban policies so as to maximise residential real-estate development in prioritised areas.

A central underlying assumption of the framework is that real-estate developers assess the attractiveness of areas before considering development there. If a zonal area is considered to be more attractive than another area according to a real-estate developer, the likelihood that (s)he decides to pursue development in the former area is higher than that in the latter area. The attractiveness of an area is determined by a function of various influencing features. These features may each have a positive or a negative impact on the attractiveness of an area in question. When considering the implementation of policies, the goal is to maximise the attractiveness of areas in which densification is a priority. This is achieved by increasing the positive influencing features in these areas whilst decreasing negative influencing features.

A policy typically involves improving at least one positive feature in an area and is crafted by targeting changeable spatial features which positively impact the perceived attractiveness of an area from the perspective of real-estate developers. Policies may be implemented in various ways, thereby creating different scenarios for different policy implementations. These different scenarios are known as *policy scenarios*. The UPSOM framework attempts to optimise the implementation of a policy by generating a new optimal policy scenario which positively impacts the attractiveness of priority areas. The framework is generic in nature and may hence be applied in respect of various ITLUMs. The ITLUM employed is, however, required to produce probabilities of selecting each zonal for development on a disaggregated level. By converting data through the application of relevant preprocessing operations, and subsequently optimising possible policy scenarios iteratively, the UPSOM framework may assist urban policy makers to improve the implementation of these policies.

The UPSOM framework initialises by invoking a preprocessing component consisting of various operations. These operations may be partitioned into two different classes. Operations in the first class are aimed at identifying areas in which policies may be implemented. These areas are known as *scenario areas*. This is achieved by grouping together all areas within a certain radius around each priority area as candidate areas where scenarios may be implemented. Operations in the second class are concerned with the estimation of regression coefficients that quantify the attractiveness of an area in terms of the influencing features under consideration.

After having executed the preprocessing component, and thereby having produced the relevant scenario areas and feature coefficients as output, the optimisation component of the framework is invoked. This component attempts to optimise the collective attractiveness of all the priority areas from the perspective of real-estate developers interested in constructing a variety of residential real-estate units. The fitness function maximised is the aggregated probability of real-estate developers developing residential units in priority areas. By maximising this aggregated probability, as opposed to the actual choices of the agents, the component deterministically optimises a stochastic model. This allows the framework to bypass the typical obstacles which accompany stochastic models, such as requiring numerous iterations in order to ascertain the impact of alternative policy scenarios. This is especially beneficial when considering the computationally expensive nature of ITLUMs and metaheuristics. An ITLUM, such as UrbanSim, would typically calculate the probability of each zone in an area being selected by a developer agent by

considering the features which impact the attractiveness of the zone. Different implementation scenarios would result in different feature values for each zone and, therefore, generate different probability sets.

By applying a GA, the optimisation component iteratively searches for policy scenarios which maximise the aggregated probability of real-estate development in priority areas. After an optimised policy scenario has been computed, the policy scenario may be implemented and an ITLUM, such as UrbanSim, may be used to simulate the resulting urban growth in that area. A high-level overview of the UPSOM framework architecture is illustrated graphically in the form of a *data flow diagram* (DFD) in Figure 4.1.

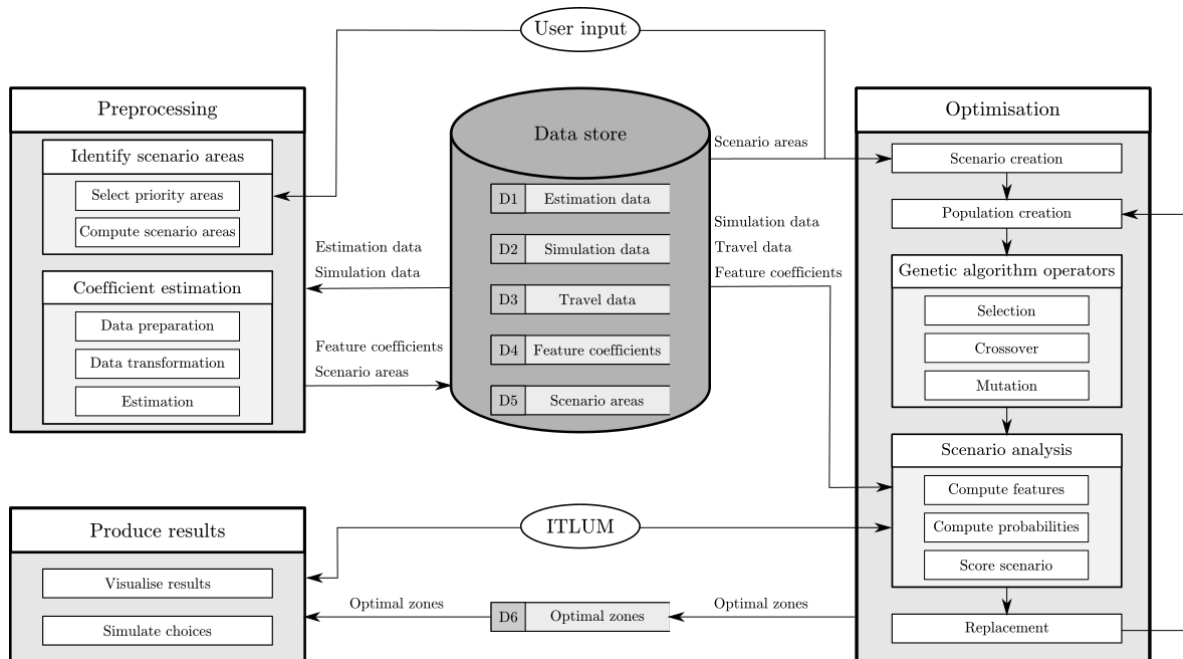


FIGURE 4.1: High-level overview of the UPSOM framework architecture in the form of a DFD.

All the input and transformed data are stored within the data store throughout execution of the framework. The preprocessing and optimisation components do not directly communicate with each other. Instead, the preprocessed data are stored in the data store again from which the required data are called by the optimisation component. The ITLUM in the framework is also detached from the other framework components and is utilised to compute the probability set during each optimisation iteration by considering different feature values for each policy scenario produced in the optimisation component.

The remainder of this chapter is devoted to an in-depth description of the two major components of the UPSOM framework depicted in Figure 4.1, namely its preprocessing and optimisation components. The nature of the policy scenarios generated within the optimisation component is also discussed, as well as how these scenarios are analysed and compared.

4.2 The preprocessing component

As mentioned above, the preprocessing component is the first component invoked within the framework and is required for data transformation. These transformed data are subsequently used in the optimisation component. The preprocessing component is elucidated in more detail

in the form of a DFD in Figure 4.2. The first part of the preprocessing component, responsible for identifying scenario areas, is further partitioned into two modules numbered 1 and 2 in Figure 4.2. These modules perform their respective operations successively in the order of their module numbers. The coefficient estimation part of the preprocessing component is executed in parallel with the scenario area identification part. The former part consists of three modules numbered 3–5 and are again executed in succession in order of their numbers. The preprocessing component draws input data from the data store and requires user input for its operations.

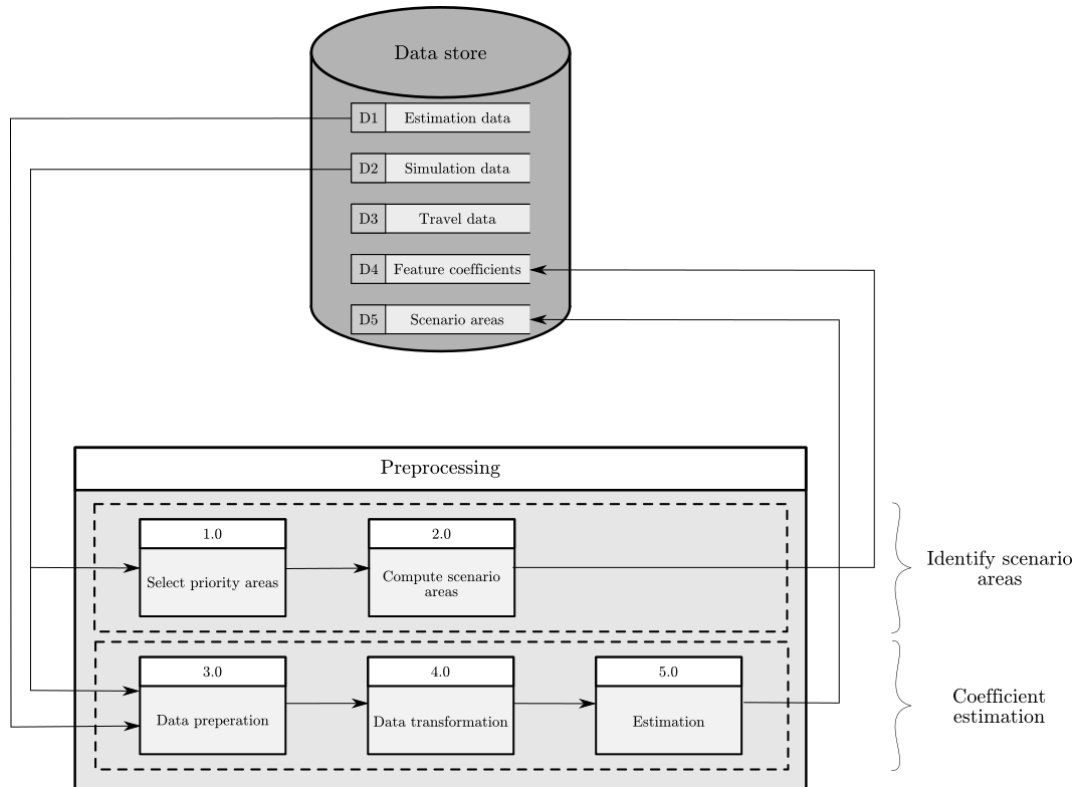


FIGURE 4.2: The preprocessing component of the UPSOM framework.

4.2.1 Scenario area identification

When urban policies are considered by policy makers, two facts must be known beforehand. First, the priority areas must be identified, and so it is assumed that the UPSOM framework has access to a set of areas labelled *a priori* as prioritised. These areas are typically identified by MPOs as areas which would benefit urban integration if their densities were to increase. Secondly, the framework user must know the nature of the candidate policy scenarios which are to be considered. The user is therefore assumed to be aware of the limitations imposed on policy scenarios. Consider, for example, a policy scenario involving the construction of arterial roads. In this case, the user should know which areas have the capacity to accommodate the construction of roads and which areas do not. If the priority areas are known and the capacity constraints of all the areas are known, the scenario area identification may proceed.

This part of the preprocessing component is required to scope the sample space of possible solutions. This is especially important when considering the size of major metropolitan census data sets and reduces the computational burden on the optimisation algorithm by discarding unnecessary zonal areas from the potential sample space.

This process requires data for the base year of the simulation, and is assumed to be the most recently available year's data. The simulation data represent the spatial composition of the area considered during the year for which the policy scenarios are to be created. This data set is typically large and comprehensive. Each entry in the data set represents a specific zonal area and is associated with a unique key, known as the zone ID. Each zonal area is also associated with multiple feature entries which describe the spatial features of that particular zone. These features may range from mean household income or number of schools to mean distance from a highway, to name but a few. One feature that is required is the geographic coordinates of the geometric centre of each zone. The longitude coordinates are denoted by the variable x and the latitude coordinates by y (both measured in degrees). The first module in this part of the preprocessing component involves the stipulation of priority areas within the simulation data set by the user of the UPSOM framework.

Before the second module may commence, the user is required to specify a radius (in kilometres), denoted by Θ , which is used to create a circular perimeter around the geometric centre point of each priority area. The distance between each pair of zones in the data set has to be approximated. This is achieved by converting the respective differences between the latitude and longitude values of the centre point of each pair of zones into kilometres. This step is simply required to scope the possible solution space, and so an approximation may be used to measure the distances between zonal areas. For the purposes of this module, an approximation of a difference of one degree in latitude is taken as $\Delta Lat = 111.32 \text{ km}$ while a difference of one degree in longitude is taken as $\Delta Long = 40075 \text{ km} \times \frac{\cos(Lat)}{360}$. After having computed a perimeter around each priority zone, all zonal areas with a geometric centre point residing within any of these perimeters, are grouped together. All these zonal areas are then produced as an output data set in the form of a scenario areas set, denoted by \mathcal{S} . The areas in which policy scenario implementations are subsequently considered are sampled from \mathcal{S} . The process of scenario area identification is illustrated in Figure 4.3.

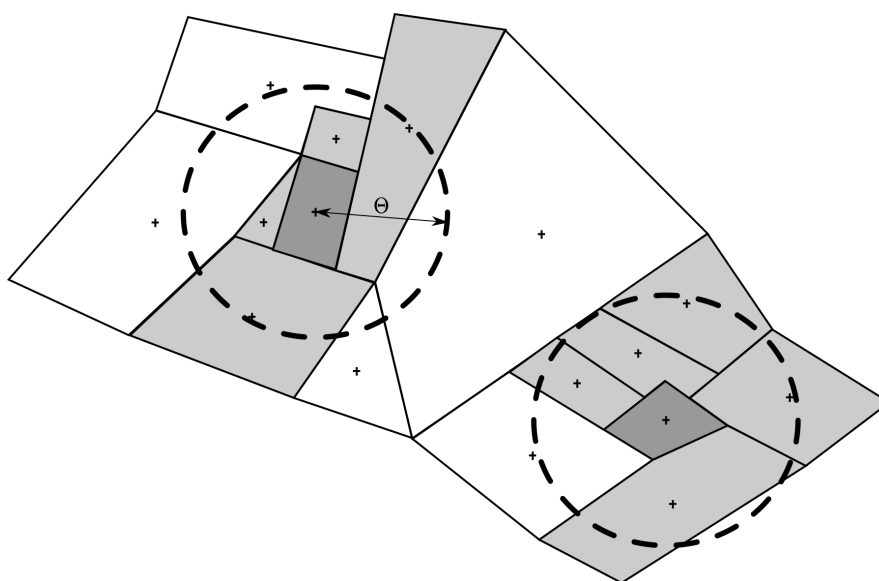


FIGURE 4.3: The process of selecting scenario areas.

4.2.2 Coefficient estimation

As mentioned above, real-estate developers assess the attractiveness of a potential area when considering development locations. The UPSOM framework employs an ITLUM to assess the attractiveness of each zonal area. Attractiveness is, however, a qualitative notion whilst ITLUMs are mathematical models which require quantitative values. As described in §2.1, an estimation of regression coefficients is required to measure the influences of features of the data set on the general propensity for real-estate development in certain areas. The estimation preprocessing step quantifies the features which describe the attractiveness of areas. The coefficients employed for this purpose describe how a real-estate developer may view a variety of features when considering alternative zonal areas. This estimation is a very important step and typically requires large data sets. These data sets have to be prepared correctly.

Data preparation

During the estimation of regression coefficients, historical estimation data are required. These data should include the features for which estimations have to be carried out. Ideally, the estimation data set should be large and detailed enough to describe the sentiment of developers within a geographic area.

Selecting a dependent variable for coefficient estimation is a very important step, as mentioned in §2.1. The nature of the data available is usually indicative of which variables are to be considered independent variables. Due to the fact that the UPSOM framework is aimed at promoting the construction of real-estate in certain areas, growth data sets are considered to be ideal for coefficient estimation. Therefore, the data preparation module combines estimation data with the simulation data of the base year. The estimation and simulation data are required to exhibit entries that are similar in nature and both must contain the feature values for which the coefficients are to be estimated. A visual representation of the nature of data for M zonal entries and N features is illustrated in Figure 4.4.

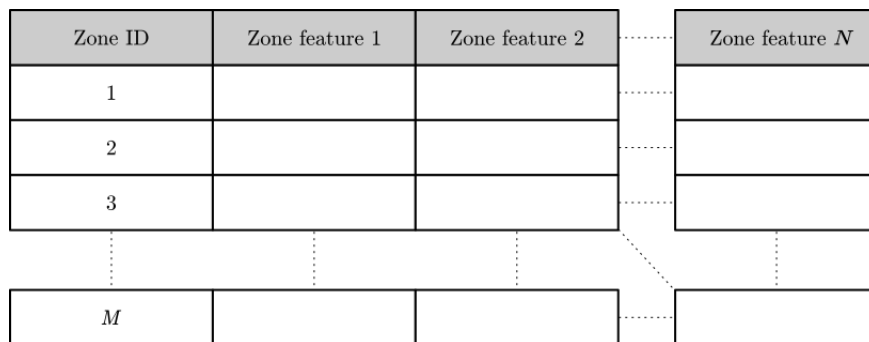


FIGURE 4.4: *The nature of estimation and simulation data.*

Data transformation

After having combined the estimation data and simulation data, these data are slightly transformed. This transformation is aimed at facilitating dependent variable selection. This decision determines how the dependent variable reacts to changes in the independent feature values. For the purposes of this framework, the relationship between the independent feature data and growth data is important.

In the data transformation module, an additional feature is computed and included in the estimation data set. This feature is computed in terms of other feature values which are already in the data set. The newly computed feature is the number of units of a certain type of residential real-estate in each zone. The number of real-estate units in the estimation data year is subtracted from the number of real-estate units in the simulation base year. This difference represents the growth of that particular real-estate type from the simulation base year date to the estimation data set date. The process is repeated for all residential real-estate types under consideration. These growth data are subsequently taken as values of the dependent variable in the estimation data set during coefficient estimation for each residential real-estate type.

As mentioned in §3.3.4, residential real-estate developments may be classified in numerous ways. For the purposes of the UPSOM framework, however, real-estate would typically be classified according to mean income. This is due to the fact that households with different income levels typically view the attractiveness of locations differently. Real-estate developers would also view the attractiveness of different locations differently, depending on the real-estate development type considered.

After having computed growth data for each real-estate development type, all the data preparation steps for the estimation preprocess are complete. In the estimation module, the growth data are taken as representing the dependent variable, while the feature values identified within the estimation data set are taken as representing independent variables. This is due to causation and correlation considerations. By considering the growth data as representing the dependent variable and the estimation feature values as representing independent variables, the estimation process takes into account the initial circumstances during the base year as the cause of growth of residential real-estate units during the period up to the subsequent estimation date. Due to the time-consuming nature of real-estate construction, it is assumed that the original feature composition in the estimation data has led to the subsequent growth that transpired within the various zonal areas (the growth data). The feature coefficients may then be estimated by employing the MLE method described in §2.1.1.

Coefficient estimation

The final module in the preprocessing component is responsible for estimating the regression coefficients. Initially, it should be clear to the user which features may be considered during estimation. There are various ways of selecting features and the UPSOM framework allows for any method of feature selection. The nature of feature selection is problem-specific and is typically limited by the data available. One method of feature selection involves consulting the literature on a set of potential features. Combinations of these features may then be tested empirically in order to ascertain which feature combination performs the best.

After the input data have been transformed and the features identified, coefficient estimation may commence by means of the MLE method. As discussed in §2.1.1, a statistical distribution must be assumed to govern the data. Due to the discrete nature of the dependent variable (the difference in the number of real-estate developments over a number of years), a PMF such as those of the Poisson or binomial distributions may be assumed for the observed data, as described in §2.1.3.

In the UPSOM framework, there is one dependent variable and a set of independent variables. These independent variables are each associated with an estimated regression coefficient. The coefficients are stored in a vector β , similarly to the coefficients described in the context of the location choice model in §3.3.4. The Poisson distribution, described in §2.1.3, is typically applicable to the data due to the probability of occurrence of each alternative being selected

during a simulated year typically being small. In contrast to the original PMF in (2.4), a conditional Poisson distribution (also known as Poisson log-linear model [24]) may be assumed, expressed mathematically as

$$f(y_j | \mathbf{X}_j) = \frac{\lambda_j^{y_j}}{y_j!} e^{-\lambda_j}, \quad (4.1)$$

where y_j denotes the value of observed data point j (dependent variable values which are independently distributed) and the parameter λ_j denotes the expected value parameter, given a discrete random variable. A total of k features are considered. A vector of values describing the various features for observation j is denoted by \mathbf{X}_j . Given the fact that λ_j is the expected value for observed data point j , it may be elaborated further as

$$\lambda_j = e^{(\beta_0 + \sum_{i=1}^k \beta_i X_{ji})}, \quad (4.2)$$

where $\beta_i \in \boldsymbol{\beta}$ denotes the coefficient for feature i and $X_{ji} \in \mathbf{X}_j$ denotes the value of feature i for the expected value of observed data point j . The coefficient vector $\boldsymbol{\beta}$ has to be re-estimated separately for each real-estate development type g . The log likelihood function (2.2) for this conditional PMF has to be maximised by adjusting the coefficients in $\boldsymbol{\beta}$. The likelihood function of a conditional poisson distribution with N observed data points is

$$L(\boldsymbol{\beta} | y_j, \mathbf{X}_j) = f(y_j | \mathbf{X}_j, \boldsymbol{\beta}) = \prod_{j=1}^N \frac{\lambda_j^{y_j}}{y_j!} e^{-\lambda_j} \quad (4.3)$$

and so the log likelihood function may be expressed as

$$\ell(\boldsymbol{\beta}) = \sum_{j=1}^N \ln \left[\frac{\lambda_j^{y_j}}{y_j!} e^{-\lambda_j} \right] = \sum_{j=1}^N y_j \ln[\lambda_j] - \sum_{j=1}^N \lambda_j - \sum_{j=1}^N \ln[y_j!] \quad (4.4)$$

and may be simplified when substituting (4.2) into (4.4) as

$$\ell(\boldsymbol{\beta}) = \sum_{j=1}^N \left[y_j \left(\beta_0 + \sum_{i=1}^k \beta_i X_{ji} \right) - e^{\beta_0 + \sum_{i=1}^k \beta_i X_{ji}} - \ln[y_j!] \right]. \quad (4.5)$$

When employing the MLE method, the goal is to maximise (4.5) by altering the values $\beta_i \in \boldsymbol{\beta}$. All the values of the variables y_j and X_{ji} are taken from the transformed estimation data set. The values of λ_j may be replaced by (4.2), which means that only the $\boldsymbol{\beta}$ values are unknown.

Thereafter, the differentiated log likelihood function has to be computed and set equal to zero. As discussed in §2.1.1, this produces the values of entries of the vector $\boldsymbol{\beta}$ for which the log likelihood is maximised. There are, however, numerous β_i values and so an analytical solution to the vector of equations obtained when equating the log likelihood derivatives to zero is not pursued. The roots of the equation left-hand sides have to be calculated in order to find the optimal coefficient values. This may be achieved by employing the Newton-Raphson method described in §2.1.2. This method is implemented in the UPSOM framework to find the root values for all k unknown variables in the vector $\boldsymbol{\beta}$. These values represent the best coefficient values for the features considered.

The estimation module is invoked for each residential real-estate development type considered in the set of real-estate development types, denoted by \mathcal{G} . The output data of the module is, therefore, is a vector of coefficients $\boldsymbol{\beta}$ which collectively describes the attractiveness of location alternatives for each real-estate development type $g \in \mathcal{G}$. These coefficients are employed by the ITLUM in the UPSOM framework in order to determine the respective probabilities of zonal areas being selected by the developer agent, as described for the location choice model in §3.3.4.

4.3 Scenario creation and analysis

The UPSOM framework has as objective encouraging real-estate development in priority areas by increasing the perceived attractiveness of the priority areas. Therefore, when policies are constructed, the coefficients estimated for the features in §4.2.2 must be consulted. The policies aim to increase the values of features with positive coefficients or and decrease the values of features with negative coefficients for the real-estate types considered.

Policy scenarios are constructed according to a *policy scenario plan*, which stipulates the high-level constituents of a policy, such as the objective of the policy and the various constraints associated with its implementation. These constraints are typically concerned with the allocation of *policy elements*. Policy elements are the entities which are locationally assigned to the scenario areas when a policy scenario is created. The policy elements which are assigned to zones depend on the features which are to be increased or decreased and may, for example, range from metres of road constructed, or the number of jobs created, to the amount of money used to subsidise property prices, to name but a few. The combination of policy elements allocated to scenario areas are vast and therefore, there are numerous policy scenarios which may be crafted from one policy scenario plan. Each different variation of the policy involves the creation of an alternative policy scenario. There are certain criteria that must be kept in mind when a policy scenario is created and these criteria typically pertain to the limitations of the user of the UPSOM framework and the data available to the user.

4.3.1 Scenario creation

When a user of the UPSOM framework generates a policy by consulting the regression coefficients estimated, certain criteria have to be considered. Feature values that are altered in the policies should be realistically changeable within a stipulated time-frame. An example of an unchangeable feature is a university campus. If the feature has a positive impact on the attractiveness of an area it is infeasible simply to create university campuses with the goal of increasing the density of real-estate development within a short time-frame. Typical changeable features include:

- *Household income.* By providing grants for households within a certain proximity to priority areas, the income of a household may be changed.
- *Road infrastructure.* The construction of roads is the responsibility of government bodies and the locations of their construction are changeable.
- *Job accessibility.* By creating employment in strategic areas, household accessibility to jobs may be changed. Furthermore, by improving transport services and infrastructure, household accessibility to employment from priority areas may be changed.
- *Real-estate price.* By subsidising the cost of real-estate in areas within a certain proximity to priority areas, the price of real-estate may be changed.

The extent of the anticipated change affected by policies has to be considered. This consideration is typically user-specific. The user has to be aware of budget constraints associated with various policy elements, and should therefore be informed in terms of the number of policy elements that may be assigned. The construction of roads may, for example, be associated with a prescribed budget limiting the length of roadways that may be constructed. The creation of jobs may,

however, experience a limitation on their number within an area, aimed at satisfying demand. The extent of variation of each changeable feature therefore has to be considered before a policy scenario plan is created.

The effectiveness of policy implementation is the final consideration. Subsequent to identifying the extent of feasibility of changeable features, the feature which is expected to have the largest impact on attractiveness should be considered. This feature would typically involve numerous policy elements (which have an effect on the feature) that may be assigned and are associated with large positive coefficients or small negative coefficients. Following the identification of suitable changeable features for policies, a policy scenario plan has to be constructed, as illustrated in Figure 4.5. The policy scenario plan is a high-level blueprint of the type of policy that is to be implemented and it typically includes:

- A budget constraint associated with the policy, denoted by b (an upper bound on the number of policy elements that may be allocated in total).
- The set of scenario areas to which the policy elements may be allocated, denoted by \mathcal{S} .
- Capacity constraints on the number of policy elements associated with each area within the set of scenario areas.
- Implementation constraints associated with the policy elements (lower bounds on the number of policy elements that may be allocated to an area), denoted by m .

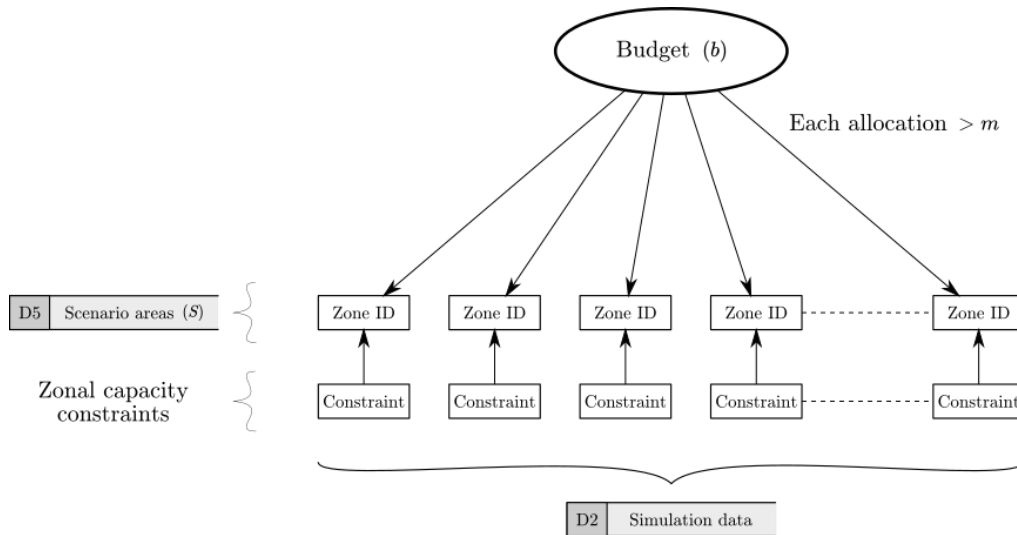


FIGURE 4.5: A visual representation of a policy scenario plan.

Following the construction of a policy scenario plan, implementation of the plan must be considered. During the creation of each policy scenario, the entire budget of policy elements are distributed among the scenario areas in \mathcal{S} . This transforms the scenario areas to *assigned scenario areas* during a mapping of the form $\mathcal{S} \mapsto \mathcal{S}_{ps}$. In order to ensure implementability of the policy in this manner, a multinomial distribution PMF is employed, as described in §2.1.3. The multinomial distribution PMF is employed because it may be altered in a manner ensuring that the entire budget of policy elements are distributed amongst the scenario areas subset during every policy scenario generation. When employing the multinomial distribution PMF, the number of trials, denoted by n , is set equal to the budget constraint of policy elements (that is,

$n = b$). Furthermore, each policy element in the budget is considered the outcome of a trial and has to be distributed among the scenario areas. Therefore, the number of occurrences k is taken as the cardinality of the scenario areas set \mathcal{S} , denoted by η . The probability of scenario area j being selected is denoted by p_j . For the purposes of the UPSOM framework, the probability of each scenario area being selected is assumed to be equal and may therefore be expressed as $p_j = \frac{1}{k}$, thereby allowing for variety in the policy scenarios.

Due to the large number of combinations of policy element allocations, an implementation strategy is required to ensure further variety in the creation of policy scenarios in the sense that policy elements should be widely spread across the scenario areas in some scenarios whilst other scenarios should result in policy elements being concentrated in a handful of scenario areas. These allocations are also required to respect the capacity constraints associated with each zonal area in a bid to generate only feasible policy scenarios.

In order to incorporate variety into allocation concentration, the budget amount and the scenario areas are stochastically compartmentalised. Both the budget and the scenario areas are partitioned into c compartments. Each compartment is then subjected to its own multinomial distribution assignment, with the value of k in the multinomial distribution assignment of each compartment being taken as the number of the scenario areas in the respective compartment. Similarly, the parameter n for each compartmental distribution is set equal to the budget segment in the respective compartment. The numbers of scenario area compartments and the budget compartments are generated stochastically and independently from one another. This occasionally results in a large segment of the budget being distributed amongst a small number of scenario areas and *vice versa*, allowing for variety in the concentration of allocations. A visual representation of the implementation of the policy scenarios may be found in Figure 4.6, producing the assigned scenario areas set, \mathcal{S}_{ps} .

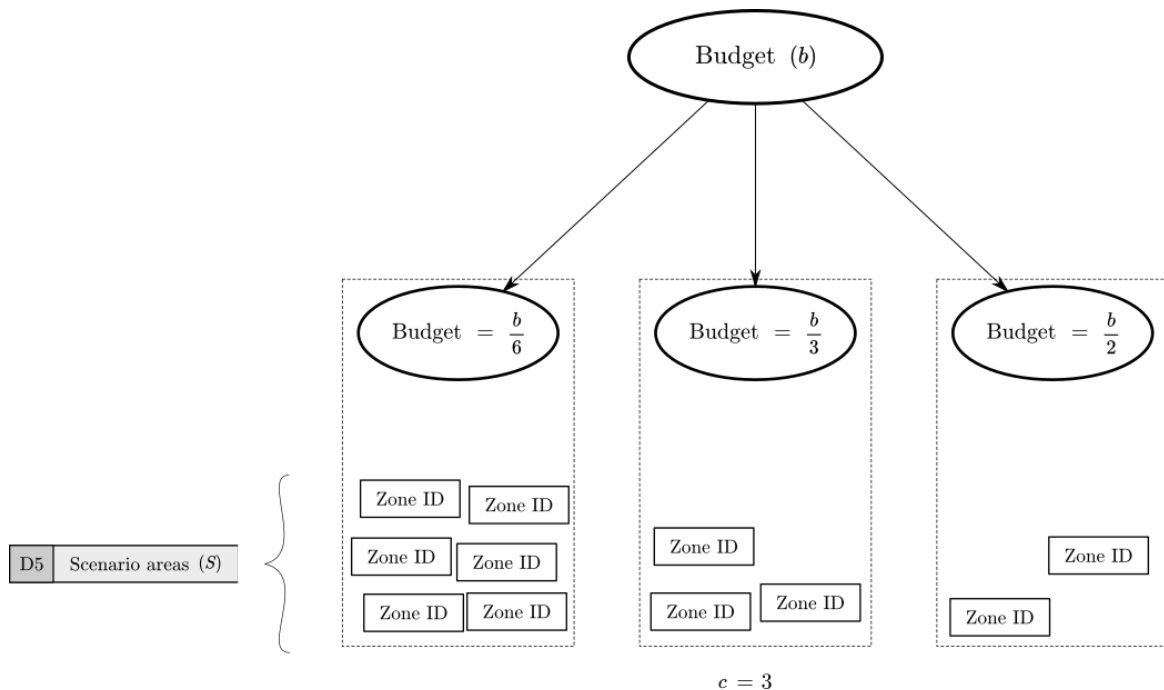


FIGURE 4.6: A visual representation of scenario implementation

4.3.2 Scenario analysis

During implementation of the UPSOM framework, numerous variations in policy scenarios will be created. For the optimisation component associated with the framework to perform properly, each policy scenario variation should be comparable with other variations. In order to facilitate comparisons, the effect of a policy scenario on the variables associated with the features used to assess locational attractiveness should be known. The policy elements allocated would typically alter the values of the features indirectly. This often happens due to the features considered being proximity- or accessibility-based variables instead of the number of policy elements in the relevant zonal area alone. For each policy scenario, feature values must therefore be recalculated before the attractiveness of areas may be assessed. The presence of proximity- or accessibility-based features means that policy scenarios typically result in a trade-off. By assigning policy elements closer to one zonal area it may improve the attractiveness of that particular zonal area more than other areas.

Subsequent to the computation of feature values upon generation of a policy scenario, the fitness of a policy scenario has to be assessed. The UPSOM framework scores the fitness value associated with a policy scenario by first computing the probability of each location alternative being considered. As mentioned above, these probabilities are computed by means of an ITLUM. When UrbanSim is used, a variation of the original location choice model, as described in §3.3.4, is implemented. The newly computed features are employed during the calculation of the probability values which are finally used to score the fitness of a policy scenario. An overview illustration of this policy scenario analysis process may be found in Figure 4.7.

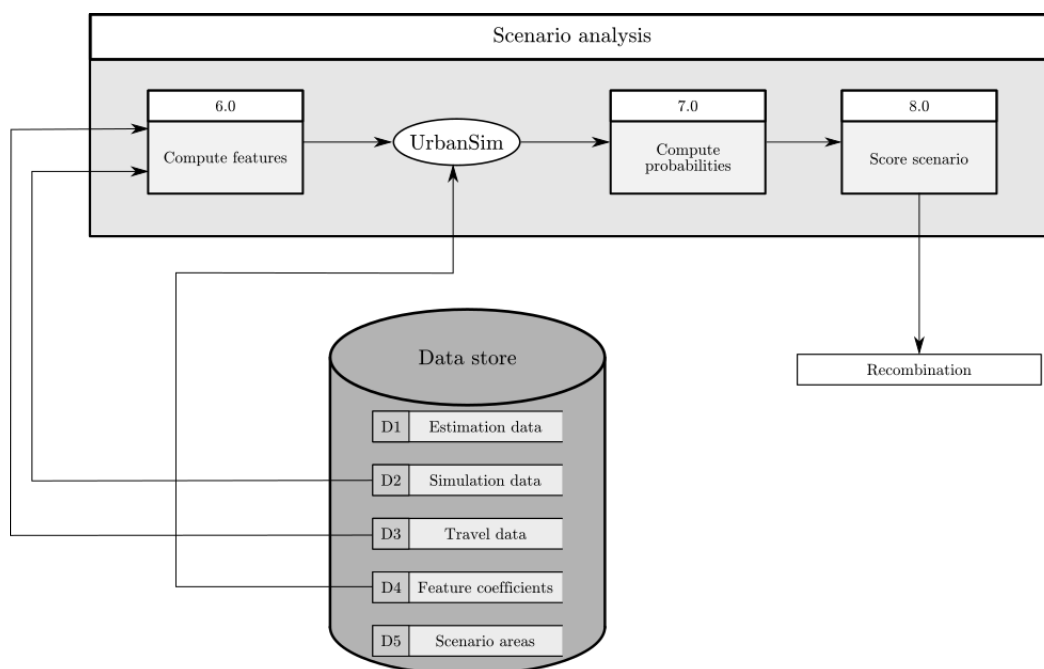


FIGURE 4.7: A visual presentation of how each policy scenario is analysed.

Computation of probabilities

Following the computation of the feature values for a policy scenario, these values replace the feature values for the current base year. The database thus updated contains information pertaining to a potential policy scenario which may transpire if the policy were to be implemented.

The updated database for the policy scenario may therefore be used to assess how well the policy scenario performs in respect of the objective of the UPSOM framework for that particular iteration of policy scenario implementation. In order to calculate the objective function, the probability value of each zonal area being selected by a developer agent for real-estate development has to be computed. The following conditional probability value is computed: Given that an agent develops a certain type of real-estate within a predetermined area, what is the probability of each respective zonal area within that predetermined area being selected for the development. The computation of these conditional probabilities is performed by an ITLUM. In the case of UrbanSim, the location choice model, described in §3.3.4, is applied for this purpose.

While the original version of UrbanSim applied the real-estate development model, as described in §3.3.4, this model has since been changed somewhat. The updated version of UrbanSim employed by the *Council for Scientific and Industrial Research* (CSIR) as of November 2020 and available to the author, employs a real-estate development model which functions in a similar manner to the household location model, described in §3.3.4. The only difference is that the mover agents tasked with selecting the locational alternatives are real-estate developers instead of households. Furthermore, the estimation of the regression coefficients in the vector β is performed in respect of real-estate development data (the number of real-estate units built) instead of household relocation data. Control totals are employed to determine the number of real-estate developments which are to occur within a certain area within a given simulation year.

Given that the coefficients in β and the newly computed scenario feature values X_{ij} are known, the utility of each location alternative, denoted by $U(j)$, may be calculated by employing the aforementioned updated version of the location choice model. In this way, an attractiveness value is associated with each zonal area pertaining to the mover agent. The utility of each zonal area for this specific real-estate development is therefore computed by the multinomial logit utility expression described in §3.3.4. Ideally, the utility values of the priority areas should have increased had good policy scenarios been implemented. The utilities of the zonal area alternatives are then normalised in the multinomial logit probability expression. The resulting output data form a collection of probability estimates captured in a vector \mathbf{P} , representing the probability of each zonal area alternative being selected for real-estate development.

Each individual zonal area, j , has its own probability of being selected within the probability set produced as output, denoted by $P(j) \in \mathbf{P}$. Due to the aforementioned normalisation step, simply maximising the number of policy elements allocated according to a policy would not increase the probability of the priority area being selected, as the policy elements typically increase the utility values of all the alternatives. Therefore, a trade-off is pursued in the UPSOM framework in which an attempt is made to maximise the utility of the priority areas being selected whilst not attempting to increase the utility of the other alternatives too much. The conditional probability is therefore used to maximise priority area attractiveness alone. It should, however, be kept in mind that due to the policy elements having a positive impact on all candidate location zonal areas, the UPSOM framework does not increase the attractiveness of priority areas to the detriment of those of alternative areas, but rather attempts to focus the policy implementation on the priority areas. In this way, the ultimate goal of the UPSOM framework is to assign policy elements to strategic zonal areas.

The coefficients used during each policy scenario probability calculation remains constant. This is because what the mover agent considers as attractive during the base year of the simulation does not change over time, although each policy scenario exhibits different feature values X_{ij} for different numbers of policy elements assigned. For this reason, the UPSOM framework attempts to create the best future policy scenario, given the current feature preferences. Following the computation of the probability vector \mathbf{P} , the fitness function has to be computed.

4.3.3 Fitness scoring

As mentioned above, the execution of ITLUMs and metaheuristics such as the GA, are computationally expensive. In addition, stochastic models typically require numerous solution iterations so as to obtain a good indication of what might be high-quality results. Therefore, the UPSOM framework rather optimises the probability of selection as opposed to the actual selection being made in the ITLUM. This bypasses the stochastic element of having to apply Monte Carlo simulation throughout the computationally expensive optimisation component of the framework, thereby optimising a stochastic model deterministically instead. This means that the same policy scenario would always produce the same result (*i.e.* poor solutions will not sometimes be scored as good solutions due to the stochastic nature of the choice process in the location choice model). The stochastic choice process is only performed subsequent to the probability optimisation process.

The fitness function of the optimisation component is determined by scoring the quality of the probability vector \mathbf{P} . The set of all the zonal areas considered for selection is denoted by J , in which the set of priority areas is denoted by $Z \subset J$. The sum of the probabilities of priority area $z \in Z$ being selected for real-estate development of type $g \in \mathcal{G}$ is maximised and is denoted by $\mathbf{P}(Z_g)$. This aggregation for each real-estate development type $g \in \mathcal{G}$ may be weighted according to the preference of the user, denoted by w_g . If the user of the framework requires more than one real-estate development type to be developed within a priority area, but prefers some types above others, the weighting of the development types may be altered. This creates further trade-offs due to developer of different types of real-estate assessing the importance of features differently, therefore giving rise to different β vectors. The fitness function to be scored is ultimately a weighted sum of the aggregated probabilities of priority areas being selected for different real-estate development types. The objective function for policy scenario ps is expressed mathematically as

$$\max f(ps) = \sum_{g \in \mathcal{G}} w_g \mathbf{P}(Z_g) \quad \text{subject to the constraint} \quad \sum_{g \in \mathcal{G}} w_g = 1. \quad (4.6)$$

4.4 The optimisation component

The policy scenarios returned by the UPSOM framework are iteratively constructed and analysed in an attempt to produce the best possible version of the policy under consideration. The process of iteratively improving the policy scenarios occurs within the optimisation component of the UPSOM framework. This component employs a GA, as described in §2.4. The GA was decided upon due to the nature of the optimisation problem at hand, its large solution space and the computational expense of considering the numerous trade-offs involved. The remainder of this section is devoted to a discussion on each operator employed within the GA embedded in the UPSOM framework.

4.4.1 Population initialisation

The initial population of the GA consists of a set of policy scenarios created by following a policy scenario plan. Each policy scenario is a candidate solution to the combinatorial optimisation problem solved by the UPSOM framework, presented as an array consisting of the assigned scenario areas S_{ps} among the set of all zonal areas. Each entry in the array represents the number of policy elements allocated to the zone in question, reflecting the composition of that

particular policy scenario. The population initialisation operator determines which of the policy scenarios created will form part of the initial population. The size of the initial population is user-specified. This population size is, however, an important determinant of the computational expense of generating a policy scenario.

A large solution space typically means that it may be ineffective to generate the initial population randomly. Such an initial population procedure would typically result in large portions of the solution space remaining unexplored which may, in turn, result in premature algorithmic convergence. The nature of the problem to be solved by the UPSOM framework typically involves a large solution space and so the population initialisation operator is particularly important for promoting diversity in the initial population, as stated in §2.4.1. For this reason, the UPSOM framework employs an SSI population initialisation process, as described in §2.4.1, which allows the initial population to avoid high concentrations of initial solutions in particular areas of the solution space. The number of dimensions in a policy scenario solution is the cardinality of the scenario zone set η . In order to calculate the distance between each pair of solutions, a high-dimensional Euclidean distance operator is employed [31]. This multi-dimensional Euclidean distance operator d , applied to two solutions a and u , is

$$d(a, u) = \sqrt{\sum_{v=1}^{\eta} (u_v - a_v)^2}.$$

As per the SSI population initialisation process, the population is initialised with a single solution, followed by $\eta - 1$ solutions in succession. The distances between every newly generated solution and each of the already existing solutions are computed. If the distance between the newly generated solution and any existing solution is less than a threshold value Δ , the newly generated solution is rejected and another solution is generated.

Before the first solution is generated, a neighbourhood threshold radius Δ is estimated. During this step, q solutions are stochastically generated beforehand. The distances between a randomly selected solution a and all the other solutions u_1, \dots, u_q are computed. A predetermined fraction, denoted by r , of the average of the distances computed is then taken as the neighbourhood threshold radius, expressed as

$$\Delta = \frac{\sum_{t=1}^q d(a, u_t)}{q} \times r.$$

Therefore, the condition for accepting a newly generated solution s_i , considering all the existing solutions s_1, \dots, s_{i-1} in the search space, is $d(s, s_j) > \Delta$ for all $j \in \{1, \dots, i - 1\}$ values. This process is repeated until the entire initial population, denoted by \mathcal{H} , has been generated.

4.4.2 Selection

Following the initial population generation, the iterative operators of the GA may be employed. The first of these operators is responsible for the selection of parent solutions. The selection operator in the UPSOM framework employs roulette wheel selection, as described in §2.4.2. Application of this operator requires the fitness value associated with every solution in the population, as expressed in (4.6). Thereafter, a typical roulette wheel selection procedure commences. Two parent solutions are selected for mating purposes from the population. These parents are then subjected to mutation and replacement operators after which they are removed from the parent population. The process is repeated until only two parent solutions remain. When this final pair of parent solutions mate, a generational iteration has been completed.

Similarly to (2.10), the probability of a parent solution $h \in \mathcal{H}$ being selected as a parent is

$$p(h) = \frac{f(h)}{\sum_{g \in \mathcal{H}} f(g)}, \quad (4.7)$$

where $f(h)$ denotes the fitness value of h . For every selection operation, the probability of each parent solution being selected is recalculated, thereby producing a new discrete PMF for every selection operation. The parent solutions which will eventually mate are selected by employing a Monte Carlo simulation sampling technique in conjunction with the updated PFM, as described in §2.2.

4.4.3 Crossover

Subsequent to the selection of two parent solutions, the crossover operator is employed within the UPSOM framework. As stated above, each population consists of a variety of heterogeneous solutions in which the concentrations of assignments are significantly different. As a result of this concentration discrepancy, it is very likely that an offspring solution will be infeasible due to the entire budget b not being assigned or more policy elements than the allowed number b being assigned. A basic crossover operation, such as single-point or two-point crossover, would create too many infeasible offspring. Therefore, the UPSOM framework employs uniform crossover. The numerous cuts applied in uniform crossover allows for a higher probability of a highly concentrated solution and a fairly spread-out solution, more often producing feasible offspring solutions.

Uniform crossover is performed on the parent solutions with a probability of p_c . In addition, if the crossover is not performed, the parent solutions are simply added to the offspring population. Crossover is performed, as described in §2.4.3, with α cuts being applied within the solution array. The value of α depends on the size of the solution array (the number of scenario areas), which is η , where a larger value of η requires more cuts and *vice versa*. In cases where parent solutions produce infeasible offspring solutions, the positions of the crossover cuts are randomly re-selected. This process is iterated until feasible solutions are produced or a maximum number of cut re-selection iterations have been performed, in which case the parent solutions are simply added to the offspring population.

4.4.4 Mutation

After the creation of offspring solutions by means of the crossover operator, these offspring solutions may be subjected to mutation, which is performed on a single offspring solution with a probability p_m . The type of mutation performed within the UPSOM framework is integer swap mutation, as described in §2.4.4. The mutation operator is required to promote diversification in offspring solutions and to offset to some extent the elitism produced by the crossover operation. Integer swap mutation was selected due to the budget constraint. This type of mutation only exchanges the values of the policy elements assigned to two different zonal areas within an offspring solution, thereby automatically respecting the total number of policy elements in each solution (and thus satisfying the budget assignment constraint b).

By upholding the budget constraint, however, integer swap mutation might produce infeasible results due to violating individual zonal capacity constraints. Simply re-performing the mutation operation until a feasible solution is generated, as in the case of the crossover operator, is not ideal as zonal areas with small numbers of policy elements assigned to them would incorrectly be favoured for mutation. Zonal areas with large numbers of policy elements assigned to them

would, however, often produce infeasible solutions due to the large number of policy elements often being larger than the capacity constraints of the majority of zonal areas. When these zonal areas are selected for mutation, they are hence more likely to produce infeasible offspring solutions. A re-performance of the mutation operation, therefore, would typically result in only small allocations being mutated. In order to allow both small and large policy element allocations to be accepted during mutation, the integer swap mutation operator is adjusted slightly. First an initial zonal area is selected for mutation. Before the second zonal area is, however, selected to perform the mutation, zonal areas with a capacity more than the number of policy elements allocated to the first zonal area (feasible zonal areas for integer swapping) are sampled from the set of scenario areas \mathcal{S} . Entry swap mutation is then performed in conjunction with a randomly selected zonal area within this feasible subset. Each mutation performed is, therefore, feasible and this approach allows for both small and large policy element number to be included in mutation swaps.

4.4.5 Replacement

The optimisation component of the UPSOM framework first initialises a population of policy scenario solutions. The population then proceeds to produce offspring solutions by means of the selection, crossover and mutation operations. The offspring solutions are next analysed and scored by computing the offspring solutions' fitness values. Thereafter, the fitness of each solution in the parent population and in the offspring population is known and a replacement strategy must be employed to create the population for the next generational iteration. The UPSOM framework employs a generational replacement strategy, as described in §2.4.5. A variation on the elitism generational replacement strategy is employed which is inspired by the strategy implemented by Chelouah and Siarry [11]. This variation allows the entire offspring population to become the parent population during the subsequent generational iteration. If the offspring population does not, however, contain the incumbent solution following a generational iteration, the incumbent solution from the parent population is carried over to the next generation, replacing the offspring solution with the weakest fitness function value, as depicted in Figure 2.14. This replacement strategy allows for a good balance between diversification and elitism in the population of the subsequent generational iteration.

4.4.6 Stopping criteria

The UPSOM framework employs an adaptive stopping criterion, as described in §2.4. The stopping criterion is therefore not known *a priori*. The operations of the optimisation component are repeated until a certain number of generational iterations have been completed without producing a new incumbent. If the incumbent solution is successfully carried over from the offspring solution over a certain number of consecutive iterations, it is assumed that convergence to a local optimum has occurred, upon which the algorithm terminates and the incumbent solution is returned as the approximately optimal final solution.

4.5 Results produced by the framework

When the stopping criterion is satisfied, the optimisation component terminates its operations and produces a policy scenario deemed to be a high-quality version of the policy considered, which is that recommended for implementation. This output takes the form of an array of policy element allocations amongst the η scenario zones, denoted by \mathcal{S}_{ps} , and referred to as *optimal*

zones. These optimal zones are presented to the user in both tabular and visual formats. An example of a visual representation of the output may be found in Figure 4.8, while an example of the same output in tabular format may be found in Figure 4.9. If the user is satisfied with the result, the optimal zones are implemented as a final policy scenario within the ITLUM. A complete simulation is then performed for a single simulation year, given the policy scenario produced. In the case of UrbanSim, the version of the location choice model described in §4.3.2 is implemented, including Monte Carlo simulation for simulating the final choices made by real-estate developers.

\mathcal{S}_{ps}

Zone ID	Assignments
1	0
2	10
3	130
⋮	⋮
η

FIGURE 4.8: A tabular representation of the optimal zones output.



FIGURE 4.9: A visual representation of the optimal zones output.

The nature of the solution produced allows for an improvement which typically seems relatively small when expressed as a percentage of improvement. The reason for this is that the UPSOM framework usually optimises only one or two features which have an impact on the attractiveness of a zonal area location. Consider, for example, the situation where a real-estate developer

considers ten features on which (s)he bases assessments of the relative attractiveness of a number of candidate areas. The framework would often improve only one of those ten feature values. The policies implemented have a positive impact on all the zonal areas and, therefore, a large increase in the utility of selecting one zonal area would typically result in a large increase in the utility of selecting many alternative zones within a certain radius from that zone. By improving the utility of a priority area, alternative zonal areas close to the priority area will therefore also see an improvement in attractiveness. This increase in utility of all the zones, in conjunction with probabilities produced when normalising the utilities of each zonal area, would typically result in a small improvement in probabilities of the priority areas being selected for development. It should, however, be considered that the number of real-estate developments within a metropolitan area is typically large and that a small improvement in the probabilities would result in many developments being influenced. The UPSOM framework, in addition, optimises a policy scenario for only one year, and so if it were to be implemented each year over a significant period of years, it would result in a large number of developments being influenced. A compounding effect on the attractiveness of priority areas would therefore take place if the framework were to be implemented annually with the same priority areas in mind over a period of several years.

4.6 Chapter summary

This chapter was devoted to a presentation of the UPSOM framework proposed in this thesis for facilitating incentivisation policy generation aimed at increasing real-estate development in prioritised urban areas. The chapter opened with a high-level overview discussion the framework architecture in §4.1, in which each the various components of the framework were described briefly. The discussion then turned to an in-depth description of the first component in the framework, namely its preprocessing component (in §4.2), containing various modules. The process applied when policy scenarios are created and analysed was described thereafter in §4.3. The optimisation of these policy scenarios is the objective of the framework. This optimisation process, embedded in the optimisation component, was next described (in §4.4). The GA operators employed within the framework, as well as the reasoning behind every design choice, was discussed. The nature of typical final results produced by the framework was also finally discussed in §4.5.

CHAPTER 5

Case study

Contents

5.1	Background	71
5.1.1	<i>City of Ekurhuleni</i>	72
5.1.2	<i>Case study dataset</i>	73
5.2	Case study preprocessing	74
5.2.1	<i>Case study scenario areas</i>	75
5.2.2	<i>Case study coefficient estimation</i>	75
5.3	Policy scenario construction	79
5.3.1	<i>The policy scenario plan</i>	80
5.3.2	<i>Policy scenario implementation</i>	80
5.3.3	<i>Policy scenario analysis</i>	81
5.4	Results and discussion	83
5.5	Chapter summary	87

This chapter is devoted to an in-depth description of a real-life case study to which the UPSOM framework proposed in the previous chapter was applied as a proof of concept. The results produced according to the framework application are presented and discussed. The chapter opens with a background discussion on the South African province of Gauteng and the City of Ekurhuleni within that province, which is the geographic area of the case study. The case study data are described after which the working of the various components of the UPSOM framework are discussed in the context of the case study data. Initially, the data preprocessing operations are described, and this is followed by a discussion on the process of policy scenario construction and analysis. The focus of discourse then turns to a brief discussion on how the optimisation component of the framework was applied to the case study data. The final results returned by the UPSOM framework are finally presented, illustrated graphically and discussed.

5.1 Background

Gauteng province is located in the north-eastern region of South Africa. It is the smallest province in South Africa, but is the most populated province in the country [39]. In 2018, the province was home to approximately 14.7 million South Africans [78], of which the vast majority resided in urban or suburban areas. Some of the largest metropolitan areas in South Africa are located in the province, such as the countrys financial, industrial and commercial

centre, Johannesburg, as well as some other notable urban areas such as Pretoria, Germiston and Vereeniging, to name but a few [23]. Due to Gauteng being considered the economic hub of the country, it typically experiences large levels of migration. It is the province which receives the majority of migrants from other areas in South Africa and from abroad [78]. The profile of migrants moving to Gauteng is relatively heterogeneous in terms of background and economic status. As of 2020, Gauteng is home to an estimated 15.5 million residents [77]. The province is, furthermore, experiencing rapid growth in the region of approximately 400 000 people per annum. With such a growth rate, a major housing demand is experienced in the province. In the building statistics report compiled by Statistics South Africa [75], it is stated that Gauteng saw the completion of 8 975 dwelling houses and the completion of 9 231 flats and town houses during 2018, built in an attempt to stratify the increase in housing demand.

This rapid growth is further contributing to an already segregated society in the nation's most populous province. Municipalities in Gauteng are, therefore, actively engaged to encourage spatial integration of the growing population in a bid to ensure a better quality of life for their residents.

5.1.1 City of Ekurhuleni

One of the major cities located in Gauteng is the City of Ekurhuleni. The city is home to approximately 3.77 million people, comprising approximately 1.3 million households. It has a significant poverty-ridden population, with 31% of its inhabitants living in poverty when using the upper bound poverty line as an indicator [13]. With a Gini coefficient of 0.633, inequality is a major concern in a city where the marginalised of society are experiencing a significantly worse quality of life than the wealthy due to the former being marginalised and not sufficiently integrated into society. The Ekurhuleni municipality has drawn up a plan for the future of the city's spatial development. This is known as the Ekurhuleni *Metropolitan Spatial Development Framework* (MSDF) [12]. The MSDF is aimed at developing an equitable spatial development model for assisting in rectifying the imbalances of the city. The framework has the objective of transforming the municipality from a low-density, private transport-dominated, dispersed urban structure to a highly dense, public transport-orientated, compact urban structure. Strategic areas within the city are earmarked in the framework for receiving spatial prioritisation with respect to capital expenditure programs. These areas are known as *geographic priority areas*. Among the geographic priority areas, there are *densification areas* and *expansion areas*. The densification of these areas is required to achieve spatial co-ordination and strive toward inclusive spatial development. These densification areas are therefore considered as priority areas within the UPSOM framework case study application of this chapter.

The case study to which the UPSOM framework is applied is aimed at providing decision support in respect of promoting the densification of the geographic priority areas identified within the MSDF. The aim of the case study is to promote residential real-estate development within these areas in order to supply housing in strategic areas for the rapidly growing population of Ekurhuleni. The user of the UPSOM framework within this case study is, therefore, considered to be the municipality of Ekurhuleni which is assigning a budget of policy elements to promote the attractiveness of the geographic priority areas identified in the MSDF as priority areas within the UPSOM framework. The preprocessing component of the framework may take the aforementioned densification areas as priority area input data, as described in §4.2. In total, 1 123 densification areas within the geographic priority zones have been identified in an attempt to support public transport and urban stability. These areas are the Albertina Sisulu Corridor, the ORTIA–Daveyton Link area and the Leeuwpoort area [12], illustrated graphically in Figure 5.1.

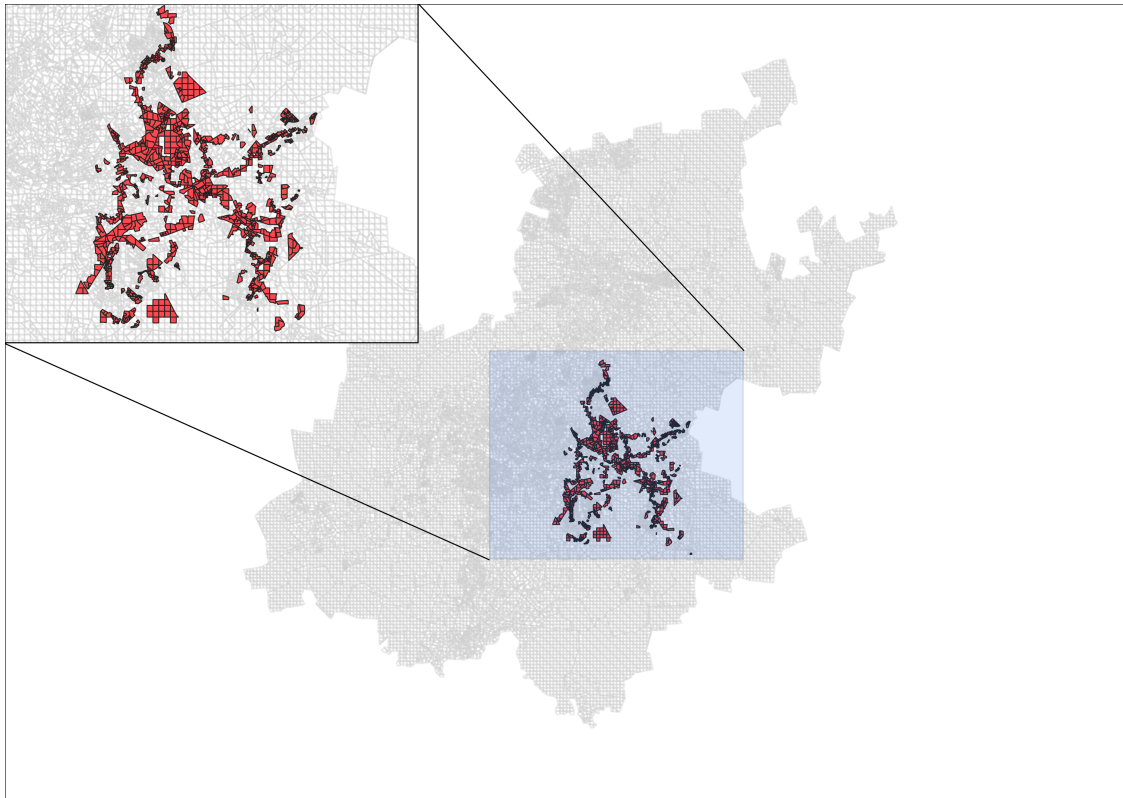


FIGURE 5.1: *Geographic priority areas in the City of Ekurhuleni, as identified in the MSDF [12].*

5.1.2 Case study dataset

As a developing country, South Africa typically does not have the capacity to collect comprehensive census-type data on an annual basis, as is required by ITLUMs. The majority of these data is acquired during the decennial census performed by Statistics South Africa [76]. The most recent census at the time of writing this thesis occurred in 2011, and so a comprehensive database on Gauteng province was available to the author, provided by the CSIR, for the year 2011. The province is partitioned into zones in this data set, with each zone being assigned a unique zone identification number. The data points for each zone describe the spatial attributes of the zone and includes features such as mean household income, number of upper class real-estate units and number of employment opportunities, to name but a few. An additional data set provided to the author by the CSIR contains travel data. This data set indicates the travel distance and travel time between each pair of zones in Gauteng province for the year 2011. All of the aforementioned data are taken as the simulation data for the case study application of the UPSOM framework, making 2011 the base year of the simulation.

No such comprehensive data sets were, however, available to the author for any subsequent years. A smaller database for the year 2018 was nevertheless available to the author, also provided by the CSIR. This database contains the number of residential real-estate units within each zone in the Gauteng area. Unfortunately, this database does not contain any other spatial features for the year 2018, other than zone identification numbers and the number of residential real-estate units in these zones.

Prior to the execution of the UPSOM framework, the simulation data set was explored by the author. The purpose of this exploration was to identify the spatial features available in the data set. Understanding the nature of the spatial features in the simulation data set beforehand is

essential, because features are selected to assess the attractiveness of a zonal area for residential real-estate development within the preprocessing component of the UPSOM framework. A solid understanding the spatial features also facilitates the identification of potential changeable features early on which may form part of urban policies within the framework.

During the simulation data set exploration, it was observed that there are three different types of spatial features within the data set. These features are:

Internal features. These spatial features describe the internal attributes of the zones. These features are located wholly within the boundaries of zones and include developable land area in the zone, number of real-estate units in the zone and mean year built for real-estate in zone, to name but a few.

Accessibility features. These spatial features describe the travel accessibility of the zone with respect to features outside the boundary of the zone, and include accessibility to employment and accessibility to a highway road, to name but two examples.

Proximity features. These spatial features describe the proximity of a zone with respect to features outside the boundary of the zone. As opposed to accessibility features, which pertain to travel distance, these features are simply based on the Euclidean distance between the zone and the feature in question. These features include proximity to informal settlements, proximity to residential real-estate units and proximity to industrial areas.

5.2 Case study preprocessing

Subsequent to the exploration of the 2011 data set and considering the practical capabilities of the Ekurhuleni municipality (the user of the UPSOM framework), potential policies were identified. For these potential policies to be identified, the limits of possible scenario areas had to be known and it had to be kept in mind that the potential policies have to be realistically implementable by the user. The policies identified for the case study of this chapter are:

- *Real-estate price.* Subsidising the cost of housing in strategic areas may be a potential policy due to the data set containing the mean cost of residential real-estate for different real-estate types. Therefore, this is a feature which may possibly be changeable. It is supposed that subsidising real-estate prices is an intuitive policy that the user may realistically implement.
- *Job accessibility.* Creation of jobs in strategic areas may be another potential policy due to the capacity of vacant job spaces in each zonal area being available in the data set. The South African government is also known to be the largest creator of employment opportunities in the country. Therefore, strategically selecting the locations of these newly created governmental jobs may be an intuitive policy that is changeable. From the middle of 2010 until the middle of 2011, approximately half of all jobs created in South Africa were in the public sector [68].
- *Grant provision.* Providing grants to households when they move to strategic areas may be a potential policy due to the mean household income per zonal area being available in the case study data set.

5.2.1 Case study scenario areas

Considering the fact that the priority areas are known, the *select priority areas* module of the preprocessing component in the UPSOM framework may be considered to have been invoked successfully. Moreover, scenario areas may be identified to scope the sample space of possible solutions. The province of Gauteng is partitioned into 28 460 zonal areas, making for a rather large sample space of potential policy scenarios. By applying the *compute scenario areas* module, described in §4.2.1, a scenario areas set may be identified. For the purpose of this case study, a radius of 1 km was selected (*i.e.* $\theta = 1$). The scenario set \mathcal{S} was identified thereafter for each of the aforementioned policies. The real-estate price and grant provision policies exhibit scenario areas which surround all the priority areas as per the usual implementation of the *scenario area identification* module. The job accessibility policy is, however, subject to certain individual capacity constraints. Each zonal area is equipped with a vacant job spaces feature. More specifically, if the number of vacant job spaces is less than a certain threshold value, then jobs may not be created without new commercial real-estate also being constructed. For the purposes of this case study, only zonal areas exhibiting more than five vacant job spaces were considered. Potential scenario area sets for the real-estate price or grant provision policies, and for the job accessibility policy, are illustrated in Figures 5.2 and 5.3 respectively.

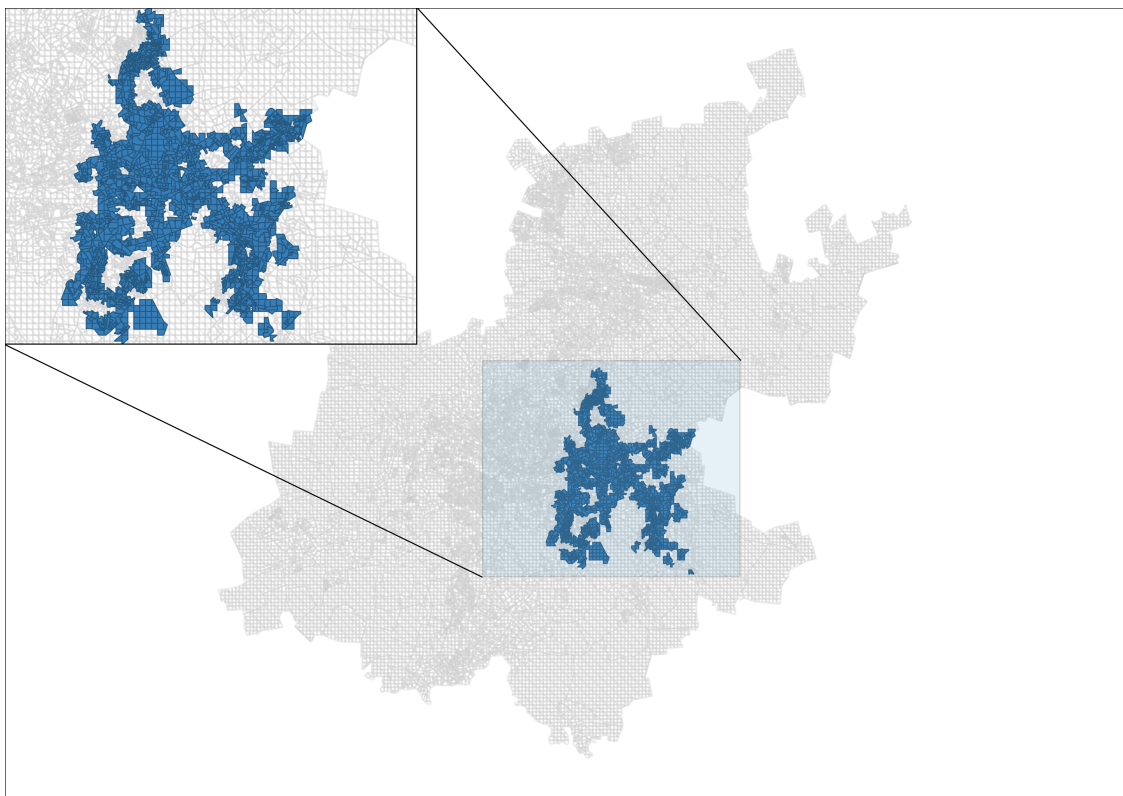


FIGURE 5.2: *Potential scenario areas for real-estate price or grant provision policies.*

5.2.2 Case study coefficient estimation

Given the limited data sets available, certain steps of the UPSOM framework required priority above others. The estimation of regression coefficients is an absolutely critical step when applying the framework, considering that these coefficients are employed in the objective function calculation. If the coefficients are not estimated carefully, the entire framework application may

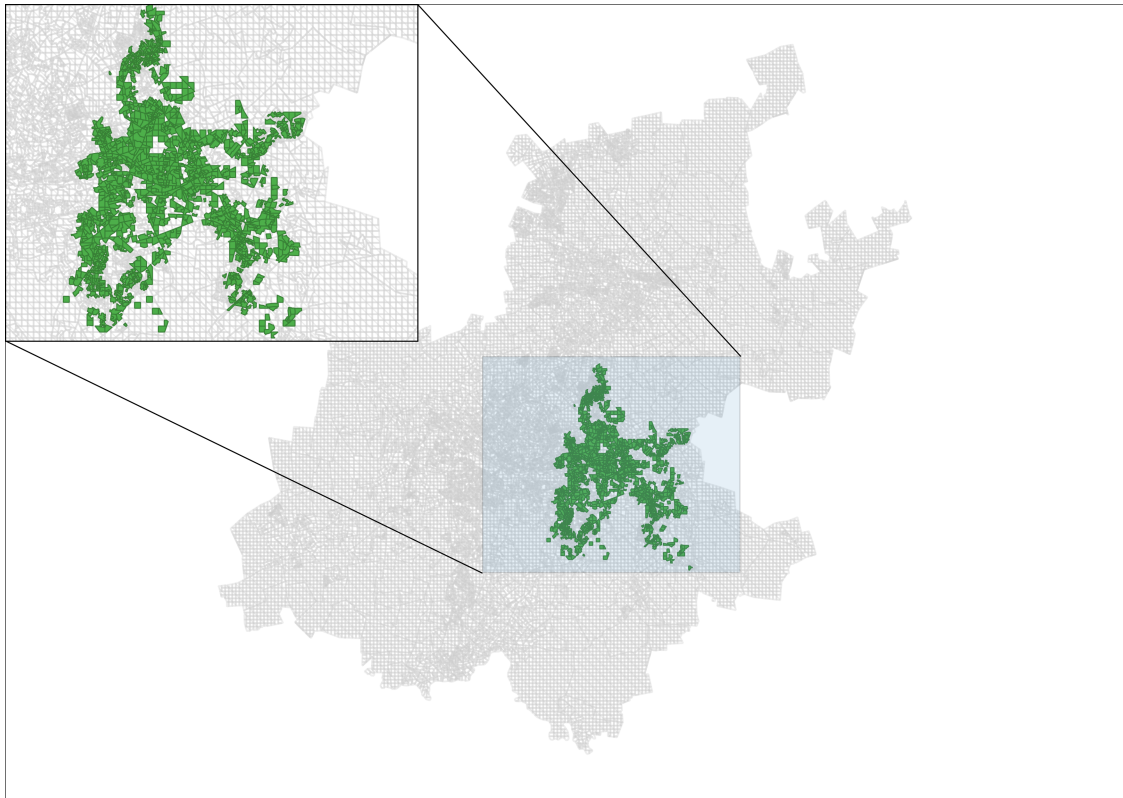


FIGURE 5.3: *Potential scenario areas for job accessibility policies.*

be rendered unrealistic as a result of scoring poor solutions well and *vice versa*. In this case study, all the data available were used to estimate the regression coefficients in a bid to ensure intuitive coefficient values which coincide with estimations from similar problems documented in the literature.

Data preparation and transformation

For the *data preparation* module of the preprocessing component, described in §4.2, the 2011 simulation data set and the 2018 real-estate number data set were combined in order to facilitate the construction of the growth data required during coefficient estimation. The newly combined data set could then be employed to construct the dependent variable required for coefficient estimation during execution of the *data transformation* module, described in §4.2.2. The difference in the numbers of real-estate units of each real-estate type in 2018 and in 2011 was computed and added to the combined data set. These differences represent the number of residential real-estate units created or destroyed during the period 2011–2018. This final estimation data set was further scoped down by only selecting those zones within the data set which exhibit a positive difference in the newly generated difference variable feature. Since the goal of the policies to be implemented is to encourage new residential real-estate developments, the construction of new residential units over a certain period of time is indicative of the fact that the circumstances during the case study base year rendered certain areas more attractive to real-estate developers than others. In other words, a positive difference in the number of real-estate units indicates that certain areas were considered more attractive for development during the base year than others due to the construction of residential real-estate being a time-consuming activity, whilst the initial decision to develop considers the environment and spatial features when the decision

was made, before construction. The computation of *growth* data ensures that the coefficients estimated for the features used to assess the attractiveness of a zonal area represents causation and not simply correlation.

The ITLUM employed during the case study was UrbanSim, which was thoroughly described in §3.3. Previous users of UrbanSim have argued that simply using the number of residential real-estate units in the base year as an indication of attractiveness is sufficient (due to the lack of comprehensive data sets spanning multiple years). The author disagrees with this sentiment for two reasons. First, the construction of real-estate is time-consuming, as mentioned above, and therefore initially attractive spatial features would result in construction during subsequent years. When simply taking the number of real-estate units during the base year as the dependent variable, the correlation between the spatial features and number of real-estate units could therefore distort the regression coefficients. This is because the temporal element of construction is not considered during the coefficient estimation process as it is the initial spatial composition which results in construction during subsequent years.

The second reason is more case study-specific. South Africa has a complicated recent past as a result of the Apartheid regime, discussed in Chapter 1. One of the goals of the Apartheid regime was to forcefully relocate vast numbers of households to specific areas. This resulted in the construction of residential real-estate in these areas simply to supply housing to the relocated people. This resulted in certain areas being densely populated whilst others were sparsely populated through social engineering by a malicious government. The author, therefore, considers this growth to be forced and inorganic. It consequentially cannot be used to predict future growth patterns due to the environment in the current South Africa, taking a significantly different long-term developmental view. The growth data constructed over the period 2011–2018 is expected to better indicate organic growth in a democratic South Africa, where freedom of movement and free choice is more accessible to the South African public.

Coefficient estimation

Subsequent to the preparation and transformation of the data, the *estimation* module, described in §4.2.2, was invoked. From the transformed data, it was observed that only two residential real-estate unit types exhibited a sufficient number of positive zonal growth data points. The growth data for these two real-estate types were taken as values the dependent variable during coefficient estimation. The two residential real-estate types in question are *real-estate type 201*, which represents high-end houses with a value of more than R 1 million, and *real-estate type 202*, which represents entry-to-middle class houses with values between R 400 000 and R 1 million. These real-estate definitions and classification of units were provided by the CSIR together with the various data sets and are based on the land-use type as well as the valuation of the building in question.

The estimation of coefficients requires that spatial features have to be identified which contribute to the attractiveness of a zonal area from the perspective of a residential real-estate developer. For the purposes of this case study, a set of features which could possibly influence the attractiveness of a zonal area were identified by a combination of intuition, consulting the literature on similar problems and consulting industry subject matter experts. The final list of possible features considered was:

- Proximity to informal settlements,
- Proximity to commercial real-estate,
- Proximity to retail real-estate,

- Proximity to light industry real-estate,
- Proximity to heavy industry real-estate,
- Mean income of households within a certain proximity,
- Accessibility to employment,
- Accessibility to highway roads,
- Mean price of similar residential real-estate units in zone,
- Mean price of all residential real-estate units in zone,
- Mean year built of similar residential real-estate units in zone,
- Household density in zone,
- Number of public schools in zone,
- Capacity for construction of similar residential real-estate units in zone, and
- Developable land area in square meters in zone.

The regression coefficients for both real-estate type 201 and real-estate type 202 were estimated as part of an application of the *estimation* module in the preprocessing component of the UP-SOM framework, as described in §4.2.2. During this estimation, the growth of real-estate units of types 201 and 202 was considered separately as the dependent variable (an indication of attractiveness of a zone), whilst a subset from the aforementioned set of possible features was employed as the set independent features (causation of attractiveness). A typical implementation of the *estimation* module proceeded during which the MLE method was applied and a Poisson distribution was assumed due to the nature of the data. The estimation process involved considering numerous subset combinations of the aforementioned feature data set. The assessment of the quality of an estimation was based on pseudo R-squared¹ values. The best fitting subset of coefficients thus estimated, together with their respective coefficients, may be found in Table 5.1. A pseudo R-squared score of 0.116 was achieved during this estimation process.

TABLE 5.1: *The coefficients estimated for assessing the attractiveness of a location for real-estate development of a type 201 real-estate unit.*

Feature	Feature type	Coefficient
Developable land area in square metres	Internal	1.89×10^{-6}
Number of informal settlements within 3km	Proximity	-5.9×10^{-3}
Number of jobs within 20 minutes commuting	Accessibility	1.396×10^{-4}
Mean type 201 real-estate price	Internal	8.839×10^{-8}
Mean year built for type 201	Internal	1.5×10^{-3}
Mean residential real-estate price	Internal	5.907×10^{-7}
Mean household income within 3km	Internal	-1.403×10^{-6}
Mean road meters to highway	Accessibility	-1×10^{-4}
Number of commercial buildings within 3km	Proximity	-5.43×10^{-5}
Number of retail buildings within 3km	Proximity	-9.884×10^{-5}

For real-estate of type 202, the pseudo R-squared score of the best fitting subset of coefficients was 0.112, and these coefficients are provided in Table 5.2.

¹The pseudo R-squared values employed, alternatively known as the likelihood-ratio index suggested by McFadden [55], compares a model without predictors with a model including all the predictors. The value is defined as the ratio of the log likelihood containing only the intercepts and the log likelihood containing all predictors subtracted from one.

TABLE 5.2: *The coefficients estimated for assessing the attractiveness of a location for real-estate development of a type 202 real-estate unit.*

Feature	Feature type	Coefficient
Capacity for number of type 202 units	Internal	1.5×10^{-3}
Developable land area in square metres	Internal	2.998×10^{-7}
Number of informal settlements within 3km	Proximity	-2.011×10^{-5}
Number of jobs within 20 minutes commuting	Accessibility	2.562×10^{-5}
Mean type 202 real-estate price	Internal	-3.638×10^{-7}
Mean year built for type 202	Internal	9×10^{-4}
Mean residential real-estate price	Internal	1.549×10^{-6}
Mean household income within 3km	Internal	4.852×10^{-7}
Mean road meters to highway	Accessibility	-2.937×10^{-5}
Number of commercial buildings within 3km	Proximity	-5.944×10^{-5}
Number of retail buildings within 3km	Proximity	1×10^{-4}

When considering the literature, the results in Tables 5.1 and 5.2 seem plausible as the magnitudes of the regression coefficients coincide with the scale of their expected feature values. Consider, for example, the very small coefficient estimated for developable land area in square metres. These feature values are typically very large and so a very small coefficient is expected. Moreover, the signs of the coefficients seem intuitive and coincide with similar coefficient estimation problems documented in the literature. Consider, for example, the number of informal settlements within 3km. It is intuitive that a zonal area situated within close proximity of informal settlements would negatively impact its attractiveness in terms of developing the residential real-estate types considered in this case study. In addition, the signs of coefficients for the number of jobs within 20 minutes commuting (accessibility to employment) and mean road metres to highway (accessibility to highway roads) coincide with estimations in similar problems documented in the literature [30, 83].

5.3 Policy scenario construction

Following a comprehensive exploration of the data sets and the execution of the preprocessing component of the UPSOM framework, the construction of policy scenarios could be initiated. Based on the features selected and the coefficients estimated in §5.2.2, it is clear that some of these features are changeable by the Ekurhuleni municipality (the user of the UPSOM framework), whilst some features are not. As mentioned in §5.2, potential policies had already been identified prior to the coefficient estimation procedure. Given the features identified for accessing locational attractiveness, the three potential policies initially considered were confirmed to alter certain features which contribute to the assessment of the attractiveness of zonal areas. The features altered by these three policies are considered changeable, which is a requirement for policy selection, as stated in §4.3.1. The potential policies are therefore considered to be pragmatically implementable. For these policies, the policy elements and the features which are changeable for each policy are identified in Table 5.3.

For the grant provision and real-estate price policies, the policy elements are the amount of capital allocated for grants or subsidies per zonal area, measured in Rands. In the case of the job accessibility policy, the number of governmental jobs created per zonal area is the policy element which is to be assigned.

TABLE 5.3: *Summary of potential policies that may be implemented in the case study.*

Policy ID	Policy name	Policy elements	Features altered
Policy 1	Real-estate price	Capital (Rands)	Mean type 201 real-estate price Mean type 202 real-estate price Mean residential real-estate price
Policy 2	Job accessibility	Employment (jobs)	Number of jobs within 20 minutes commuting
Policy 3	Grant provision	Capital (Rands)	Mean household income within 3km

For the purposes of this case study, Policy 2 in Table 5.3 was selected. As mentioned above, the South African government is the largest employment provider in the country. When the municipality of Ekurhuleni is creating employment opportunities or moving the locations of already existing employment, it would seem intuitive to select the employment locations strategically within the municipality boundaries so as to assist with spatial integration as the employment will be created in any case. Given that the policy elements (jobs to be allocated) and zonal capacity (vacant employment locations) are known, the subsequent step during policy scenario creation would be to construct a policy scenario plan.

5.3.1 The policy scenario plan

The creation of a policy scenario plan is required to take into account a high-level overview of all the variables involved during policy scenario construction, as stated in §4.3.1. The scenario area set for Policy 2 has been identified in §5.2.1 and consists of 1 494 possible zonal areas. These zonal areas are therefore possible locations for the assignment of policy elements in the form of governmental jobs.

In order to remain pragmatic during the assignment of jobs to zonal areas, jobs were assigned in multiples of 5 and zonal areas with vacant job capacities smaller than 5 were not considered in the scenario areas set. Moreover, the total number of jobs b to be allocated during every policy scenario is altered during different implementations of the optimisation component of the UPSOM framework for illustrative purposes. Realistic budget capacities for multiple implementations of the optimisation component in this case study were considered to be 2 000 and 3 000. The policy scenario plan constructed for this case study is illustrated in Figure 5.4.

5.3.2 Policy scenario implementation

Following the construction of the policy scenario plan, the implementation of the plan could commence. The scenarios were created, as described in §4.3.1, to allow for concentration variety in different policy scenarios. For the purposes of this case study, the budget and scenario zones were compartmentalised into five compartments (*i.e.* $c = 5$). This resulted in a variety of job assignment solutions being created for which variety in assignment concentration was prevalent. Two policy scenarios created for the case study exhibiting assignment concentration variety are illustrated in Figure 5.5, with both policy scenarios assigning a total of 3 000 jobs.

The number of jobs in a zone was not a feature considered when the attractiveness of zonal areas were assessed from the perspective of a real-estate developer. The number of jobs within a 20 minute commute was, however, a feature considered when this attractiveness was assessed. The latter is, therefore, an accessibility feature. This feature was calculated by employing the simulation data set for 2011, which includes the numbers and locations of employment

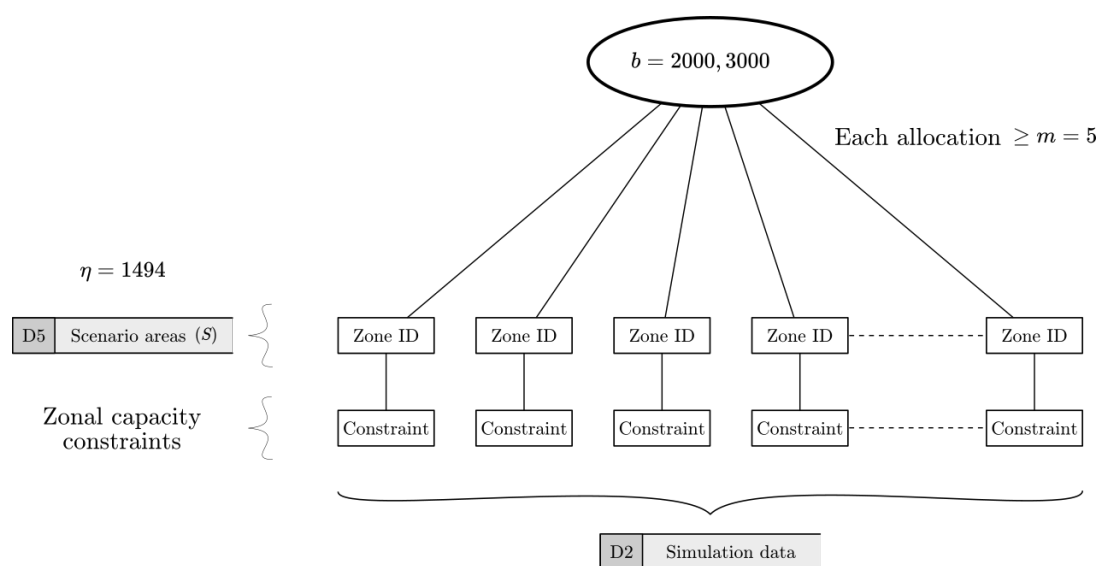


FIGURE 5.4: Policy scenario plan for the case study.

opportunities on a zonal level. Moreover, the travel data set was employed, which contains the average commute time between every pair of zones within the simulation data set. Therefore, the number of employment opportunities in all zones within a 20 minutes travel time was aggregated to compute this feature value for every zone in the data set.

The aforementioned feature had to be recalculated for every policy scenario created. This was achieved within the *compute features* module in the scenario analysis part of the optimisation component in the UPSOM framework. When the jobs were assigned as a policy scenario, the number of jobs in each zone within the database was updated by adding the new jobs that were to be created for the policy scenario considered. The simulation data set was therefore altered. The travel data remained constant and were employed to re-calculated the number of jobs within the 20 minute commute feature of the simulation dataset. The updating of the simulation data set, therefore, represents what the conditions would have been had the policy been implemented as described by that particular policy scenario. As mentioned in §4.3.2, the computation of an accessibility or proximity feature adds complexity to the optimisation procedure because the feature value for every scenario varies over zones in which the policy is not even being implemented. Accessibility and proximity features therefore create trade-offs in attractiveness when the policy elements are assigned. Some priority areas in the case study did not have any job vacant capacity, however, due to the feature being an accessibility feature, and the attractiveness of these zones could also be altered.

5.3.3 Policy scenario analysis

Following the construction of a policy scenario and the accompanying feature calculation, the quality of the policy scenario had to be assessed. Prior to the scoring of the quality of a policy scenario, the probability vector \mathbf{P} was computed, as described in §4.3.2. In this case study, the location choice model within the updated version of UrbanSim was employed for this purpose, as explained in §4.3.2. The probability vector was computed for both real-estate unit types 201 and 202 separately, and are denoted by \mathbf{P}_{201} and \mathbf{P}_{202} , respectively. Each vector was computed by employing the relevant coefficients estimated in §5.2.2. The probability vectors contain the probability of each locational alternative being considered as the location for the

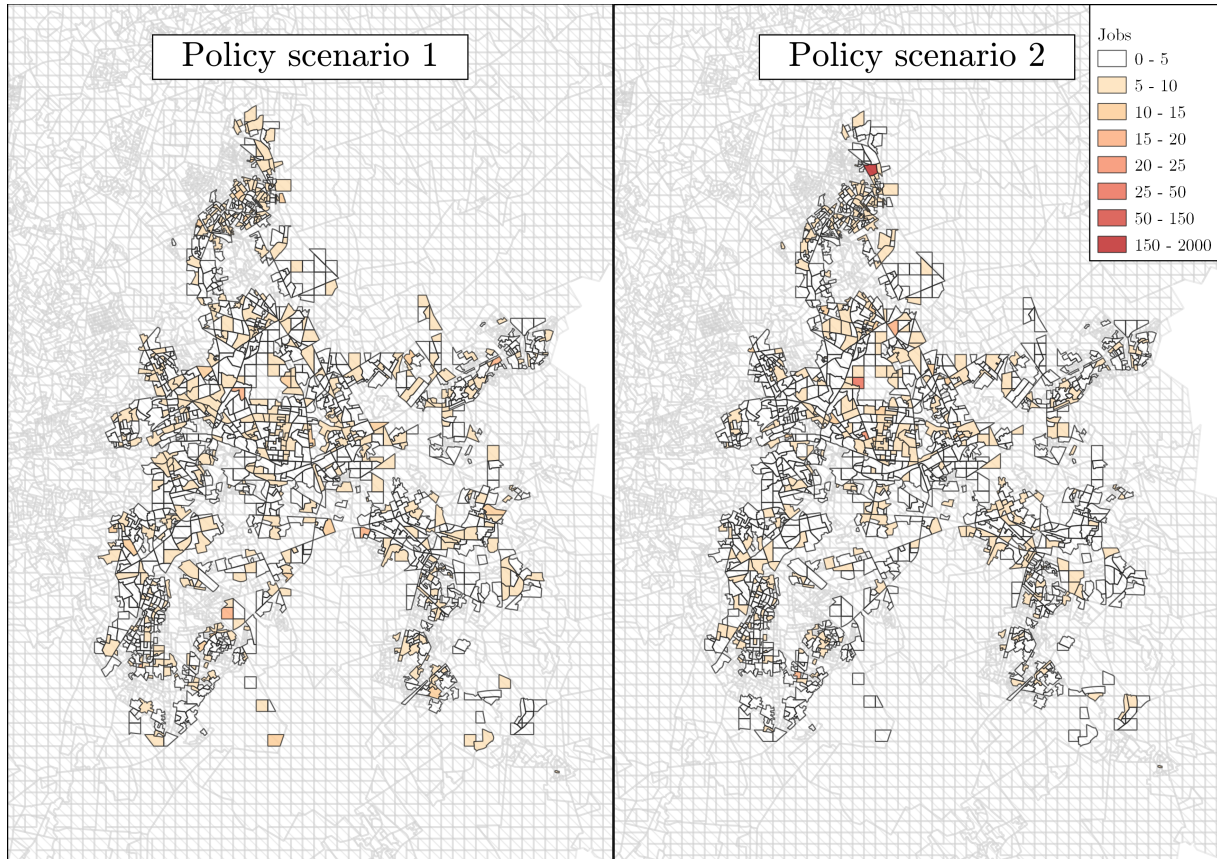


FIGURE 5.5: Policy scenarios exhibiting variation in policy element concentration.

construction of a single residential real-estate unit, for both types 201 and 202 independently. During the computation of the probability vectors in this case study, every zone within Gauteng was considered as an alternative. The probabilities of zones within Gauteng being considered for the construction of a real-estate unit of type 201 or 202 were computed separately.

The probability vectors were employed to calculate the fitness function values used to assess a policy scenario. The UPSOM framework is intended for maximisation problems and, as described in §4.3.3, the fitness function is a weighted sum of the aggregated probabilities of priority areas being selected for real-estate development, which has to be maximised. The priority area probability subvector $\mathbf{P}(Z)$ was computed for each real-estate type and is denoted by $\mathbf{P}(Z_{201})$ and $\mathbf{P}(Z_{202})$, respectively.

The fitness values associated with each policy scenario was then calculated. The fitness function (4.6) was employed to assess the suitability of a policy scenario. The weighting of each real-estate type w_g was considered to be equal in this case study and are denoted by w_{201} and w_{202} . The objective function for this case study may therefore be expressed mathematically as

$$\max f(ps) = w_{201} \sum \mathbf{P}(Z_{201}) + w_{202} \sum \mathbf{P}(Z_{202}). \quad (5.1)$$

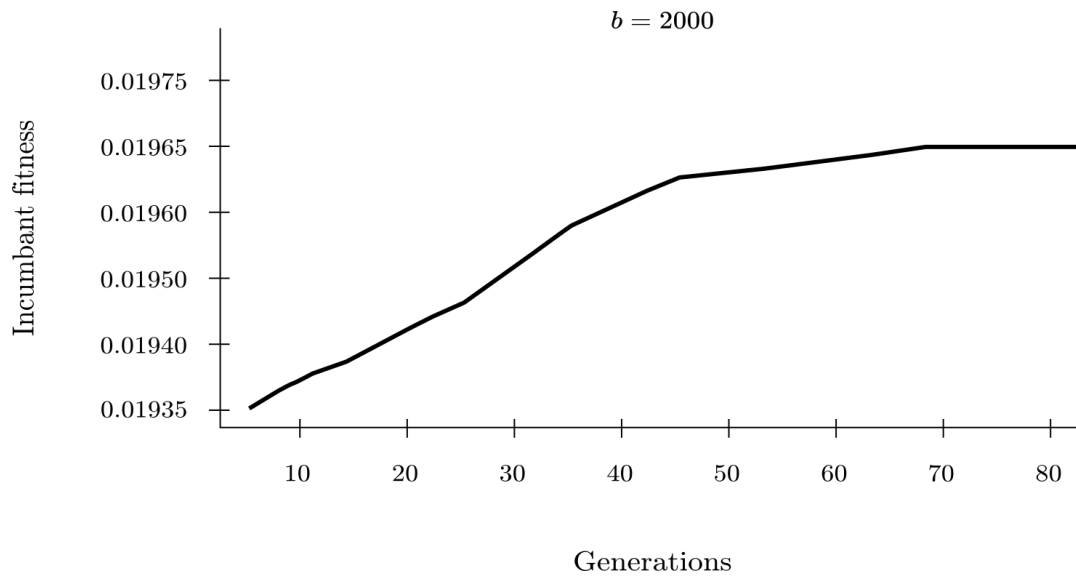


FIGURE 5.6: The fitness values of the incumbent solutions as a function of generational iterations during an application of the UPSOM framework for $b = 2000$.

5.4 Results and discussion

Recall that the optimisation component of the UPSOM framework employs a GA, as described in §4.4, in an attempt to construct an urban policy which may be implemented in the best way possible. In this case study, the framework was implemented for two different budget constraint amounts, namely $b = 2000$ and $b = 3000$. The two corresponding optimisation processes produced different job assignment sets for the implementation of the policy.

The parameters of the GA employed in this case study were as follows. The initial population size was taken as 40 solutions, the maximum number of generations that could elapse without a new incumbent solution being encountered was taken as eight (the stopping criterion), the probability of a crossover operation being performed was taken $p_c = 0.8$ and the probability of a mutation being performed was taken as $p_m = 0.1$. Moreover, the number of cuts made during the crossover operator was $\alpha = 20$. For the population initialisation described in §4.4.1, the fraction employed during the SSI approach was taken as $r = 0.4$ and the size of the solution set employed for the initialisation process was taken as $q = 50$.

The UPSOM framework was implemented for each variation of the budget constraint, as mentioned above. For the budget constraints $b = 2000$ and $b = 3000$, the incumbent fitness function values for each generational iteration of the optimisation process are presented graphically in Figures 5.6 and 5.9, respectively.

The final solution sets \mathcal{S}_{ps} produced during these implementations of the UPSOM framework in the form of optimal zones, described in §4.5, have fitness function values of 0.01965 and 0.02030, respectively. The number of policy elements assigned, in the form of jobs, associated with these solutions, as well as the locations of the assignments, are illustrated in Figures 5.8 and 5.9.

The sizes of the job allocations and the frequencies of the job assignments vary for the different implementations of the UPSOM framework. It was observed that some solutions consisted of more concentrated assignments of policy elements whilst other solutions seemed to be more evenly spread out. The sizes and frequencies of the two best solutions produced in this case study are illustrated in Figures 5.10 and 5.11.

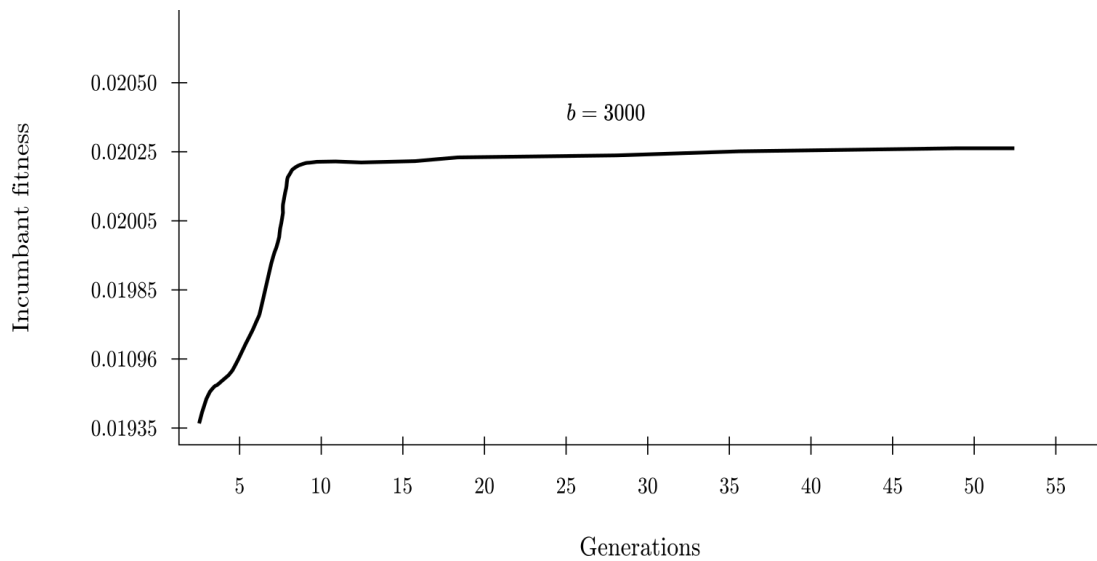


FIGURE 5.7: The fitness values of the incumbent solutions as a function of generational iterations during an application of the UPSOM framework for $b = 3000$.

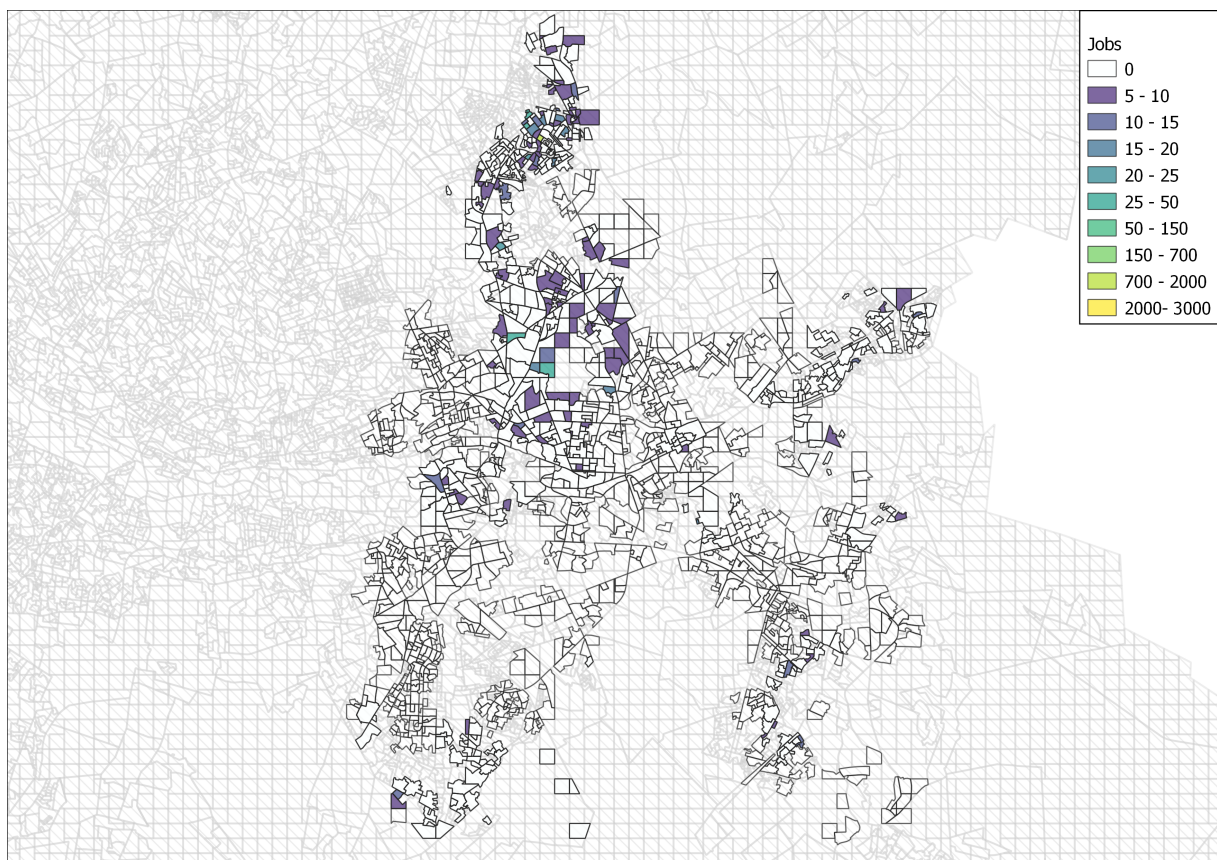


FIGURE 5.8: Assignment of policy elements in the form of jobs in the incumbent solution during an implementation of the UPSOM framework for $b = 2000$.

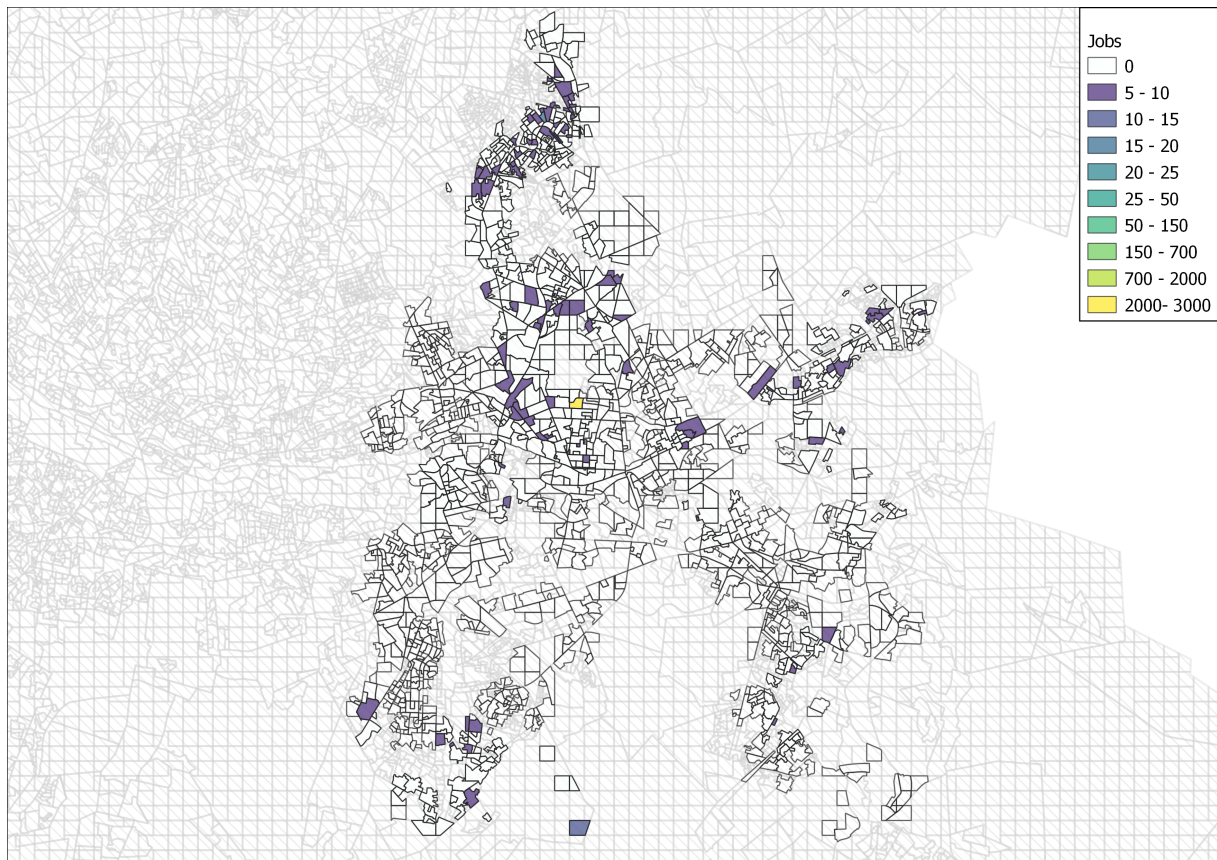


FIGURE 5.9: Assignment of policy elements in the form of jobs in the incumbent solution during an implementation of the UPSOM framework for $b = 3000$.

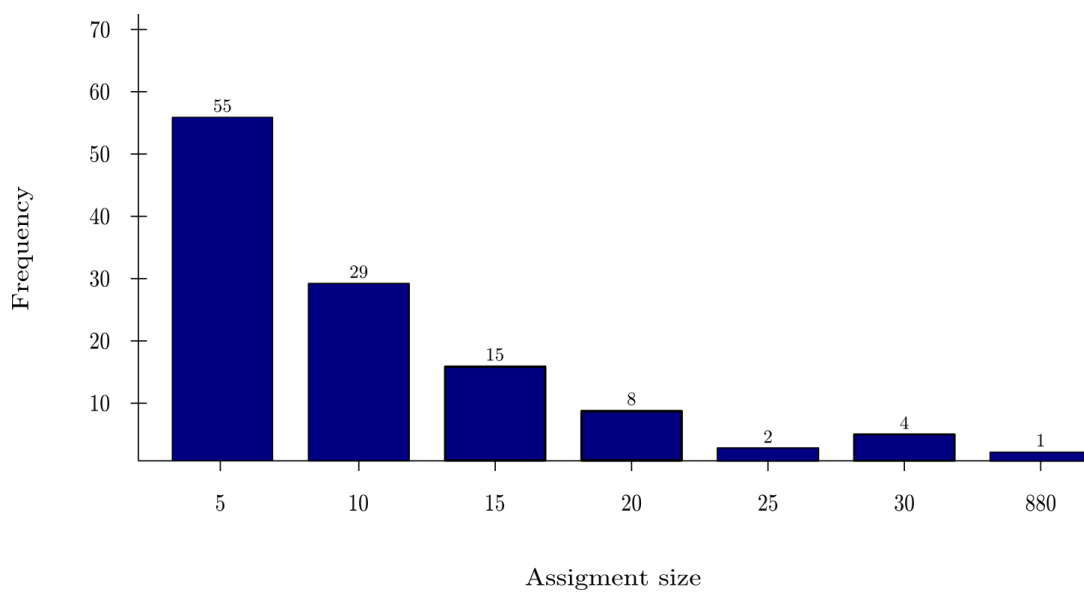


FIGURE 5.10: The sizes of policy elements assigned to zones in the form of number of jobs during an implementation of the UPSOM framework for $b = 2000$.

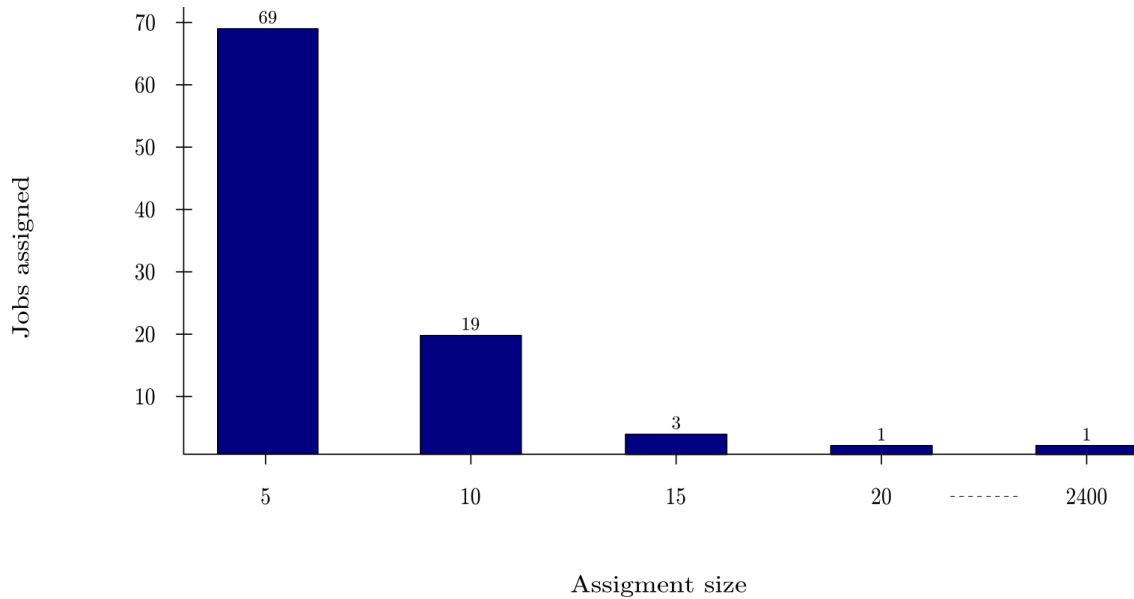


FIGURE 5.11: The sizes of policy elements assigned to zones in the form of number of jobs during an implementation of the UPSOM framework for $b = 3000$.

As mentioned above, the fitness function in (5.1), which was maximised, is a weighted sum of aggregated probabilities of priority areas being selected for real-estate development. For the purposes of this case study, the fraction of all the residential real-estate units to be developed during the simulation year within priority areas is double the fitness function of the solution. This is due to the weighting of each real-estate type being taken as 0.5. The alternatives for construction of the real-estate units are the remaining zonal areas in the Gauteng province, as mentioned in §5.3.3. When considering the large number of zones in Gauteng (28 460, as stated in §5.2.1), the relatively small values associated with the fitness functions of the policy scenarios computed coincide with the scale of the number of alternatives. Moreover, as stated in §5.1, a vast number of residential real-estate units are constructed annually in Gauteng. A small difference in the fitness function would, therefore, impact the construction of a reasonable number of residential real-estate units.

For a perspective on the results obtained, the number of residential real-estate units being completed annually may be assumed to be the sum total of the numbers of dwelling houses, flats and town houses completed, as stated in §5.1. It is therefore assumed that 18 206 residential real-estate units are constructed annually in Gauteng. In addition, 5 440 of these real-estate units are dwellings smaller than 80 square metres in area [75], which have an average price of R 308 000 per unit. For purposes of illustration of the results obtained, it is assumed that half of the dwelling houses smaller than 80 square metres in area, cost more than R 400 000. The number of residential real-estate units of type 201 or 202 built annually in Gauteng is, therefore, assumed to be 15 486.

Fitness function values were calculated for a set of stochastically created policy scenarios. This set of scenarios represents the implementation of a policy without employing the UPSOM framework. A total of 50 policy scenarios were constructed randomly, achieving an average fitness function value of 0.01930 (resulting in the construction of 598 units in priority areas), with the best solution achieving a fitness function value of 0.01940 (resulting in the construction of 601 units in priority areas). The results obtained by applying the UPSOM framework were compared to this situation in which policies were implemented without the benefit of the optimisation component of the UPSOM framework. The assumed total number of real-estate units of type 201 or

202 constructed annually, namely 15 486, was taken as the control total as a means to assess the results obtained *via* the UPSOM framework. The number of residential real-estate units which are expected to be constructed additionally in priority areas as a result of applying the UPSOM framework may be found in Tables 5.4 and 5.5 for budget values of $b = 2\,000$ and $b = 3\,000$, respectively.

TABLE 5.4: *Improvements upon randomly generated results produced during four applications of the UPSOM framework, given a budget of $b = 2\,000$.*

Incumbent	Units built	Improvement (best) %	Improvement (average) %	Improvement on (average) units
0.01965	609	1.29 %	1.8 %	11
0.01950	604	0.52 %	1.0 %	6
0.01967	609	1.39 %	1.9 %	8
0.01999	619	3.04%	3.6 %	21

TABLE 5.5: *Improvements upon randomly generated results produced during four applications of the UPSOM framework, given a budget of $b = 3\,000$.*

Incumbent	Units built	Improvement (best) %	Improvement (average) %	Improvement (average) units
0.02020	626	4.12 %	4.66 %	28
0.02100	650	8.25 %	8.81 %	52
0.02149	666	10.77 %	11.35 %	68
0.02056	637	5.98 %	6.53 %	39

From the results observed in Tables 5.4 and 5.5 it would seem that various implementations of the UPSOM framework resulted in significant improvements in respect of the incentivisation of the development of residential real-estate in priority areas. When a budget constraint of $b = 3\,000$ was employed, an especially noteworthy improvement in the attractiveness of priority areas was observed. Given the improvements observed when implementing the UPSOM framework over the course of a single year, such as observed above, it may be hypothesised that the annual use of the framework may result in an exponential effect in promoting the utility of priority areas in the long term.

5.5 Chapter summary

This chapter was devoted to a detailed discussion of a case study in which the UPSOM framework proposed in this thesis was applied in the context of real data. The chapter opened with a discussion in §5.1 on the geographical background of the area for which the case study was performed. The process of applying the framework preprocessing component to the case study data was then discussed in §5.2. The construction, implementation and analysis of solutions returned by the UPSOM framework in the form of policy scenarios were discussed thereafter in §5.3 in some detail. The discussion finally turned to a brief review in §5.4 of the application of the framework's optimisation component as well as the final results produced by the framework when implemented in the context of the case study.

CHAPTER 6

Conclusion

Contents

6.1	Thesis summary	89
6.2	Appraisal of thesis contributions	90
6.3	Suggestions for future work	92

The purpose of this final chapter of the thesis is threefold. The chapter opens with a summary of the contents of the thesis. An appraisal follows of the contributions of this thesis. Suggestions are finally made as to possible future work that may be pursued as natural follow-up investigations on the contributions of this thesis.

6.1 Thesis summary

This thesis opened with an introductory chapter, Chapter 1, which provided the reader with a background contextualising the problem considered. The origin of the segregated South African city was described, as were the resulting deleterious effects on the lives of its citizens. The needs for spatial transformation plans pertaining to South African cities and computerised models capable of assisting with urban spatial planning were motivated. This was followed by a formal description of the problem considered in this thesis. Moreover, the seven objectives pursued in this study were outlined. The scope of the study, in terms of changing of spatial features, the assignment of policy elements and the use of control totals, was delimited thereafter. The research methodology employed during the study was elaborated upon and the chapter finally closed with a description of the organisation of material in the thesis.

Two literature review chapters followed the introductory chapter. Chapter 2 was devoted to a discussion on relevant preliminary mathematical concepts from the literature aimed at familiarising the reader with notions that underpin the material presented in subsequent chapters of the thesis. More specifically, mathematical techniques and statistical distributions employed later in the thesis to quantify the attractiveness of spatial features and simulate decision making were reviewed in fulfilment of Objective I(a) of §1.3. In addition, combinatorial optimisation techniques which may be employed to solve urban planning models were identified and discussed in fulfilment of Objective I(b). One of these techniques, the GA, was implemented during incentivisation optimisation in later chapters.

Chapter 3 provided the reader with a basic insight into the area of urban simulation and the tools available in this environment. The evolution of urban simulation tools over time was

discussed with reference to various ITLUMs in the literature, in pursuit of Objective I(c). The discourse then focused on a comprehensive description of a particular ITLUM, called UrbanSim, which was also used in later chapters of the thesis. The location choice models embedded in UrbanSim were therefore described in some detail. The material presented in Chapter 3 stands in fulfilment of Objective I(d).

The novel UPSOM framework proposed in this thesis was presented in detail in Chapter 4. The framework is generic in nature and facilitates incentivisation policy generation aimed at increasing residential real-estate development in prioritised urban areas. The chapter opened with a high-level overview of the framework architecture in pursuit of Objective II. An in-depth description of the two main components of the framework followed, namely its data preprocessing and optimisation components, as well as the various modules embedded in these components. An accompanying description of a suitable course of action during the design of urban policies employed in the framework was established in fulfilment of Objective III(a)–(c). The chapter contained discussions on the design, and various design considerations involved in the development, of the UPSOM framework and concluded with a discussion on the nature of typical results returned by the framework.

Chapter 5 was devoted to a detailed description of a case study in which the UPSOM framework was applied as a proof of concept to data pertaining to the City of Ekurhuleni. The reader was provided with a background context on the case study geographical area. The application of the various components and modules of the framework was described, as was the course of action adopted during the construction of policy scenarios. A description was also provided of the optimisation process within the UPSOM framework in respect of urban policy scenario generation within the context of the case study, in fulfilment of Objective IV. The results produced were subsequently interpreted and discussed in fulfilment of Objective V. The results returned by various applications of the UPSOM framework were subsequently compared with multiple random implementations of policies in order to verify and validate that the UPSOM framework is capable of outperforming arbitrary implementations of policies, in fulfilment of Objective VI.

6.2 Appraisal of thesis contributions

The contributions of this thesis are five-fold. This section is dedicated to a brief overview and summary of these contributions.

Contribution I *Comprehensive research on spatial planning tools and their developments over time with a specific focus on the UrbanSim software suite*

A comprehensive study of spatial planning tools and their development over time form part of the research contributions of this thesis. The development of spatial planning tools was reviewed in Chapter 3, and various perspectives were proffered of how spatial development tools have developed and improved over time as a result of a variety of contributions in the literature [69, 84, 89]. The review included a comparison of ITLUMs in the literature. Moreover, a detailed description of the simulation software suite UrbanSim was provided, drawing from various literature resources in order to provide a comprehensive overview of the development and functioning of this particular ITLUM.

Contribution II *The proposal of a generic framework for optimising the implementation of urban densification policies*

The urban densification policy scenario optimisation framework proposed in Chapter 4 constitutes an initial attempt at the optimisation of the implementation of urban policy scenarios for incentivising densification of residential real-estate development in priority areas. Adopting a modular approach during the design of the UPSOM framework allowed the framework to be generic in nature. The various components and modules which form part of the framework may, therefore, be replaced by or exchanged for other mathematical techniques available in the literature. In cases where the mathematical techniques embedded in the framework are exchanged, the replacing techniques should, of course, adhere to the requirements of the framework in order to be implemented effectively. Existing ITLUMs have typically been employed in the literature to quantify how various policy alternatives compare with regard to the potential spatial compositions of urban areas in the future. The UPSOM framework further contributes to urban policy selection by providing a mechanism for optimising the implementation of a policy subsequent to the policy being selected for implementation. The framework is therefore capable of providing decision support with respect to the implementation of urban densification policies.

Contribution III *Application of the proposed framework to a real-world case study involving a South African city*

The pragmatic applicability of the framework was illustrated by applying a computerised instantiation of it to a real-world case study as a proof of concept, as described in Chapter 5. It was demonstrated in this case study that the framework is capable of contributing successfully to the encouragement of densification of residential real-estate development within priority areas in a real-world application. By applying the framework to assist with the objectives stated in the MSDF for the City of Ekurhuleni, the framework, as an initial attempt at the optimisation of policy scenario implementation, has proved to be workable and practical. The particular case study area was selected due to Gauteng province experiencing rapid growth whilst simultaneously not effectively integrating numerous residents spatially into society.

Contribution IV *Illustration of the UPSOM framework as a proof of concept aimed at encouraging further census type data collection*

The aforementioned successful application of the UPSOM framework illustrates that the framework may be implemented pragmatically within a real-world context. A requirement for the application of the framework, however, is the availability of large census-type data sets over multiple years. A more effective implementation of the framework would require more detailed data sets than are currently available. A historical validation of the regression coefficients estimated would provide more confidence in the results returned by the framework. Such a validation would require additional data sets. The UPSOM framework and its application to the real-world case study based on the City of Ekurhuleni stand as a proof of concept which motivates the further collection of comprehensive census-type data sets over multiple years.

Contribution V *Suggestion of potential future projects following on contributions by the thesis*

The final contribution of this thesis is presented in the next section of this chapter, where recommendations are made in respect of further improving the research documented in this thesis. These suggestions may assist researchers in determining the direction of their future work and stand in fulfilment of Objective VII of §1.3.

6.3 Suggestions for future work

Following on the contributions of this thesis, a number of suggestions have been crafted for potential future follow-up work. These suggestions are aimed at improving of the novel UPSOM framework proposed in this thesis.

Suggestion I *Incorporate alternative feature selection techniques within the coefficient estimation module as part of the preprocessing component of the UPSOM framework*

The features selected for regression coefficient estimation within the UPSOM framework were selected through a process of consulting the literature as well as by carrying out a structured empirical comparison between various combinations of feature sets identified from the literature. Due to the generic nature of the UPSOM framework, the adoption of more structured analytical approaches towards feature selection may result in regression coefficients being estimated more accurately. If sufficient data are available, the application of machine learning techniques may perhaps produce coefficients that better assess locational attractiveness for real-estate development. With a more realistic assessment of real-estate attractiveness in hand, the fitness function calculation within the framework may also become more accurate, resulting in improved policy scenarios constructed by the framework.

Suggestion II *Incorporate alternative optimisation techniques within the optimisation component of the UPSOM framework*

In its current form, the UPSOM framework incorporates a GA within its optimisation component. The GA was selected due to the fact that it is capable of exploring a large solution space effectively and can easily be executed whilst adhering to the constraints involved in the construction of solutions in the form of policy scenarios. A drawback of the GA, however, is that the algorithm is less adept at exploiting the neighbourhoods of good solutions and hence requires numerous iterations in order to produce very high-quality solutions. This presents a challenge due to the computation of the fitness function of a solution being relatively computationally expensive. This computational burden is incurred during the recalculation of proximity-and-accessibility features and the subsequent use of an ITLUM to produce priority vectors. In order to be able to produce high-quality results involving a larger scenario area set (*i.e.* a larger radius around priority areas) or larger policy element assignment constraints (*i.e.* the value of b), other less computationally expensive optimisation algorithms may be required. Designing a computationally less expensive optimisation component for the UPSOM framework would also allow for better parameter tuning so as to produce higher-quality results.

Suggestion III *Adapt the UPSOM framework to accommodate a multi-objective approach towards policy scenario optimisation*

The UPSOM framework in its current form has been designed as a single-objective optimisation framework. It has the objective of densification of all the priority areas, which already involves various trade-offs when implemented. A multi-objective optimisation approach may, however, improve the practicality of the policy scenarios produced. As opposed to incorporating the various residential real-estate types into a single-objective optimisation problem, the structuring thereof as a multi-objective optimisation problem is expected to yield improved trade-off results in the form of a Pareto set. For a transition to multi-objective optimisation, the optimisation component of the UPSOM framework will have to employ a multi-objective optimisation technique such as a non-dominated sorting GA.

Suggestion IV *Adaptation of the UPSOM framework for a multi-period temporal urban policy implementation problem*

The initial design of the UPSOM framework was conducted by considering only a single year of urban policy implementation. It was hinted in Chapter 5 that the re-application of the UPSOM framework on an annual basis may result in an exponential increase in attractiveness of prioritised areas. If the framework is adapted so that its optimisation component is capable of multi-year predictions by ITLUMs during the generation of policy scenarios, temporal factors may be incorporated to assist with higher-quality urban policy implementation. This is desirable considering the long-term spatial planning requirements of a city.

References

- [1] ALONSO W, 1964, *Location and land use: Toward a general theory of land rent*, Harvard University Press, Cambridge (MA).
- [2] ANONYMOUS, *Monte Carlo simulation*, [Online], [Cited October 2020], Available from https://www.palisade.com/risk/monte_carlo_simulation.asp.
- [3] ANONYMOUS, 2010, *Nelson Mandela released from prison*, [Online], [Cited March 2019], Available from <https://www.history.com/this-day-in-history/nelson-mandela-released-from-prison>.
- [4] ANONYMOUS, 2015, *Sophiatown — The mix*, [Online], [Cited March 2019], Available from <https://www.portfoliocollection.com/visit/sophiatown-the-mix>.
- [5] AXELSON E, 2019, *Cape Town, national legislative capital, South Africa*, [Online], [Cited January 2019], Available from <https://www.britannica.com/place/Cape-Town#ref10617>.
- [6] BARBARIN O & RICHTER L, 2013, *Mandela's children: Growing up in post-apartheid South Africa*, Routledge, New York (NY).
- [7] BARD J & MOORE J, 1990, *A branch and bound algorithm for the bilevel programming problem*, SIAM Journal on Scientific and Statistical Computing, **11**(2), pp. 281–292.
- [8] BARONE A, 2020, *Binomial distribution*, [Online], [Cited October 2020], Available from <https://www.investopedia.com/terms/b/binomialdistribution.asp#:~:text=Binomial%20distribution%20summarizes%20the%20number,a%20specified%20number%20of%20trials..>
- [9] BROHIER D, 1985, *Who are the Burghers?*, Journal of the Royal Asiatic Society — Sri Lanka Branch, **30**, pp. 101–119.
- [10] BROOKS-BARTLETT J, 2018, *Probability concepts explained: Maximum likelihood estimation*, [Online], [Cited October 2020], Available from <https://towardsdatascience.com/probability-concepts-explained-maximum-likelihood-estimation-c7b4342fdbb1>.
- [11] CHELOUAH R & SIARRY P, 2000, *A continuous genetic algorithm designed for the global optimization of multimodal functions*, Journal of Heuristics, **6**(2), pp. 191–213.
- [12] CITY OF EKURHULENI, 2015, *Metropolitan spatial development framework: 2015*, (Unpublished) Technical Report, City of Ekurhuleni, Ekurhuleni.
- [13] CITY OF EKURHULENI, 2020, *City of Ekurhuleni Metropolitan GAU*, (Unpublished) Technical Report EK/52, City of Ekurhuleni, Ekurhuleni.
- [14] DARWIN C, 1859, *The origin of species by means of natural selection: Or, the preservation of favored races in the struggle for life*, John Murray, London.
- [15] DIGGLE P, 1983, *Statistical analysis of spatial point patterns*, Academic Press, New York City (NY).

- [16] DiPASQUALE D & WHEATON W, 1996, *Urban economics and real-estate markets*, Prentice Hall, Englewood Cliffs (NJ).
- [17] DORIGO M, 1992, *Optimization, learning and natural algorithms (in Italian)*, PhD thesis, Politecnico di Milano, Milan.
- [18] DOWLING RG, IRESON R, SKABARDONIS A, GILLEN D & STOPHER P, 2005, *Predicting air quality effects of traffic-flow improvements: Final report and user's guide*, Transportation Research Board, Washington (DC).
- [19] DUTHIE J, KOCKELMAN K, VALSARAJ V & ZHOU B, 2007, *Applications of integrated models of land use and transport: A comparison of ITLUP and UrbanSim land use models*, Proceedings of the 54th Annual North American Meetings of the Regional Science Association International, Savannah (GA), No page numbers.
- [20] ECHENIQUE M, FLOWERDEW A, HUNT J, MAYO T, SKIDMORE I & SIMMONDS D, 1990, *The MEPLAN models of Bilbao, Leeds and Dortmund*, Transport Reviews, **10(4)**, pp. 309–322.
- [21] ELIASON S, 1993, *Maximum likelihood estimation logic and practice*, SAGE Publications, Newbury Park (CA).
- [22] ENCYCLOPAEDIA BRITANNICA, 2015, *Dutch East India Company*, [Online], [Cited November 2019], Available from <https://www.britannica.com/topic/Dutch-East-India-Company>.
- [23] ENCYCLOPAEDIA BRITANNICA, 2017, *Gauteng*, [Online], [Cited November 2020], Available from <https://www.britannica.com/place/Gauteng>.
- [24] FAN Z, 2016, *STATS 200: Introduction to statistical inference — Poisson regression*, Lecture Notes, Stanford University, Stanford (CA).
- [25] FERNANDO J, 2020, *Multinomial distribution*, [Online], [Cited October 2019], Available from <https://www.investopedia.com/terms/m/multinomial-distribution.asp>.
- [26] FROST J, 2020, *Regression coefficients*, [Online], [Cited October 2020], Available from <https://statisticsbyjim.com/glossary/regression-coefficient/>.
- [27] GLEN S, 2016, *Gumbel Distribution: Definition, Examples*, [Online], [Cited October 2020], Available from <https://www.statisticshowto.com/gumbel-distribution/>.
- [28] GLOVER F, 1986, *Future paths for integer programming and links to artificial intelligence*, Computers and Operations Research, **13(5)**, pp. 533–549.
- [29] GOLIAS M, MISHRA S & PSARROS I, 2014, *A guidebook for best practices on integrated land use and travel demand modelling*, University of Memphis Press, Memphis (TN).
- [30] HABIB M & MILLER E, 2009, *Reference-dependent residential location choice model within a relocation context*, Transportation Research Record, **2133(1)**, pp. 92–99.
- [31] HALABISKY B, 1999, *Classification of neuronal data*, [Online], [Cited November 2020], Available from https://hlab.stanford.edu/brian/cluster_analysis.html.
- [32] HEERDEN QV, 2015, *Urban Simulation*, [Online], [Cited March 2019], Available from http://stepsa.org/urban_sim_background.html.
- [33] HILLIER FS, 2012, *Introduction to operations research*, Tata McGraw-Hill Education, New York (NY).
- [34] HOLLAND JH, 1975, *Adaptation in natural and artificial systems*, University of Michigan Press, Ann Arbor (MI).

- [35] HUNT J, KRIGER D & MILLER E, 2005, *Current operational urban land-use-transport modelling frameworks: A review*, *Transport Reviews*, **25(3)**, pp. 329–376.
- [36] HUSSAIN K, SALLEH M, CHENG S & SHI Y, 2019, *Metaheuristic research: A comprehensive survey*, *Artificial Intelligence Review*, **52(4)**, pp. 2191–2233.
- [37] IZIKO MUSEUMS OF SOUTH AFRICA, 2019, *Inhabitants of the Lodge*, [Online], [Cited March 2019], Available from <https://slavery.iziko.org.za/inhabitantsofthelodge>.
- [38] JANSEN JD, 2009, *Knowledge in the blood: Confronting race and the apartheid past*, Stanford University Press, Stanford (CA).
- [39] KAMER L, 2020, *Total population of South Africa 2018, by province*, [Online], [Cited November 2020], Available from <https://www.statista.com/statistics/1112169/total-population-of-south-africa-by-province/>.
- [40] KENNEDY J & EBERHART R, 1995, *Particle swarm optimization*, Proceedings of the ICNN'95 International Conference on Neural Networks, Washington (DC), pp. 1942–1948.
- [41] KERR A, 2015, *Commuting costs the poor dearly*, [Online], [Cited February 2019], Available from <https://mg.co.za/article/2015-11-15-commuting-costs-the-poor-dearly>.
- [42] KIRKPATRICK S, GELATT C & VECCHI M, 1983, *Optimization by simulated annealing*, *Science*, **220(4598)**, pp. 671–680.
- [43] KLUSMAN A, 2012, *Slegs vir blankes*, [Online], [Cited March 2019], Available from <https://bkbcampaignwatch.nl/slegs-vir-blankes/>.
- [44] KOHLHEPP DB & KOHLHEPP KJ, 2012, *Real estate development matrix*, Routledge, St. Petersburg (FL).
- [45] KURI-SEBINA G, 2016, *State of South African Cities Report*, (Unpublished) Technical Report ISBN No. 978-0-620-71463-1, South African Cities Network, Johannesburg.
- [46] LAND A & DOIG A, 2010, *An automatic method for solving discrete programming problems*, pp. 105–132 in JÜNGER M, LIEBLING T, NADDEF D, NEMHAUSER G, PULLEYBLANK W, REINELT G, RINALDI G & WOLSEY L (EDS), *50 Years of integer programming 1958–2008*, Springer, Heidelberg.
- [47] LÖTTER DP, 2017, *Design of a weapon assignment subsystem within a ground-based air defence environment*, PhD thesis, Stellenbosch University, Stellenbosch.
- [48] LOWRY I, 1964, *A model of metropolis*, (Unpublished) Technical Report No. RM-40535-RC, Rand Corporation, Santa Monica (CA).
- [49] LOZANO M, HERRERA F & CANO J, 2008, *Replacement strategies to preserve useful diversity in steady-state genetic algorithms*, *Information Sciences*, **178(23)**, pp. 4421–4433.
- [50] MAARANEN H, MIETTINEN K & PENTTINEN A, 2007, *On initial populations of a genetic algorithm for continuous optimization problems*, *Journal of Global Optimization*, **37(3)**, pp. 405.
- [51] MAHARAJ S, 2014, *Economics of South African townships — Special focus on Diepsloot*, (Unpublished) Technical Report ISBN No. 978-1-4648-0302-4, World Bank Group, Washington (DC).
- [52] MARTINEZ F, 1996, *MUSSA: Land use model for Santiago city*, *Transportation Research Record*, **1552(1)**, pp. 126–134.
- [53] MAYLAM P, 1995, *Explaining the apartheid city: 20 years of South African urban historiography*, *Journal of Southern African Studies*, **21(1)**, pp. 19–38.

- [54] MCCLELAND D, 2017, *Port Elizabeth of Yore: Early black settlements — Part 1*, [Online], [Cited February 2019], Available from <http://thecasualobserver.co.za/port-elizabeth-yore-early-black-settlements/>.
- [55] MCFADDEN D, 1973, *Conditional logit analysis of qualitative choice behaviour*, pp. 105–142 in ZAREMBKA P (ED), *Frontiers in econometrics*, Academic Press, New York (NY).
- [56] MCKENNA A, 2009, *Cape Colony, British Colony, South Africa*, [Online], [Cited March 2019], Available from <https://www.britannica.com/place/Cape-Colony>.
- [57] METROPOLIS N, ROSENBLUTH A, ROSENBLUTH M, TELLER A & TELLER E, 1953, *Equation of state calculations by fast computing machines*, *Journal of Chemical Physics*, **21(6)**, pp. 1087–1092.
- [58] NOTH M, BORNING A & WADDELL P, 2003, *An extensible, modular architecture for simulating urban development, transportation, and environmental impacts*, *Computers, Environment and Urban Systems*, **27(2)**, pp. 181–203.
- [59] O’NEIL R & HOFFMAN K, 2018, *Exact methods for solving traveling salesman problems with pickup and delivery in real time*, [Online], [Cited October 2020], Available from [Optimization-Online.%20org:%20http://www.%20optimization-online.%20org/DB%5C_HTML/2017/12/6370.%20html.%20Accessed](http://www.optimization-online.org/DB_HTML/2017/12/6370.html).
- [60] OLIVEIRA J & CARRAVILLA M, 2009, *Heuristics and local search*, Lecture Notes, Faculdade de Engenharia da Universidade do Porto, Porto.
- [61] PAMLER J & ROOT C, *South Africa overcoming Apartheid, building democracy*, [Online], [Cited February 2019], Available from <http://overcomingapartheid.msu.edu/multimedia.php?id=65-259-6>.
- [62] PATTERSON Z & BIERLAIRE M, 2010, *Development of prototype UrbanSim models*, *Environment and Planning B: Planning and Design*, **37(2)**, pp. 344–366.
- [63] PEASE C, 2018, *Recalling District Six*, [Online], [Cited October 2020], Available from <https://towardsdatascience.com/an-overview-of-monte-carlo-methods-675384eb1694>.
- [64] PUTNAM S, 1983, *Integrated urban models: Policy analysis of transportation and land use*, Pion, London.
- [65] RONG Q, 1982, *Constructive heuristic methods*, Lecture Notes, University of Nottingham, Nottingham.
- [66] ROUTLEDGE R, 2020, *Poisson distribution*, [Online], [Cited October 2020], Available from <https://www.britannica.com/topic/Poisson-distribution>.
- [67] SAINI N, 2017, *Review of selection methods in genetic algorithms*, *International Journal of Engineering and Computer Science*, **6(12)**, pp. 22261–22263.
- [68] SAPA, 2011, *Government is SA’s biggest employer, jobs creator*, [Online], [Cited November 2020], Available from <https://www.timeslive.co.za/politics/2011-10-25-government-is-sas-biggest-employer-jobs-creator/>.
- [69] SARGENT R, 2011, *Advanced tutorials: Verification and validation of simulation models*, *Proceedings of the 2011 Winter Simulation Conference*, Syracuse (NY), pp. 183–198.
- [70] SARGENT T & STACHURSKI J, 2020, *Data and empirics*, [Online], [Cited October 2020], Available from <https://python.quantecon.org/mle.html>.
- [71] SOUTH AFRICAN HISTORY ONLINE, 2011, *Cape Town the segregated city*, [Online], [Cited March 2019], Available from <https://www.sahistory.org.za/article/cape-town-segregated-city>.

- [72] SOUTH AFRICAN NATIONAL BIODIVERSITY INSTITUTE, *Kirstenbosch NBG: Van Riebeeck's hedge*, [Online], [Cited March 2019], Available from <https://www.sanbi.org/gardens/kirstenbosch/history/kirstenbosch-nbg-van-riebeecks-hedge/>.
- [73] SOUTHWORTH F, 1995, *A technical review of urban land use transportation models as tools for evaluating vehicle travel reduction strategies*, (Unpublished) Technical Report ORNL-6881, Oak Ridge National Laboratory, Oak Ridge (TN).
- [74] SPAULL N, 2015, *Schooling in South Africa: How low-quality education becomes a poverty trap*, South African Child Gauge, **12**, pp. 34–41.
- [75] STATISTICS SOUTH AFRICA, 2018, *Building statistics, 2018*, (Unpublished) Technical Report 50-11-01, Statistics South Africa, Pretoria.
- [76] STATISTICS SOUTH AFRICA, 2020, *Census*, [Online], [Cited November 2020], Available from http://www.statssa.gov.za/?page_id=3836#:~:text=A%20population%20census%20is%20typically,place%20of%20the%202006%20census.
- [77] STATISTICS SOUTH AFRICA, 2020, *Mid-year population estimates*, (Unpublished) Technical Report P0302, Statistics South Africa, Pretoria.
- [78] STATISTICS SOUTH AFRICA, 2018, *Mid-year population estimates*, (Unpublished) Technical Report P0302, Statistics South Africa, Pretoria.
- [79] TALBI E, 2009, *Metaheuristics: From design to implementation*, John Wiley & Sons, Hoboken (NJ).
- [80] TOBIN J, 1969, *A general equilibrium approach to monetary theory*, Journal of Money, Credit and Banking, **1(1)**, pp. 15–29.
- [81] URBANSIM, 2014, *Urban Sim*, [Online], [Cited March 2020], Available from <https://urbansim.com/urbansim>.
- [82] VUJIC J, *Monte Carlo sampling methods*, [Online], [Cited October 2020], Available from <http%20://www.nuc.berkeley.edu/All-Courses>.
- [83] WADDELL P, 2000, *A behavioral simulation model for metropolitan policy analysis and planning: Residential location and housing market components of UrbanSim*, Environment and Planning B: Planning and Design, **27(2)**, pp. 247–263.
- [84] WADDELL P, 2002, *UrbanSim: Modeling urban development for land use, transportation, and environmental planning*, Journal of the American Planning Association, **68(3)**, pp. 297–314.
- [85] WADDELL P, 2005, *Confronting the bane of endogeneity in modelling urban social dynamics*, Proceedings of the 1st Workshop on Modelling Urban Social Dynamics, University of Surrey, pp. 4–10.
- [86] WADDELL P, 2010, *Modelling Residential Location in UrbanSim*, pp. 165–180 in PAGLIARA F, PRESTON J & SIMMONDS D (EDS), *Residential location choice: Models and applications*, Springer Science & Business Media, Berlin.
- [87] WADDELL P, BORNING A, NOTH M, FREIER N, BECKE M & ULFARSSON G, 2003, *Microsimulation of urban development and location choices: Design and implementation of UrbanSim*, Networks and Spatial Economics, **3(1)**, pp. 43–67.
- [88] WADDELL P & ULFARSSON G, 2003, *Dynamic simulation of real estate development and land prices within an integrated land use and transportation model system*, Proceedings of the 82nd Annual Meeting of the Transportation Research Board, Washington (DC), **(8)**, pp. 12–16.

-
- [89] WADDELL P & ULFARSSON G, 2004, *Introduction to urban simulation: Design and development of operational models*, pp. 203–236 in HENSHER D, BUTTON K, HAYNES K & STOPHER P (EDS), *Handbook of transport geography and spatial systems*, Emerald Group Publishing Limited, Amsterdam.
- [90] WALLS P, 2019, *Newton's method*, [Online], [Cited October 2020], Available from <https://www.math.ubc.ca/~pwalls/math-python/roots-optimization/newton/>.
- [91] WEGENER M, 2004, *Overview of land-use transport models*, *Handbook of Transport Geography and Spatial Systems*, **5**, pp. 127–146.
- [92] WEINER E, 1992, *Urban transportation planning in the United States — An historical overview*, (Unpublished) Technical Report DOT-T-93-02, United States Department of Transportation, Research and Special Programs, Washington (DC).
- [93] WORDEN N, VAN HEYNINGEN E & BICKFORD-SMITH V, 2004, *Cape Town: The making of a city*, David Phillips Publishers, Cape Town.
- [94] ZULULAND OBSERVER, 2016, *History of Indian indentured labourers*, [Online], [Cited March 2019], Available from <https://zululandobserver.co.za/129031/history-of-indian-indentured-labourers/>.