Calculating the Energy Potential of Solar PV Located on Northern Cape Mining Properties Using R

by

Waldo van der Merwe

Thesis presented in fulfilment of the requirements for the degree of Master of Engineering Management in the Faculty of Engineering at Stellenbosch University

Supervisor: Prof. A.C. Brent
Co-Supervisor: Ms I.H. de Kock

April 2019

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.

Signature: ..Waldo van der Merwe..

Date: April 2019

Abstract

Calculating the Energy Potential of Solar PV Located on Northern Cape Mining Properties Using R

W. van der Merwe

Department of Industrial Engineering,
University of Stellenbosch,
Private Bag X1, Matieland 7602, South Africa.

Thesis: MEng. (Engineering Management)

April 2019

South-Africa, as a sovereign member of various international treaties and agreements, is bound to agreed-upon objectives set to limit the proliferation of global climate change. The ramifications of these objectives have the potential to be particularly severe in a country where the social-, political- and economic structures have been intertwined in what has become known as the *minerals-energy complex*. Electricity generation, as a significant representative of the energy sector, is a key sector targeted for change by policymakers. Rightly so, as coal-based electricity generation constitutes the vast majority of all generation types and this has earned the country a precariously high position as a greenhouse gas emitter compared to other countries with similar levels of gross domestic product output.

       The policy arena, past and present, is analyzed in conjunction with other research results obtained by combining the same policy and technical aspects, with the aim of revealing a growth path for the renewable, and

specifically solar photovoltaic, energy market. A Multi-criteria decision-making system is identified during the literature study as the most applied technique when used in conjunction with geospatial information systems. A customized version of such a system is applied to the relevant sourced datasets in order to quantify the solar potential on mining land which currently holds mineral rights within the borders of the Northern Cape province. In contrast with the literature study, these areas were pre-selected based on known technical and current policy requirements. The entire quantification process was completed with the use of R and publicly available data in order to promote repeatability and prove the use of R as a cost-effective alternative in geospatial analyses.

The results of the quantitively performed analysis revealed that mining land in the Northern Cape province has enough solar photovoltaic potential to, at least, satisfy the entire country's electricity consumption on an annual basis. To be able to extract this potential, recommendations are made to stakeholders with future policy amendments in mind. Currently, the model of own-consumption is still the easiest to access in the current policy climate, given profitability can be proven. However, given the restrictions in terms of timeframe imposed on mines, using rehabilitated mining land as a long-term solution is proposed as another alternative, given the envisaged policy scope can be utilized as envisaged.

Uittreksel

Berekening van die Energie Potensiaal van sonkrag FV Geleë op Noord Kaap Mynwese Eiendomme met gebruik van R

W. van der Merwe

Department Bedryfsingenieurswese,
University van Stellenbosch,
Privaatsak X1, Matieland 7602, Suid Afrika.

Tesis: MIng. (Ingenieursbestuur)

April 2019

As `n soevereine lid van verskeie internasionale verdraë en ooreenkomste, is Suid-Afrika verplig om voorafbepaalde teikens te behaal om die ongetemde vooruitgang van globale klimaatsverandering te stuit. Die gevolge van die behaling van hierdie teikens kan moontlik elders baie negatief wees, jeens die geskiedkundige verhouding tussen die sosiale-, politiese- en ekonomiese-strukture van die land in `n konsep wat bekend geraak het as die *minerale energie kompleks*. Elektrisiteit opwekking verteenwoordig `n noemenswaardige breukdeel van die energie sektor en is dus `n sleutel rolspeler wat deur wetgewers geteiken word. Aldus, omdat steenkool gebasseerde elektrisiteit opwekking die oorgrootte meerderheid van alle opwekking bemaak en die land gevolglik `n onkarakteristieke hoë posisie beklee as `n uitsetter van groenhuis gasse vergeleke met lande met soortgelyke vlakke van bruto nasionale produk.

Die hede en verlede van die staatsbeleid arena word tesame met ander navorsing resultate vergelyk, wat verkry is deur dieselfde tegniese and beleids aspekte te kombineer, met die doelwit om `n groei pad te ontdek vir die ontwikkeling van die hernubare energie mark, met die fokus op foto-voltaise sonkrag. Tydens die naslaan van die literatuur was multi-kriterië besluitneming stelsels identifiseer as die mees toepaslike metode wanneer dit gebruik word in samewerking met geo-ruimtelike inligting stelsels. Die verkrygde data is deur `n aangepasde weergawe van so `n besluitneming stelsel gevoer om die sonkrag potensiaal van mynwese grond binne die grense van die Noordkaap provinsie te bepaal. In teenstelling met die metodes vanuit die naslaan van die literatuur, was hierdie areas vooraf gekies op grond van reeds bekende tegniese en beleids vereistes. Hierdie proses was in geheel uitegvoer met die gebruik van R en publieke data om herhaalbaarheid te promofeer en die gebruik van R vir geo analises as `n koste effektiewe alternatief te bewys.

Die resultate van die kwatitatiewe analise het onthul dat die mynwese grond in the Noordkaap provinsie beskik oor genoegsame son fotovoltaise potensiaal om op minimum, die elektrisiteit verbruik van die hele land te voorsien op `n jaarlikse basis. Te midde hierdie potensiaal te ontgin, word voorstelle gemaak aan alle belanghebbendes met die blik op toekomstige energie beleid uitbreidings. Tans is die model van eie-verbruik steeds die maklikste om te ontgin in die huidige beleidsklimaat, op voorwaarde dat dit as winsgewend getoon kan word. Nesdieteenstaande, gegewe dat tydlyn beperkinge op mynwese lisensies afgedwing word, is die meer stabiele lang termyn opsie om gebruik te maak van gerehabiliteerde mynwese grond, sou die beoogde beleidsraamwerk benut kon word.

Acknowledgements

I would like to express my sincere gratitude to the following people and organizations:

- Professor Alan Brent, my supervisor, for his guidance and allowing me the freedom to carve out my own path.
- All the staff at the Chief Directorate: National Geo-spatial Information, who have always been courteous and utterly professional.
- To my dear friends, Doctor Jacobus Müller and Doctor Stefanie Malan-Müller, who's example was a living inspiration for me to start this thesis.
- To my loving wife, Elizabeth van der Merwe, your support through the last four years has been an indispensable ingredient to finishing this thesis and any success is as much yours as it is mine.

The Author, December 2018

Contents

List of figures

List of tables

Acronyms

ANN	Artificial Neural Network
BRICS	Brazil Russia India China South-Africa
CRS	Coordinate Reference System
CSIR	Council for Scientific and Industrial Research
CSP	Concentrated Solar Power
CSV	Comma-separated values
DEM	Digital Elevation Model
DLR	German Aerospace Center
DME	Department of Minerals and Energy
DNI	Direct Normal Irradiation
DSS	Decision Support System
EIUG	Energy Intensive User Group
ESRI	Environmental Systems Research Institute
FBC	Fluidised Bed Combustion
GB	Gigabyte
GDP	Gross Domestic Product
GHI	Global Horizontal Irradiation
GW	Gigawatt
GWh	Gigawatt hour
IDW	Inverse Distance Weighted
IEP	Integrated Energy Plan
IPP	Independent Power Producer
IRP	Integrated Resource Plan
ISMO	Independent Systems and Market Operator
kB	Kilobyte
kWh	Kilowatt hour

LHS	Lef-hand side
LiDAR	Light Detection and Ranging
LTI	Latitude Tilt Irradiation
MB	Megabyte
MCA	Multi-Criteria Analysis
MCDA	Multi-Criteria Decision Analysis
MCDM	Multi-Criteria Decision Making
MCE	Multi-Criteria Evaluation
MEC	Minerals-Energy Complex
MW	Megawatt
NASA	National Aeronautics and Space Administration
NERSA	National Energy Regulator of South Africa
NREL	National Renewable Energy Laboratory
OECD	Organisation for Economic Co-operation and Development
PPA	Power Purchase Agreement
PV	Photovoltaic
RE	Renewable Energy
REFIT	Renewable Energy Feed-In Tariff
RED	Regional Energy Distributor
REIPP	Renewable Energy Independent Power Producer
RFI	Request For Information
RGB	Red Green Blue
RHS	Right-hand side
RSA	Republic of South Africa
SA	South Africa
SAWS	South African Weather Services
SWERA	Solar and Wind Energy Resource Assessment
TB	Terabyte
TWh	Terawatt hour
USDE	United States Department of Energy

Chapter 1

1.   Introduction and Background

1.1 The Minerals-Energy Complex

In 1882 Thomas Edison pioneered the first central power station, the Pearl Street Station, in New York. In the very same year, electrically powered street lamps were rolled out in Kimberley, South Africa (Gratwick & Eberhard 2008). These two seemingly unrelated events perfectly capture the essence of the so-called Minerals-Energy Complex (MEC) as first coined by Ben Fine and Zavarah Rustomjee (Fine & Rustomjee 1996) in their 1996 book on the topic of the deeper entrenched socio-political background to the South African economy. The term describes the national activities organized in and around the energy and mining sectors and associated sub-sectors of manufacturing (Baker 2011; Krupa & Burch 2011). The result is that the public/private divide is intrinsically linked at the state and private capital level due to a core set of activities around mining and energy (Fine & Rustomjee 1996). Following this reasoning, it is no accident that a small mining town far away from western civilization was one of the first in the world to install electrically powered street lights. Mining for precious stones, in this case, diamonds, was the catalyst to the economy centred around the energy-intensive business of extracting minerals. And street lights, an energy utilizing luxury, were the unintended consequence.

Since then this has only been cemented. Mining is the cornerstone of the economy and at the turn of the century accounted for 18% of the South African GDP and 60% of exports annually (Boyse et al. 2014), but being as low as 8% of GDP and accounting for 30% of total exports (Votteler 2016) in more recent times. This trend is also visible in the country's falling position as a coal exporter, sitting as high as the world's fifth largest coal producer (Baker 2011) and gently slipping down to the sixth largest producer (Krupa & Burch 2011) and even lower in recent times. While the mining glory days of the 1970's are long gone, it is still a very significant portion of GPD and it can be argued that the downward trend has been exaggerated in GDP terms because the economy has evolved over time to be two-tiered. The first, as expected, includes the primary sectors such as agriculture, manufacturing, and mining, while the second tier consists largely of a sophisticated and internationally competitive financial and services market

(Nakumuryango & Inglesi-Lotz 2016). This movement away from an industrial economy, where GDP and energy consumption are intrinsically linked, to a services-oriented economy, also leads to the decoupling of GDP from energy consumption where each energy unit required per GDP output unit is less (Fischer-Kowalski et al. 2011). If not for this, the economic consequences could possibly have been even more dire when rolling electricity blackouts hit the country in 2008 after demand started to outstrip supply (Krupa & Burch 2011). Loss of GDP due to electricity shortage is estimated to have been $2.2 billion in the year 2008 alone and forced the permanent closure of several key mines (Boyse et al. 2014).

The relationship between electricity production and GDP is, therefore, something that needs to be managed carefully, especially since both of these are slow moving and require a big inertia change in order to effect long-term change. Coal is a key element in this relationship. The country is not only a big exporter of coal but also a big user of coal to power its economy. It can be seen as a positive feedback loop where the mining of coal (and other minerals) drives the economy and this, in turn, creates a hunger for electricity which is fed by mining more coal. The most recent number from the World Bank in 2014 indicated that South Africa produces 93% of all electricity from coal sources (World Bank 2018). In fact, coal-based electricity production supplies 29% of South Africa's energy demand (total energy demand including transportation fuels), making it the largest energy sub-sector in the country, responsible for 50% of all local carbon emissions (Baker 2011).

This spells trouble for the future where carbon emissions are seen with increasing hostility and South Africa is one of a few number of countries that emit a disproportionately high level of greenhouse gasses compared to GDP output due to its reliance on coal-based electricity and the nature of the economy in the context of the MEC (Krupa & Burch 2011). The government is acutely aware of this and the country has pledged itself to reduce greenhouse gas emissions by 34% by 2020 and by 42% by 2025 at the 2009 Copenhagen climate change summit (Baker 2011; Oxford Analytica 2013). This is quite a mammoth target for a extraction based economy, given that South Africa was solely responsible for 90,6% of all the CO_2 emitted on the African continent during 2002 and together with its BRICS partners, are predicted to account for the highest contribution to the world average increasing by 37% until 2030 (Votteler 2016). The share of these greenhouse gas emissions that can be coupled to electricity generation, is predicted to rise from 237 million tons of CO_2 in 2010 to 272 million tons in 2030 (Baker 2011). The concept of a carbon tax, while still priced very low, was

implemented as one avenue of achieving the country's promised carbon reduction goal with the aim of reaching a CO_2 emissions plateau by 2025 to appease international governments enough in order to avoid possible trade restrictions (Votteler 2016). Although not directly related, the White Paper on Renewable Energy from as far back as 2003 estimated that investment in renewable energy could translate in a cost saving of R62 billion and 294 Megatonnes of carbon dioxide (Department of Minerals and Energy 2003).

1.2 Electricity Policy

The large-scale generation, transmission and distribution of electricity in South Africa has always been the domain of Eskom alone. The recent events around private generation and the establishment of an Independent System and Market Operator (ISMO) which is repeatedly discussed in parliament, but showing frustratingly little progress (Boyse et al. 2014) is only the latter half of a story that has its beginning firmly rooted much further back in history. The entire policy sphere required an overhaul and the first visible step in this direction came in the very constitution, freshly drafted in 1996 with this statement: *"Government must establish a national energy policy to ensure that national energy resources are adequately tapped and delivered to cater for the needs of the nation. Energy should be made available and affordable to all citizens, irrespective of geographic location. The production and distribution of energy should be sustainable and lead to an improvement in the standard of living of citizens."* (Republic of South Africa 1996) At that time, the Department of Mineral and Energy, as it was known then, came from a background where supply security was the most important consideration, now going to one where the large-scale rollout of electrification as promised through the constitution became the biggest focus. In all likelihood, this will remain the focus in future energy planning, as apportioning each citizen the right to fulfil his or her basic needs and to live a dignified life is enshrined within the highest law of the land. And it was arguably this noble cause that transformed the country from one where only 30% of households had access to electricity in 1994, to 70% electrification in 2010 (Reed et al. 2003; Edkins 2012).

This success has to be attributed to Eskom, in the absence of any competition, and surely contributed to the company's mammoth size to this day, standing at 95% of all electricity generated in South Africa with an installed capacity of around 47 GW, 39 GW of which being coal-powered plants (Eskom 2017). From these numbers, it seems that no attempt was

made to introduce private sector competition. This, however, is not the case and even before the amendment to the constitution, Act 46 of 1994 amended the Electricity Act of 1987 by stating that anyone who wishes to generate and sell electricity is able to do so if a relevant license was obtained (Republic of South Africa 1994). This theme was carried over into the White Paper on the Energy of 1998[1] with the statement that "*Government intends to steadily increase competitive pressures in the generation sector in order to improve efficiencies and reduce electricity prices*" (Department of Minerals and Energy 1998). This steady pressure was reiterated three years later with the promulgation of the Eskom Conversion Act (Act no 13 of 2001) in August 2001. It was this Act that converted Eskom into a public listed company, although shareholding still resides with the government. The Act also allows for power stations to be sold into the private sector (Reed et al. 2003). The private sector was not easy to convince and to seemingly set investor risk managers at ease, Cabinet ruled in 2001 that Eskom would not be allowed to build new generation plants (Baker 2011). Even Eskom tried to do the completely unnatural by trying to stimulate the creation of its own competition. The Pilot National Cogeneration Programme, the Medium-Term Power Purchase Programme, and the Multisite Base-Load Independent Power Producer Program, were conceptualized by Eskom in 2007-2008 with the primary objective of expanding generation capacity. These programmes were, however, all interrupted due to the lack of readiness from both government and the private sector (Montmasson-Clair & Ryan 2014).

Still, investors had too many unanswered questions that implied too much risk. The Electricity Regulation Act of 2006 was the first concrete move from government to address these. This Act appointed the National Energy Regulator of South Africa (NERSA) to determine the electricity tariffs, set the conditions under which electricity may be sold in the country, approve licenses for generation, distribution and transmission, and oversee the import, export and trading of electricity (Baker 2011; Republic of South Africa 1996). Distribution also underwent change, with the country divided into six regional distributors (REDs) that works through a central distribution holding company, (Reed et al. 2003) although this holding company exists internal to Eskom. Lastly, the Electricity Pricing Policy 2008 created a very complete guide on the various avenues allowed for the

---

[1] Interestingly this document predicted that Eskom's generation capacity surplus would have been depleted by 2007. A prediction that was proven surprisingly precise a decade later.

distribution of electricity from various generators to various users. These included single buyer-, wholesale-, distributor- and retail-avenues as well as the corresponding tariff structure to be used for each (Department of Minerals and Energy 2008).

Until this point, the majority of documents were Acts that sought to change the laws of the country to allow for the private generation and sale of electricity on the national grid. The Integrated Resource Plan 2010 (IRP2010) released in March 2011 differs from this in that it had no power to change laws, but its function was to align the focus of all spheres of government on energy security for the following 20 years. A closer look at the IRP2010 shows an allocation for coal-based power generation to IPPs, but it is doubtful whether this will materialize, especially in the light of the ISMO draft bill (Republic of South Africa 2011) being scrapped at the last second in parliament recently for the umpteenth time. The IRP makes another fact clear and that is that coal-fired power generation will remain the primary source of electricity for the full extent of the plan, (Department of Energy 2011) but does not leave the generation entirely in the hands of only Eskom. It requests a firm commitment from the private sector for the funding, construction and operation of coal fluidised bed combustion (FBC) power plants (Department of Energy 2011) in IRP2010 and the 2016 revised version of this document, Integrated Energy Plan 2016 (Republic of South Africa 2016) entrusts an even greater proportion to IPPs. According to IEP2016, 30% of all new build requirements in coal, gas and solar CSP and 100% of Solar PV, Wind and biomass will be allocated to IPPs.

The move towards IPPs is not singularly motivated. The Eskom generation fleet is fast ageing and to replace them would require raising capital to the amount of ZAR 337 billion. This comes from various loans, with the government providing the guarantees to the lenders up to ZAR 350 billion and it is becoming a very large liability for government indeed. The only way to relieve the government of this burden is to raise capital through IPPs and adjusting the price of electricity to be more cost reflective than what it was in the past (Van den Berg 2013). But Eskom still poses another threat to the long-term success of IPPs because it owns all of the transmission equipment and half of the distribution network, with the other half of the distribution network owned by individual municipalities (Krupa & Burch 2011). The key to unlocking this lies in the ISMO bill, which is meant to create a state entity independent of electricity generators (including Eskom) and distributors, and serve as a buyer of electricity from generators and seller of power to customers at wholesale level, but it remains stalled in Parliament as of September 2014 (Montmasson-Clair & Ryan

2014). This is the last remaining piece of the puzzle and without it, nearly all the legislative and policy changes described in the chapter would amount to very little except for a very successful, but limited renewables program.

1.3   Renewable Electricity Policy

The South African government's privately funded REIPP Procurement Program is very well regarded in the world and classified among the top ten renewable energy programs (Nakumuryango & Inglesi-Lotz 2016). Its success is in the progress of being replicated as it is being exported to 11 other African countries (PR Newswire Europe Including UK Disclose 2016). Evidence of its success is visible in how it has brought online 2902 MW of renewable energy generation capacity from 56 projects through R201.8 billion of investment (Nakumuryango & Inglesi-Lotz 2016). But it is important to note that looking at this program in isolation would be erroneous. None of this lauded success would have been possible if the legal frameworks were not set in place by all the policy changes, as described in the previous section.

Evidence of this can be found in all the policy documents dealt with earlier. The first of these was the White Paper on Energy in 1998 directly following the constitution change in 1996. Amongst other things it stated that "*further development of renewable and environmentally benign generation technologies such as hydro, wind, solar thermal, and waste incineration will also be encouraged,*" but it is clear from the tone of the document that renewable energy was mostly seen as biomass being used as feedstock for cooking fuel (Department of Minerals and Energy 1998). This tone changed dramatically when renewable energy was the main focus of an entire bill on its own with the arrival of the White Paper on Renewable Energy in 2003. This document was the first to hint at renewable energy at the utility scale (Department of Minerals and Energy 2003), but another eight years had to go by before the release of the IRP2010 (Department of Energy 2011). This was the first time that any policy document went beyond rhetoric insofar as setting a measurable target of 17.8 GW of energy to be sourced from renewable sources by 2030 (Edkins 2012). Before this, it was only the White Paper on Renewable Energy which set an arbitrary goal of producing 10 GWh of energy from renewable sources (Department of Minerals and Energy 2003), which also brings up the question of why a target was set up in terms of Watt-hours.

This looming target seemed to instil some urgency and a Renewable Energy Feed-In Tariff (REFIT) system was hastily researched after a Request For Information (RFI) was sent out before the drafting of the IRP2010. This put NERSA in a precarious position, as they are responsible for the review and regulation of electricity tariffs (Republic of South Africa 2006) and by the time the IRP2010 was released one year late in 2011, they were already one year behind schedule. Acting to rectify this, they proceeded with REFIT, only to be overruled by the Department of Energy who surprised everyone by scrapping REFIT in favour of a competitive bidding system by the end of 2011. This created some conflict between the two parties, but eventually, it was found that NERSA was acting beyond its mandate stipulated in the Electricity Regulation Act No. 4 of 2006 (Montmasson-Clair & Ryan 2014).

The success of the REIPPP Procurement Program, was, therefore, the result of many years of events that led to a point where one arm of government, the Department of Energy, had the political will to be the champion that renewable energy and IPPs required to succeed. Another cornerstone of this success can be attributed to the Power Purchase Agreement (PPA). The PPA is probably the most important document that an IPP can possess, as it is the only source of revenue for developers and for commercial banks financing IPPs and is indispensable to the success of any IPP programme (Montmasson-Clair & Ryan 2014). In all African countries where the standard model has been implemented in some degree, it was in the PPA that details who will buy the power, details about power capacity, specific energy charges, fuel metering, termination, interconnection, financing arrangement, force majeure and dispute resolution (Eberhard & Gratwick 2011). But beyond the sphere of the REIPPP, however, there is again no certainty about whether it is even possible to negotiate a PPA with the government.

1.3.1   Solar Energy as a Natural Resource

In the context of the MEC and South Africa's position in the world as the sixth largest coal mining industry, the realisation of a strong solar industry would seem to be expected, given that the solar resources in the country are of the highest in the world. While having some of the world's biggest coal reserves, this pales in comparison to its position in the world rankings on solar resource, as together with Chile, South Africa has some of the highest solar insolation numbers in the world (C. Parrado; A. Girard; F. Simon; E. Fuentealba 2016; Nakumuryango & Inglesi-Lotz 2016) with

enough of these and other renewable resources to satisfy a large percentage of all current GDP activities (van Niekerk 2014; Edkins 2012).

The failure to extract this resource is even more evident when comparing the local solar industry to that of Germany. In 2011 Germany was the leading installer of solar PV installations in Europe, totalling roughly 24.8 GW through a million installations, enough to represent 37% of total global cumulative installations at that time (Fraunhofer Institute 2012). In 2016 this figure moved closer to 40 GW or almost the equivalent of Eskom's entire generation capacity (Council for Scientific and Industrial Research (CSIR) 2016). Figure 1.2 shows a map of the Global Horizontal Irradiance found in that country. No specific value is of concern here, but more tellingly comparing the scale at the bottom of this map with the scale at the bottom of the equivalent GHI map of South Africa (Figure 1.3) reveals that the top end of this scale is lower than the lowest end of the scale on the South African map. This clearly illustrates the abundance of the resource that the country is endowed with.
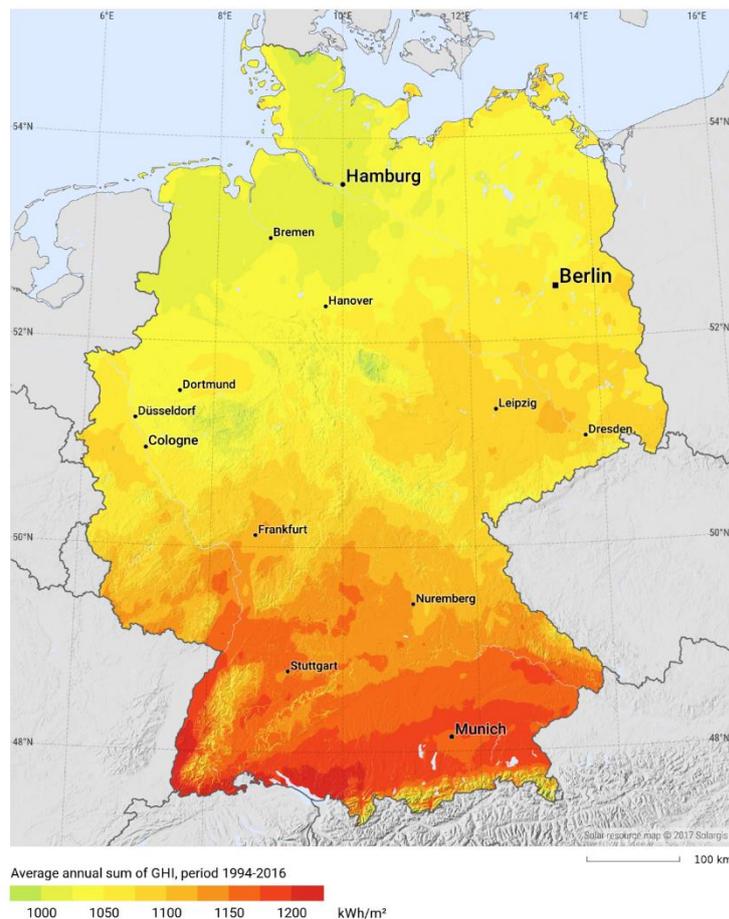
Figure 1.1: Global Horizontal Irradiance in Germany. Source: (SolarGIS 2018)
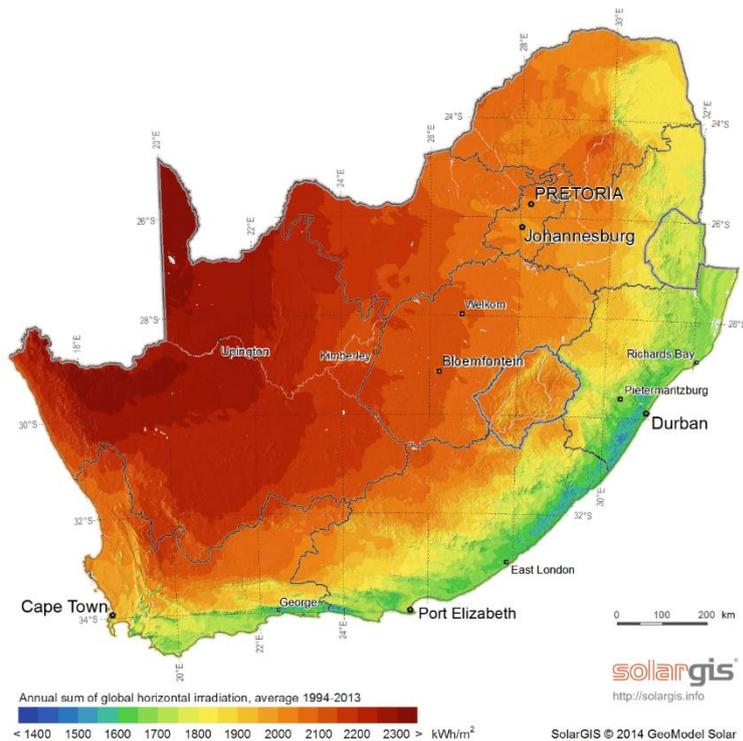
Figure 1.2: Global Horizontal Irradiance in South Africa. Source: (Stellenbosch University et al. 2014)

Despite this abundant resource and the success of the REIPP procurement program, there is a part of the program which does not attract headlines. Since inception, the program has had an allocation for small projects with capacities ranging from 1-5 MW. This has been included as part of the allocation for, amongst others, landfill gas, small hydro, biomass and biogas. In the IPP overview report (Independent Power Producer Office 2017a) released in June 2017, there was an allocation of 940 MW for this category, with 400 MW of that allocated to small-scale projects, including small-scale solar, with 99 MW of this being allocated. Of this 99 MW the bulk, 80 MW, was small-scale solar and more than half of these solar projects, 55 MW, was in the Northern Cape according to the provincial report (Independent Power Producer Office 2017b). However, this progress has been very recent, as the same report of 2015 (Independent Power Producer Office 2015) reported a zero uptake of this allocation while the DOEs report of the same year (Department of Energy 2015) pointed out this slow progress being made on this front, pointing the blame to a disinterest by local banks to provide funding.

1.3.2 The Northern Cape Province

When looking at Figure 1.3, the areas of highest GHI almost completely follows the borders of the Northern Cape province of South Africa. Seeing that the solar energy potential of the country is not nearly fully exploited, it would make sense to focus on the Northern Cape as it makes sense to pluck the low hanging fruit first. It is for this reason that this study focusses on the Northern Cape exclusively. This is by no means a unique perspective, as visible in Figure 1.4, developers of utility-scale renewable energy sources have overwhelmingly settled here first, especially with solar technologies. The total for the province ranges in the 3 - 4 GW, which is significant, seeing as the total consumption in the province in 2013 was a mere 725 MW (Urban-Econ 2013), which means it has become a net exporter of electricity.
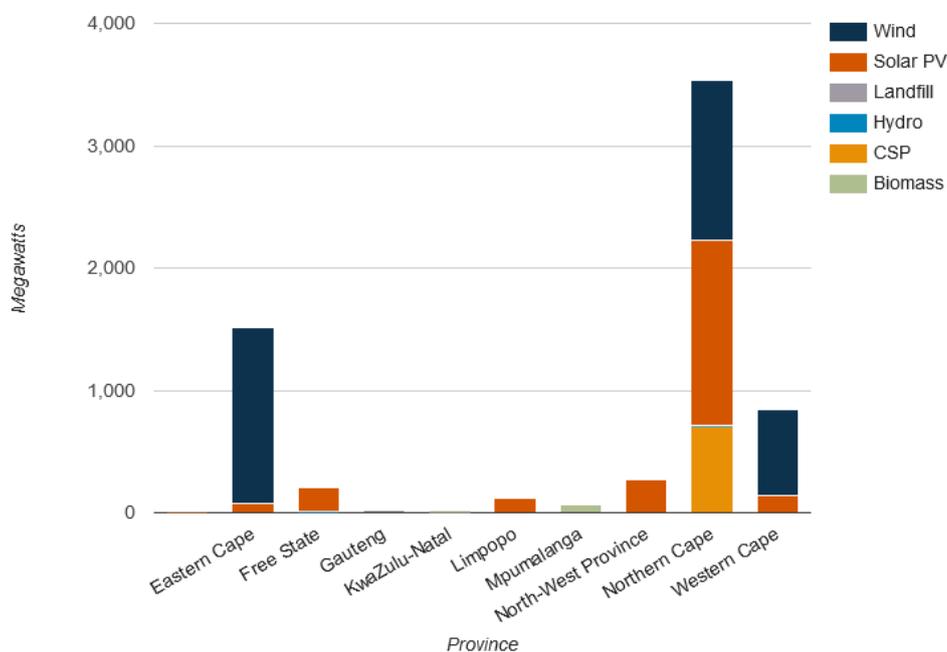
Figure 1.3: South Africa utility scale RE technologies by province. Source: (Energy Blog 2018)

The REIPP Procurement Program is largely driven by the national government through the Department of Energy, while environmental approvals are under the mandate of National Department of Environmental Affairs (Urban-Econ 2013) implying that the local government authorities have little scope in controlling the rollout and beneficiation of utility-scale projects. Municipalities also cannot sign contracts that last longer than three years due to limitations in the Municipal Management Finance Act. If local government, therefore, wants to be in control of developing the RE sector

as a means of stimulating economic activity, then a good place to look might be with smaller projects in conjunction with private sector players on private land administrated by the municipalities.

1.4 Minerals Extraction Policy

1.4.1   Electricity as an Input Resource

Mines are large electricity consumers. During 2011 the mining sector was responsible for 15% of South Africa's entire electricity demand (Boyse et al. 2014). The number becomes even bigger when considering downstream activities. The Energy Intensive Users Group of Southern Africa (EIUG) represents the top electricity consumers in the country and, while not exclusively, their membership consists of a large number of mines. The remaining members are mostly involved in the downstream beneficiation of mining activities and in total, the EIUG consumes 40% of South Africa's total electricity production (EIUG 2018; Baker 2011). This happens on a nearly 24-hour scale in most of the cases and therefore mines require a
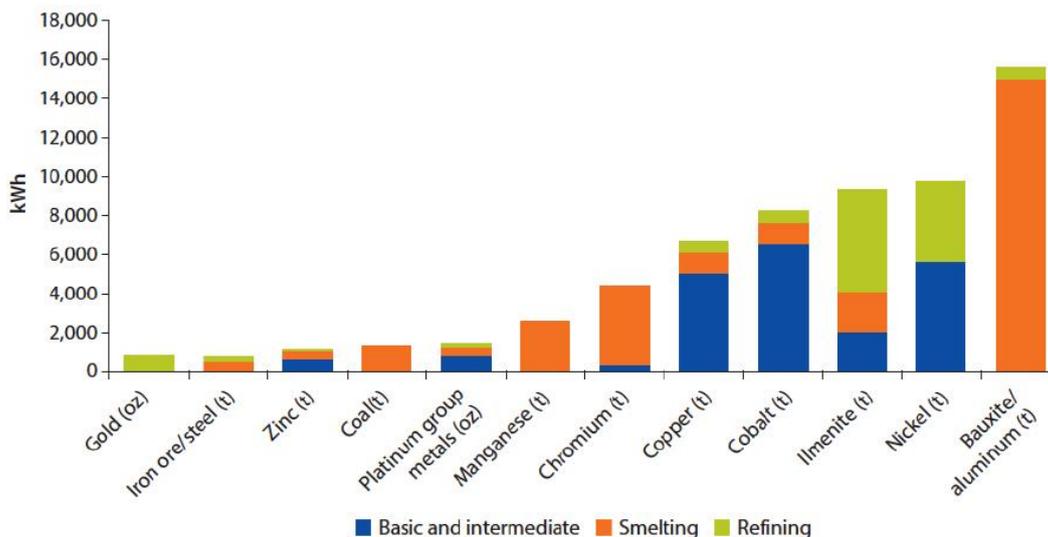
Figure 1.4: Cumulative power requirements at stages of beneficiation. Source: (Votteler 2016)

constantly available supply of electricity (Votteler & Brent 2017a). The electricity supply profile is also varying dependant on the stage of beneficiation, as shown in Figure 1.5, but this graph also reveals another characteristic of Northern Cape mines. Table 1.1 is a summary of information obtained from the Department of Minerals. Most of the mines in the Northern Cape are open cast mines. Underground mining operations require significantly higher quantities of electricity than surface mining, due to a great rise in hauling requirements, ventilation, water pumping and other

operations (Votteler 2016). Owing to this and that the commodities listed in Table 1.1 are not present on the right-hand side of Figure 1.5, it is safe to say that the majority of the mines are on the lower end of the electricity usage spectrum. Nevertheless, electricity is a large portion of the day to day expenses of running a mine. In South Africa, information on these costs is hard to come by, as many of the larger players such as BHP Billiton, Anglo American and Xstrata, negotiated secretive special electricity purchasing agreements with Eskom that have been in place since the apartheid era (Baker 2011). For comparison, energy costs for mines in Chili amount to 20% - 40% of operational costs (Walker 2015).

Table 1.1 Total number of each type of mine. Source: (Department of Mineral Resources 2015)

Commodity	Number of licensed mines	Type of majority
Aggregate, Sand	17	Opencast
Copper	1	Underground
Diamonds: Alluvial, Marine, Kimberlite	167	Opencast (except for Kimberlite)
Granite	6	Opencast
Sandstone	1	Opencast
Feldspar & Gypsum	3	Opencast
Iron & Manganese Ore	27	Opencast
Salt	13	Opencast
Semi-precious stones	19	Opencast
Shale brickmaking	2	Opencast
Kieselguhr	1	Opencast
Limestone	3	Opencast
Zinc, Lead, Silver	1	Underground

It is hard to see how these costs will change in any way other than by increasing dramatically. Figure 1.6 shows Eskom's own supply prediction increasing well into the future, which is a bold prediction in the context of an economic sector which has shown a long-term decline. Eskom has also been testing the limits of their monopoly by increasing the price of electricity by 26% from 2007 to 2012 (Boyse et al. 2014; Krupa & Burch 2011). This is not the only form of energy utilised by mines and when looking at the combined price of both, total operational expenditure on energy has increased from 7% to 20% in the seven-year period from 2008 to 2014 for

the 47% of companies in the mining sector who are part of the EIUG (Votteler 2016). Diesel alone increased by 15.7% during the period from 2007 to 2012 (Boyse et al. 2014).
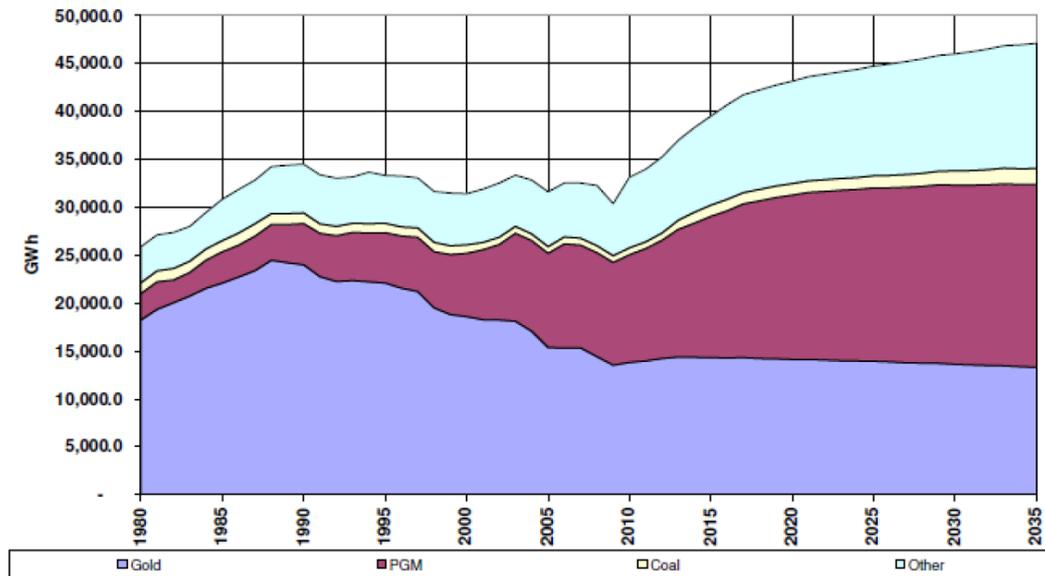
Figure 1.5 Eskom historical and predicted sales to mining sector. Source: (Baker 2011)

None of the above accounts for the current and future price of carbon. In 2009 a 2c/kWh levy was introduced on electricity generated from non-renewable sources (Baker 2011), which can hardly account for the overall increases. The government has been vying to implement a carbon tax since 2010 (Boyse et al. 2014), which didn't materialise at the time, but a new draft carbon tax bill has been published in December 2017 (Minister of Finance 2017). If implemented as envisioned from January 2019, the price of diesel may increase from a minimum of 11.4 cents per litre to 28.6 cents per litre (BusinessTech 2018) while the immediate price of electricity might stay unaffected until the tax-free thresholds expire. If the carbon tax would truly account for the life-cycle burdens and damages of coal-derived electricity conservatively it would double to quadruple the price of electricity, making renewable energy sources such as wind and solar attractive alternatives (Nkambule & Blignaut 2017). Combine the fact that Eskom has become the cornerstone of state capture (Nakumuryango & Inglesi-Lotz 2016) with the tariff increases and insecurity of supply by Eskom and it is understandable that more and more mines in South Africa are moving towards greater independence and self-supply (Votteler 2016).

1.4.2　　Conditions of Granting a License

A number of legal requirements from government imposed on mining companies are discussed below. The aim of this section is to highlight current legislation imposed on mining companies which might be repurposed for energy generation with no or little adaptation. These will be tied to previous research within this domain in the next section. All the requirements discussed below are contained within two documents: The *Minerals and Petroleum Resources Development Act* (Republic of South Africa 2002) and *A Guideline for a Mining Work Programme to be Submitted for an Application for a Mining Right in Terms of the Minerals and Petroleum Resources Development Act* (Department of Mineral Resources 2002). While this is not an exhaustive list of overlapping requirements compared to the REIPP procurement program, it aims to show the overlap that exists from a policy perspective. This opens the possibility of inter-departmental cooperation to remove some barriers to entry.

The government Acts regulating the mining industry has been well developed to cover all the permutations the industry can create. The result is that there are many stages of application, each with unique terminology, but also many overlapping terminologies. The first to note is the difference between a prospecting-, exploration-, mineral/mining-right and a mining license. In its simplest form, a holder of an exploration-right cannot perform any commercial scale mining activities the land area relating to the right, but a holder of a mining permit cannot have obtained the permit if it did not first apply for a prospecting- and then a mining-right. Each category of right is therefore bound in relation to each other and differentiated with respect to validity period and allowed activities. These differences are the basis on which each right might be more suitable for one specific model of energy generation than another.

REIPP procurement program overlap. Every entity applying for a mining right must conduct an environmental impact assessment and submit an environmental management programme which must include the socio-economic conditions of any person who might be directly affected by the prospecting or mining operation. This represents a direct overlap with the requirements mandated on bidders of the REIPP procurement program. Although it is not known how much the detail of each overlaps, should the mining sector requirements supersede these, it represents an opportunity to integrate these applications. Another opportunity exists at the time of closure of a mine. An applicant for a mining right or permit must make the prescribed financial provision for the rehabilitation or management of negative environmental impacts. They must as far as it is reasonably

practicable, rehabilitate the environment affected by the prospecting or mining operations to its natural state or pre-determined state or to a land use which conforms to the generally accepted principle of sustainable development. The term *pre-determined state* is possibly a phrasing in the Act which might allow for the development of a solar plant on top of the rehabilitated ground with a lifespan which far exceeds that of the mine or of any associated right or permit. This would directly address one of the main concerns raised by Boyse et al. (2013) in his research on the topic. Should a mine not consider this as part of their expertise, the responsibility may be transferred to another entity, such as a solar developer. In respect of this, the Act states that: On written application by the holder of a prospecting right, mining right or mining permit, the minister may transfer such environmental liabilities and responsibilities as may be identified in the environmental management plan or the environmental management programme and any prescribed closure plan to a person with such qualifications as may be prescribed. In this instance, a model with shared ownership through combined capital might be one model which exploits these policies to the benefit of all parties. If not a solar developer, then the owner of the land itself (if not the mining company) may be the beneficiary, which could be incorporated within the pre-determined social development plan, because the Act provides that all structures built for the purpose of mineral extraction and its associated activities, must remain undemolished if the terms of agreement between the rights holder and the land owner requires it and it was approved by the Minister in writing.

Own consumption. The simplest model would be for mines to be allowed to generate their own power to be fully utilised through their own consumption needs. While not mentioning solar plants explicitly, the Act does state that the rights of a prospecting-, exploration- or mining-right holder include being allowed to bring onto the land any plant, machinery or equipment and build, construct or lay down any surface which may be required for the purpose of mining or production. This right is extended to include carrying out any other activity incidental to mining or production operations which does not contravene any other provisions found in the Act. This would seemingly cater to the construction of a solar plant. Still, a big concern is the lifespan of the mine and whether it is long enough to cover the payback period of the solar plant. According to the Act, a mining right may be renewed any number of periods which does not exceed 30 years. This is a far longer period than that of a mining permit, which is only valid for two years and may only be renewed for three periods of one year each. These periods may or may not include periods in between where a retention permit is granted. A

retention permit is essentially an application to be allowed to freeze mining activities due to market-related concerns, should a mining company be able to prove that prevailing market conditions do not allow for the profitable extraction of minerals. This permit can be issued for a maximum period of three years. While not legally black and white, the retention permit combined with the maximum allowed renewal periods of a mining permit might be sufficient to at least achieve parity on the payback period.

Government ownership. While it might be considered a fringe option, the Minister may use expropriation of land as a mechanism to establish a solar plant on the premises of the mine. The government itself, or a beneficiary such as the local community may be the owner, or a third party acting on behalf of either. Section 55 of the Act makes provision for the expropriation of property in order to achieve the objectives of section 2 of the Act. These include:

- To promote equitable access to the nation's mineral resources.
- To expand opportunities for historically disadvantaged persons.
- To promote economic growth.
- To promote employment and advance the social and economic welfare of all South Africans.
- To provide for the security of tenure in respect of prospecting, exploration, mining and production activities.

The last point, in particular, would allow for mining companies and government to work together in such a manner which flouts the maximum validity period of all the various rights, with ownership being transferred back and forth between the mine and the government as a sort of artificial retention permit.

1.5 Research Objective

The relationship between mining operations in South Africa, which is an energy intensive business, and renewable energy production is not studied in great detail. This is surprising given the obvious relationship between energy production and consumption. Although, it is also not surprising given the traditional role of Eskom played in the MEC and it is not known what research mines are performing which is not in the public domain. A report by Boyse et al. (2014) explored this possible relationship, but mostly focussed on off-grid options. This was built upon by two research papers (Votteler & Brent 2017b; Votteler & Brent 2017a) which explored this

relationship further by conducting interviews across the mining industry and developing a multi-criteria decision assessment to establish the best fit.

Relevant highlights of the report by Boyse et al. (2014) are:

- One of the potential benefits (a few is listed) of distributed renewable energy for mining and large industrials in South Africa is opportunities to repurpose land used by the mining community.
- There is a strong business case for solar-diesel hybrids to power off-grid mines via the Self-Generating model.
- Distributed renewable energy projects allow project developers to work directly with private sector actors to install renewables on private land – skiting many aspects of the REIPPP process and its associated costs.
- Financing distributed renewable energy projects continues to prove challenging due to the small size of the projects, the risky nature of the investment, which is both caused by volatility in commodities prices and the uncertain lifespan of mines and the risk profile of investing in many of the regions where off-grid mines operate.
- The self-generating model represents a promising opportunity for mining operators in South Africa and beyond, as well as for other heavy industries.
- The net-metering or Self Generation plus Powering townships models would only require regulatory changes in order to be feasible.

Relevant highlights of the research by Votteler & Brent (2017b; 2017a) are:

- Based on the current legislative and regulatory framework in South Africa, the business model of self-generation in the form of own investment or a power purchase agreement, has the greatest potential.
- Owing to the intermittency of solar PV and the constant demand of mining operations, hybrid versions with current electricity sources were identified as the best option, specifically a hybrid of Eskom and solar PV.
- There are three main stakeholders in the selected model of self-generation and own investment: the mining corporation, the project developer and the regulatory and legislative body.
- Mining corporations are peculiarly more profit-oriented in comparison to national electricity providers.

There are a few common themes that occur between these research outcomes. Firstly, solar PV seems the most likely candidate to succeed in the mining environment due to having one of the lowest prices per installed power unit and it has one of the most predictable and steady power supply curves of the renewables stable, which is a good fit with the almost constant demand of mines. Second, while there are useable avenues of wheeling power to locations other than the mine where it is generated, small regulatory hurdles dictate that self-generation (implying self-consumption) is the most attractive option. Lastly, due to the nature of mining and mineral rights legally being under the custodianship of the state (Republic of South Africa 2002) and energy production not being a specialist skill being employed by mining companies, the opportunities for energy production in this sector will always, at the very least, be a tripartite project between mining companies, solar energy developers and the government.

The previously referenced studies set the stage by establishing the compatibility between mines and solar PV on a commercial and policy level. The main objective of this paper is to use data in the most constructive possible way to inform all the stakeholders with the aim of pushing the mining and solar PV agenda forward. In the context of typical solar audits, which will be discussed exhaustively in the next two chapters, the first question is the one of *where*? But, given that the answer to this is already mining property, the main objective of this paper moves past this to the next question, which is the question of *how much*? This refers to the amount of usable solar insolation that can be converted into electrical energy. In essence, the objective is to calculate the energy potential of solar PV on mining properties located in the Northern Cape province.
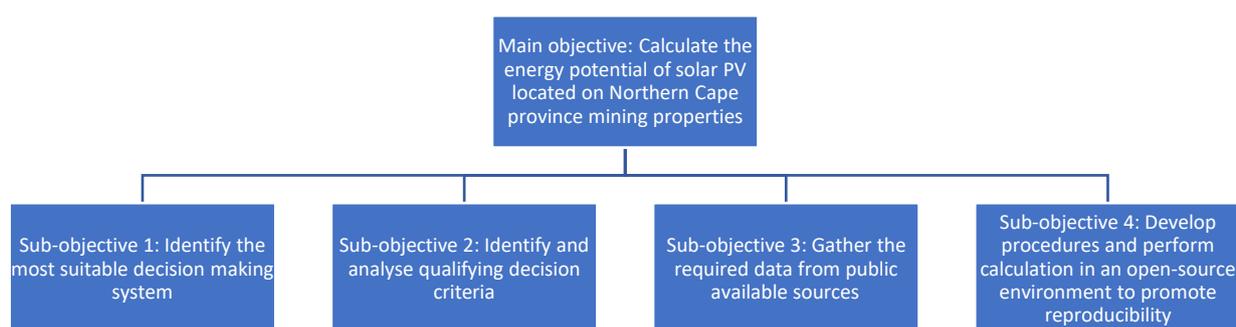
Figure 1.6: Research objectives

1.6 Research Strategy

The nature of science is to evolve human knowledge. One of the safeguards of the quality of performed research is reproducibility. While some believe that the already limited funds available to perform research are better applied exploring new ideas (Science as fact 2018), there is a corner of the scientific community concerned about this lack (National Academy of Sciences 2017). As a consequence, this research was attempted using only freely available public domain data and computed using open-source software. This would allow for a framework for reproducibility with only human capital as cost.

The conceptual division of the research objective into discernible sub-objectives allows for the research strategy to be broken down into sections, shown in Table 1.2 below. According to Petticrew & Roberts (2006), there are six approaches to synthesizing research. This paper makes use of a combination of the realistic synthesis review and rapid review methods because the focus is on the methods used in order to generalise the applied theory rather than the outcomes of each. However, as many studies were included as possible in order to ensure that regional differences might reveal nuances which might be applicable to the Northern Cape region. This brings the literature study close to a systematic review.

Table 1.2: Summary of research strategy

Sub-objective	Approach	Process	Chapter
Identify suitable decision-making system	Qualitative	Literature review	2
Identify suitable solar audit method and criteria	Qualitative	Literature review	2
Data gathering	Quantitively	Exploratory fieldwork	3
System design and processing for performing calculations	Quantitively	Applied simulation	3

The literature review as described above covers the first two sub-objectives. All the studies found to be relevant to energy audits, which initially included a number of studies which included other technologies such as wind- and biomass energy, were used to identify the first sub-objective. The next step was to exclude studies of other technologies and only focus on solar PV audits or where solar PV was included as part of a multi-technology audit. This was done to determine the specific framework used, down to the fine detail of selection criteria, in order to formulate a customised framework to apply to this research. The result of this effort is given at the end of Chapter 2. Chapter 3 deals with the third and fourth

sub-objectives. Data gathering was achieved by assembling a database of all the resources used in the literature study and adding further studies to this which focus on locally relevant content. This was followed by exploratory fieldwork to obtain access to quality local data. Lastly, the majority of the remainder of Chapter 3 deals with how all of the above was applied practically to build and execute a model to input data and calculate meaningful information from it. To achieve this, additional research had to be done qualitatively outside of the literature study sphere in order to gain the required knowledge on the specialised subject of GIS systems.

Chapter 2

2. Literature Study

This chapter describes how the literature study was conducted, what was learned from it as well as what decisions were made as a result of new facts learned. In order to better organise the results, the most important compared results were tabulated and displayed here as Table 2.1 in this section. The following sections will continually refer back to this table.

In order to achieve the main objective, as set out at the end of the previous chapter, a search was conducted which indiscriminately searched for GIS-based energy audits, regardless of energy source. Besides the detail that was obtained, there were two main themes that emerged from this review of the most recent research.

Firstly, and probably more surprisingly, GIS-based studies on wind energy potential had more in common with GIS-based studies of utility-scale PV plants than what urban PV potential studies has with utility-scale PV. The urban-based PV studies, such as those by Brito et al. (2012), Jakubiec & Reinhart (2013), Lukač et al. (2013) and Sing et al. (2016) for the areas of Lisbon, Cambridge (USA), Maribor (Slovenia) and Hong Kong respectively, were based on very precise Light Detection and Ranging (LiDAR) data. This data is heavily processed, mostly with interpolation techniques, and then automatic building detection algorithms are run to establish the location of all the rooftops in the relevant cities. It is then processed further to establish the slope, or pitch, of the roofs as well as the azimuth angle of the normal component of the roof (the direction of slope). The result is combined with population density information and solar radiation datasets to determine uptake suitability. In almost all cases, these studies focus on the data crunching processes required to make sense of the LiDAR data. This stands in contrast with the wind energy audits such as those conducted by Gorsevski et al. (2013), Janke (2010) and Latinopoulos & Kechagia (2015) for Ohio, Colorado and Greece respectively, which all follow the procedure of first establishing some sort of decision-making system dependent on multiple input criteria and then adapting this system to consider the factors for the case at hand. This, as will be discussed in the next sections of this chapter, much more closely represents the general principle followed with utility-scale PV GIS analysis.

Secondly, in the most recent GIS-based PV potential studies, all the authors used some form of Multi-Criteria Decision Analysis (MCDA) with some form of weighted selection criteria, such as Analytical Hierarchy Process (AHP), Fuzzy Logic or even an Artificial Neural Network (ANN) in the case of Borgogno Mondino et al. (2014). The exception to this was by Domínguez Bravo et al. (2007) and Carrión et al. (2008), although these closely resemble Boolean logic, which can be argued is also a weighted MCDA, but with equal weight given to all criteria and all factors having only two levels. In the case of Carrión et al. (2008), this was actually one of two papers published on the same data. The second by Arán Carrión et al. (2008) basically repeated the steps of the first study but included an MCDA implementing AHP. The application of an MCDA to a GIS-based energy audit will be discussed in the next section, given the prominence, this had in all the latest literature.

Table 2.1 Results of the most important factors compared from all the studies included in the literature study

Study	Renewable Technology	Decision Framework	Study Area	Slope Limits	Excluded Areas (Only Solar PV)	Ranked Inclusion Criteria (Only Solar PV)
(Aly et al. 2017)	CSP PV	Multi-Criteria Decision Making (MCDM) with Analytical Hierarchy Process (AHP)	Tanzania	CSP: Slope > 2.1% Excluded PV: Slope > 3% Excluded	Protected Areas Land Cover Water Bodies Urban Areas	Road/Rail Proximity Mine (User) Proximity Utility Grid Proximity Solar Resource Terrain Slope
(Abed & Nagasaka 2017)	CSP PV Wind	Multi-Criteria Decision Analysis (MCDA)	Afghanistan	PV: Slope > 5% Excluded CSP: Slope > 3% Excluded	Mountains Protected Areas Lakes Sand dunes Permafrost Marshland Irrigated Land Natural Forest Urban Areas Productive Farmland Areas with vegetation of importance.	Road/Rail Proximity River proximity Urban Areas Soil type Terrain Slope
(Domínguez Bravo et al. 2007)	CSP PV Wind Hydro electric Wave Energy Biomass	N/A	Spain	Slope < 3% Included 3% < Slope <10% Only SE-SW included Slope > 10% Excluded	Special Protection Areas Sites of Community Interest Protected Natural Areas Discontinuous Urban Industrial, Commercial, Transport Mine, Dumps and Construction Artificial Vegetated Areas Pastures Agriculture Grasslands, Moorlands, Various Bush and Shrubs	
(Arán Carrión et al. 2008)	PV	Multicriteria Evaluation (MCE) with Analytical Hierarchy Process (AHP)	Spain	Slope < 30 % Divided into 10 discreet boxes Slope > 30% Excluded	Coastal Zones Livestock Trails Road Networks Rivers Sites of Community Interest Protected Nature Reserves	Land Use Distance to Urban Areas Slopes Slope Orientation Highway Access Distance to Substations Global Irradiance Diffuse Irradiance Equivalent Sun Hours Visual Impact Average Temperature
(Carrión et al. 2008)	PV	N/A	Spain	Slope > 2% Excluded	Nature Parks National Parks Sites of Community Interest Bird Sanctuaries Historic Cattle Trails Road Networks Irrigated Land Forest and Bushes	N/A
(Charabi & Gastli 2011)	PV	Multi-Criteria Analysis (MCA) with Analytical Hierarchy Process (AHP) and Ordered Weight Averaging (OWA) with fuzzy logic	Oman	Slope > 5% Excluded	Flood pathways Rivers Dams Urban Areas Sensitive Areas Sandy Soil Historic Monuments	Solar Resource Proximity to major roads
(Garni & Awasthi 2017)	PV	Multi-Criteria Decision Making (MCDM) with Analytical Hierarchy Process (AHP)	Saudi Arabia	Slope > 5% Excluded	Wildlife sanctuaries National Parks Industrial Cities Sacred Places Urban Areas Road Networks	Proximity to roads Utility Grid Proximity Distance to Urban Areas Temperature Solar Resource
(Jahangiri et al. 2016)	PV Wind	Boolean Logic	Middle-East region	N/A	Solar radiation	N/A
(Sánchez-lozano et al. 2014)	PV	Multi-criteria Decision Analysis (MCDA) with Elimination and Choice Translating Reality (ELECTRE)	Spain	Best: 0% < Slope < 30% Worst: Slope > 50%	Urban areas Protected land Road and Rail networks Water infrastructure Watercourses Military zones Cattle trails	Solar radiation Temperature Distance to power lines Distance to urban areas Proximity to transformer substations Distance to roads

Table 2.2 (Continued)

				Divided into 10 discrete blocks	Archaeological sites Paleontological sites High landscape value areas Community interest sites	Agrological Capacity Plot size
(Sánchez-lozano et al. 2013)	PV	Multi-Criteria Decision Making (MCDM) with Analytical Hierarchy Process (AHP) and Technique for Order Preference by Similarity to Ideal Solution (TOPSIS)	Spain	Best: 0% < Slope < 30% Worst: Slope > 50% Divided into 10 discrete blocks	Land use Archaeological sites Road and Rail networks Watercourses Cattle trails Military zones Cultural heritage sites Paleontological sites Mountains Bird sanctuaries Sites of community interest	Power lines Proximity to roads Proximity to substations Slope and aspect Agrological Capacity Solar radiation Temperature
(Alami et al. 2018)	PV	Multi-Criteria Decision Making (MCDM) with Analytical Hierarchy Process (AHP)	Morocco	Slope > 5% Excluded	Protected areas Forest Active Agricultural areas Dams and rivers Urban areas Roads and railways	Solar radiation Slope Distance from urban areas Proximity to road and rail networks Proximity to electric infrastructure Distance from waterways and dams
(Merrouni et al. 2016)	PV	Boolean Logic	Morocco	Slope > 5% Excluded	Natural Forests Protected areas High slopes Proximity to road and rail network Proximity to dams and rivers Proximity to power grid	
(Uyan 2013)	PV	Multicriteria Evaluation (MCE) with Analytical Hierarchy Process (AHP)	Turkey	Slope > 3% Excluded	Rivers, lakes, wetlands and dams Road network Urban areas Protected areas	Distance from residential areas Land use Distance from transmission lines Slope Distance from roads
(Borgogno Mondino et al. 2014)	PV	Boolean pre-selection followed by quantitative assessment where weights are selected through training an Artificial Neural Network (ANN)	Italy	Slope > 15% Excluded Slope < 3% Included South Slope 3% < 15% Included	Restricted Areas Protected Areas Land use Slope and Aspect	Solar radiation Temperature Rainfall
(Noorollahi et al. 2016)	PV	Multi-criteria Decision Support System MCDSS) using a Fuzzy Analytical Hierarchy Process (FAHP	Iran	Slope > 11% Excluded Divided into 5 discrete blocks	Solar Radiation Protected areas Rivers, lakes, wetlands and dams Land use Power infrastructure Urban areas High altitude High slopes	Solar radiation Temperature Proximity to power infrastructure Distance from major roads Number of cloudy days Elevation Slope Land use Average annual Dusty days Distance from residential areas Average annual Humidity
(Tahri et al. 2015)	PV	Multicriteria Evaluation (MCE) with Analytical Hierarchy Process (AHP)	Morocco	Slope > 24% Excluded Divided into 4 discrete blocks		Land use Slope Aspect Distance to road network Distance to urban areas Solar radiation Temperature
(Yushchenko et al. 2018)	CSP PV	Multi-Criteria Decision Making (MCDM) with Analytical Hierarchy Process (AHP)	West-Africa	Slope > 10% Excluded	Urban areas Rural Population density Flood risk areas Protected areas Slope Land use	Solar radiation Distance to electricity grid Distance to road network Population density Distance from rural settlements

24

2.1 Application of a GIS as a Research Tool and GIS-Based Energy Audits

While databases are extremely useful for the storage and analysis of large sets of data in general, the ability to have data points coupled to a geographic reference provides the ability to make data more human interpretable. When the subject of solar potential and plant location is investigated, GIS-based tools, often combined with decision-making systems, are the preferred method for the past two decades (Abed & Nagasaka 2017). This combination is mainly because while GIS is excellent at the management and storage of large quantities of spatially referenced data, it lacks a number of analytical tools and this is mitigated by combining it with decision-making systems (Arán Carrión et al. 2008). The next two sections summarise the findings on the use of GIS and decision-making systems in the literature study.

2.1.1   Decision Making Systems

For the sake of clarity, the various authors use the terms Multi-Criteria Decision Analysis (MCDA), Multi-Criteria Decision Making (MCDM), Multi-Criteria Analysis (MCA), Multi-Criteria Evaluation (MCE) and Decision Support System (DSS) interchangeably, but in this paper, only MCDA will be referred to.

The main idea of MCDA is to start with a complex problem consisting of multiple input criteria and then to condense it, using various rankings, into the best possible scenarios of options (Malczewski 2006). The objective, therefore, is to determine the possible outcomes when certain decisions are taken in order to help the task manager (Arán Carrión et al. 2008). This methodology is very generic and allows for the customization of such systems to solve specific problem types. When that type is an environmental problem, it becomes known as an Environmental Decision Support System (EDSS) (Rizzoli & Young 1997). This is yet another analogy for what will be referred to as MCDA in this paper. This specific type of MCDA requires large amounts of spatial information to facilitate the decision-making process as well as visualising the results. It is for this reason that GIS systems are such a good match for this customised type of MCDA. A GIS system is very well suited to implement some of the geographically defined criteria which make up part of the decision system, such as Digital Elevation Models (DEM), solar radiation maps, proximity to transmission lines and road networks as well as restricted or protected areas. Back on the MCDA side, it would be possible to give these criteria different weights,

depending on what is perceived to be the importance of each (Diakoulaki & Karangelis 2007).

These weights and how they are appropriated are the biggest difference in the studies summarised in Table 2.1, as the weights of the attributes themselves not only differ, but the way they are related to the MCDA also differ. The most common method of assigning weights is the Analytical Hierarchy Process (AHP) and is the preferred method in nine of the 17 studies and a further one study implementing a variant using fuzzy logic (Noorollahi et al. 2016) for its AHP. Two studies ventured on the exotic side, by combining the Eliminating Choice and Translating Reality (ELECTRE) method (Sánchez-lozano et al. 2014) with their MCDA and another using the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) method (Sánchez-lozano et al. 2013) to their MCDA. One of the most interesting cases was by Borgogno Mondino et al. (2014), who trained an Artificial Neural Network (ANN) what the conditions around existing solar plants in the investigated area are and letting it come up with the criteria and associated weights for the MCDA itself. This level of sophistication was not found everywhere, especially in studies where more than one type of renewable energy source was under investigation. It is assumed that because the data management aspect is much larger for these types of study, the deliberate decision was taken to simplify the MCDA. Five of the 17 (Abed & Nagasaka 2017; Jahangiri et al. 2016; Merrouni et al. 2016; Domínguez Bravo et al. 2007; Carrión et al. 2008) used boolean logic as the basis of their MCDA. The latter two of these five did not explicitly specify their MCDA as using boolean logic, but this could be inferred after a careful look at their methodology.

At their core, nearly all of the 17 studies implemented boolean logic in their process. The generic layout of the MCDA is given in Figure 2.1 below and shows how boolean logic was implemented as exclusion criteria to pre-filter the provided datasets before applying their various models of weighted criteria on the remaining dataset. The items listed in Figure 2.1 is not exhaustive and only represents examples of how this was applied.
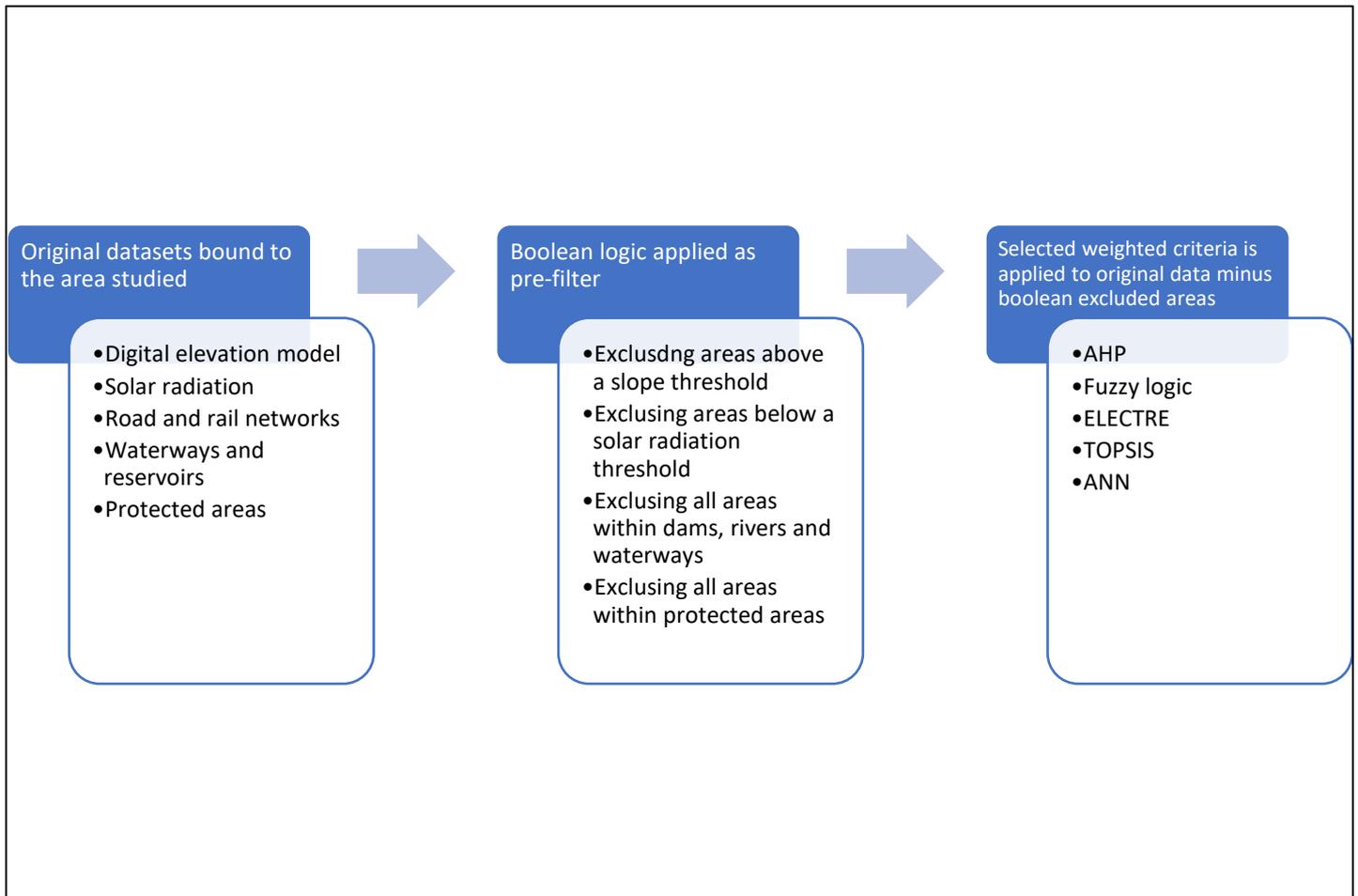
Figure 2.1: Generic process summarizing the broad methodology implemented by the studies in the literature study

2.1.2   Exclusion Criteria and Inclusion Factors

This section will discuss the most prevalent exclusion criteria and inclusion factors used in the processes summed up in Figure 2.1 of the previous section. First and foremost, it is important to understand that the elements of these two processes are not exclusive to a single process alone. In many cases, what is used as an exclusion, will later be used as an inclusion factor, albeit with different parameters. Examples of such cases will be discussed when it is relevant during each criteria discussion.

Slope and aspect. These two are discussed together as they are related closely as inclusion/exclusion factors. The slope is considered an important factor as the cost of erecting a solar plant on a large incline might be considered to be too expensive. If the slope is too great and combined with an incorrect facing aspect, it might be impossible to receive any solar radiation in that position. An example would be the southern side of a prominent hill present in the southern hemisphere. However, many of the authors in the literature study note that there is no accepted standard definition of this, as is the case for solar CSP technology, where slopes above 2.1% (or 1.2°) is almost universally accepted to be excluded (Aly et al. 2017). This leads to many variations in the literature as summarised in Table 2.1. There are two main schools of thought, with some authors (Aly et al. 2017; Abed & Nagasaka 2017; Domínguez Bravo et al. 2007; Carrión et al. 2008; Gastli & Charabi 2010; Charabi & Gastli 2011; Garni & Awasthi 2017; Merrouni et al. 2016; Alami et al. 2018; Uyan 2013; Borgogno Mondino et al. 2014; Yushchenko et al. 2018) choosing to use a blanket exclusion based on just a slope or a combination of slope and aspect. The numbers range from as low as 3% and 5%, to 10%, 11% and 15% up to 30% and 50%. When combined with an aspect, there is usually two numbers, such as 3% for north facing land (the majority of the studies are northern hemisphere based) and 5% for south facing land. The second school of thought uses the slope as one of the inclusion factors (Arán Carrión et al. 2008; Sánchez-lozano et al. 2013; Sánchez-lozano et al. 2014; Noorollahi et al. 2016; Tahri et al. 2015). This process requires the use of brackets set between two limits, with lower valued brackets being the more preferred and higher valued brackets being less preferred.

Solar radiation. This was used as an exclusion criterion only in studies where the area under investigation has a low average value for solar insolation. This means there are areas where erecting a solar plant would not be financially feasible at current construction cost (Aly et al. 2017; Abed & Nagasaka 2017) with the GHI limit set at 1700 kWh/m^2 and 1277.5 kWh/m^2 annually in Tanzania and Afghanistan respectively. For the remainder of the studies, solar radiation was exclusively used as part of the inclusion factors in a similar fashion to the brackets used for slope.

Road and rail networks. Road and rail networks also play a dual role of exclusion criteria and inclusion factor. Because it is not possible to build a solar plant on top of an existing road or rail it is logical that it be an exclusion criterion. As inclusion factor, it is slightly more nuanced. Road and rail networks are considered as an indicator of how accessible the surrounding area is for development. If an area under investigation is far

away from any existing road or rail network, it is considered inaccessible and excluded from consideration. Typical numbers start around a 100m buffer zone as being preferable up until 5km being undesirable.

Electricity infrastructure. This requirement seems obvious. The generated electricity has to be transported somehow and building an electricity network is not a trivial or inexpensive activity. The way the different authors deal with this requirement varies. Only two studies use access to the electricity network as an exclusion criterion; the rest, if included, tended to choose to add this to the inclusion factors. Areas far away from existing electricity infrastructure would therefore only be ranked as highly undesirable and not completely excluded from consideration. The manner in which this was applied also varied. Some studies chose to use the distance to the nearest sub-station or transformer as the weighted factor, while others were content in using the shortest distance to an electricity line.

Protected areas. This term encompasses a broad range of terminology and definitions assembled from the literature study. What defines a protected area varies greatly depending on the region of study. For instance, in studies conducted in Spain, historic cattle trails were always included as one of the exclusion criteria along with bird sanctuaries, wildlife reserves, paleontological sites and archaeological sites. Military zones were also considered as a protected area for the purposes of this study. In Oman, historic monuments were part of the definition and in Saudi Arabia, this included national parks and sacred places. For all of these definitions, the typical use case was to exclude these areas before continuing to the step of evaluating inclusion factors.

Water bodies. This definition includes lakes, dams, rivers and watercourses. Like road and rail networks, water bodies are also present on both sides. The use as an exclusion criterion is obvious; solar plants are not typically built on dams, but the reasons for adding water bodies as an inclusion factor is slightly more surprising. Depending on local climatic conditions, the solar panels can become soiled very quickly and with this there is a considerable associated performance drop (Gastli & Charabi 2010). A stable source of water for washing of panels is therefore considered an economic advantage to hauling water from a distant source as well as revenue gained from higher panel performance.

Land use (urban). Land use is very broadly defined. Urbanization is considered in many of the literature study cases. Because all of these studies are concerned with utility-scale solar plants and not rooftop PV (which differs enormously in methodology), urban areas are dealt with as an exclusion criterion. Again, urbanization crops up in many studies as one of

the weighted inclusion factors, because the relevant authors are concerned with the losses incurred by transporting energy over long distances. A shorter distance to an urban area is therefore seen as preferred due to fewer transmission losses between the point of generation and the point of consumption.

Land use (agrological capacity). Food security is high on the agenda of many governments. Solar PV is much more expensive in terms of land use than, for example, wind farms. Therefore, if land suitable for high yield agricultural activities overlaps with land suitable for solar PV, then a conflict of interest arises. The way this is dealt with between the various studies is different for each. In the studies where it is included as an exclusion criterion, the limit is usually set within some definition, such as excluding irrigated agricultural land. When it is included as an inclusion factor, it usually relies on a predefined scale such as agricultural yield, if such information is available.

Land type. This criterion is only relevant to a few studies and is highly dependent on the region investigated. For the studies performed on Oman (Gastli & Charabi 2010) and Afghanistan (Abed & Nagasaka 2017) sandy soil and sand dunes were considered as exclusion criteria which were not present in any of the other studies. The latter also included permafrost, which is unique in the literature study.

Temperature. This is only considered in studies performed in areas with high temperatures. This is because, at high temperatures, PV panels lose performance (Dekker et al. 2012) and areas with extremely high temperatures are best avoided, validating its addition as an inclusion factor in certain study areas (Garni & Awasthi 2017; Sánchez-lozano et al. 2013; Sánchez-lozano et al. 2014; Borgogno Mondino et al. 2014; Tahri et al. 2015). Notably, the temperature is never used as an exclusion criterion.

2.2 Chosen Methodology

The generalised process illustrated in Figure 2.1 lacks a background in the methodology behind the studies in the literature study. All of these studies have the objective of revealing the potential for solar PV in a broadly defined region. This region is too broad to generalise on solar insolation alone and hence the exclusion criteria and inclusion factors are brought into the fold in order to reach the set objective, but tied to geographically limited areas within that region. This is the first point of departure between the studies of the literature study and this particular study. The Northern Cape was deliberately chosen as the starting point of the study and this province

can be generalised in term of solar radiation in exactly the way that the regions of the literature could not. If the nine provinces of South Africa were to be ranked on average solar radiation received, including quality, then this province would be ranked at one (Suri et al. 2014). In addition, it would not be meaningful to use a ranking system to order the different mining areas on solar insolation, as the objective is to reveal the collective solar insolation that these areas hold.

This brings into question the function of using weighted criteria at all. The studies in the literature study made extensive use of industry expert interviews to determine the weighted criteria. This methodology automatically brings with it a subjective bias towards solar energy developers. Also, while there were trends visible between the different weights for each study, it varied from study to study and from country to country. Since this study has the objective of informing government, mines and solar energy developers, the best methodology forward is muddy the results with as little bias as possible and leave it to each interested party to build upon the results aligned with their own interests. It was decided that the best way to perform this is by only applying Boolean logic. This is essentially what all of the studies in the literature study performed when applying exclusion criteria.

This logic can be tested with a few assumptions. These assumptions are built on the fact that the chosen study area is limited to mining areas, which is another point of departure from the studies in the literature study. If it is assumed that the majority of mines must have some type of road or rail infrastructure in place to haul equipment and people to the mine and extracted ore from the mine, then would there be a meaningful reason to build in weighted criteria to exclude areas far away from roads? The same argument can be applied to electricity infrastructure. On this subject, the assumption that mines have access to electricity is more concrete, given the energy intensity of mines and the finding that all the mines investigated as part of the study by Votteler & Brent (2017b) were connected to the grid. Lastly, it can also be assumed with relative confidence that mining rights would not be granted for a specific area if that area contained any known historical landmarks, paleontological significant sites, archaeological sites and protected wildlife sanctuaries. The result after removing the requirement of weighted inclusion factors and testing the remaining exclusion criteria against assumptions like those above is shown in Figure

2.2 below. This represents Figure 2.1 after all of the above deductions were applied and useful criteria were customised.
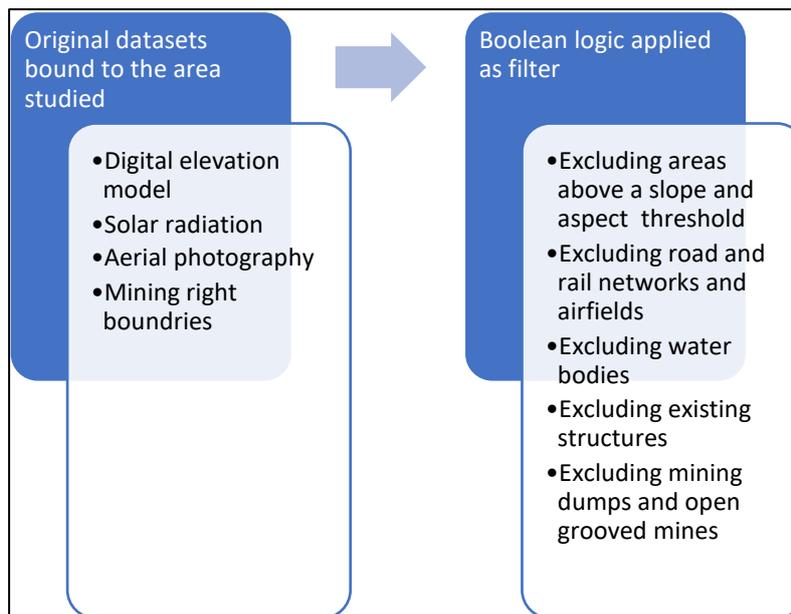
Figure 2.2: Chosen methodology

2.3 Conclusion: Literature Study and Methodology

The aim of this chapter was to perform a literature study to firstly, discover how GIS-based audits were performed by other authors in general, and secondly, specifically how this is applied to utility-scale solar PV energy audits. The literature study revealed that MCDA combined with numerous types of weighted inclusion factors was the method implemented in nearly all cases. Chapter 1 guided the study toward the use of mines in the Northern Cape province and this largely displaced the use of weighted inclusion factors, as the objective of using it was already achieved. Therefore, combining the context provided in Chapter 1 with the methods discovered in the literature study, resulted in the choice of implementing the MCDA as a boolean logic system. After considering each type of exclusion criteria and inclusion factors individually in context of this study, a list of the broad types which will be used to perform a solar audit on the Northern Cape province's mining areas were decided on.

The next chapter discovers the specific nuances that applies to GIS analysis and how GIS analyses can be performed in the R environment. The system developed in this chapter is then applied in detail to achive the main objective of this study.

Chapter 3

3. Applied Simulation Methodology

3.1 GIS Basics

When unfolding a three-dimensional sphere like earth onto a two-dimensional piece of paper, there will inevitably be some type of distortion. An example is the manner in which the Mercator projection distorts the size of the land area in an attempt to preserve shape. This is very extreme near the poles with the most famous example of this being the projected size of Greenland compared to the size of Africa. In reality, Africa is at least 10 times as large as Greenland, while on a typical Mercator projected map, it appears roughly equal in size.

This is not a problem when calculating two areas near each other on the map with the intention of comparing them to each other, for instance calculating the size of Uganda and Kenya. Coincidentally these countries are near the equator where the distortion is the least. For this example, thus, it could just as well have been Uganda and Ecuador being compared. The problem becomes quite prominent with a large difference in latitude, such as Uganda and Iceland, where the land area of Iceland would be greatly over-represented.

Equal area projections, on the other hand, distort shape while preserving area. The result is that some countries might appear elongated, but a square kilometre area in Iceland will be equal to a square kilometre area on a map of Iceland, while also being equal to a square kilometre in Uganda, both on the ground and on the map. This is very relevant as the area under investigation, the Northern Cape, spans at least nine degrees in latitude and therefore the relative error in the calculated area becomes significant.

The equal area projection selected is from the conical family of projections named Albers. While the shape is distorted, as explained in the previous paragraph, Albers conical equal area projections have the characteristic that such visual distortion is limited between the so-called standard parallels. This means that an Albers projection can be adapted to a localised area in order to be accurate while also having nearly no distortion when compared to a Mercator projection in that same area. The *South African Albers NGI Aerials* projection is exactly customised for this purpose.

All the data for this paper resides in two formats. Raster data and shapefiles. Not incidentally, these are the two main types of data storage systems for GIS data, with commercially available software usually favouring either one slightly above the other since translation between the two are not trivial. Raster data are akin to image files, where data is stored as a type of pixel which can also store underlying data as part of the properties of each pixel. Each pixel location is typically referenced from the top left corner in an XY grid with the absolute location of each pixel being calculated with respect to this corner. Projection and datum information might be stored in a separate file or in the header of the file itself. This, combined with the XY grid is used to provide the absolute pixel position. Usually, there is also some type of system to cater for multiple layers, which provide a way of storing various types of different data for the same location. In its simplest form, it provides a way of storing the three different colours of an RGB aerial photograph.

Shapefiles, on the other hand, store data in a completely different manner depending on the use-case with the intent of minimizing data storage. Typically the shapefile framework consists of three files with extensions .shp, .shx and .dbf with geometry data in the first and the spatial index and attribute data in the second and third (Bivand 2006), respectively. A polygon shapefile, for instance, would only store the coordinates of the beginning and end point of each line segment. The polygon is then only drawn once the content of the file is requested to be displayed. The polygon is drawn according to a predetermined order between the saved points. Each element, or polygon in this instance, can hold several data elements such as a height parameter for topographical shapefiles. Polygons shapefiles also cater to the concept of *holes*. For example, this allows for drawing a flat version of a doughnut where the area on the inside of the inner circle is not part of the shape as a whole. As a more relevant example, this would allow for the creation of a shapefile of the borders of South Africa including the area which should not be included where Lesotho is located. The hole itself is drawn using exactly the same method as all of the other polygons with only an entry in the index of the shapefile identifying the relevant holes. Shapefiles consisting of lines or points are simplified versions of the same structure.

3.2 Data Sources

As mentioned earlier in Chapter 1, the competitive bidding process for RE projects was indeed very successful. It is this success which in turn makes it difficult to source data on nearly any aspect of RE projects, as

competitors will not likely publish their data, for fear of losing a competitive edge. To address this obstacle, data from government institutions were given preference. In many cases, government agencies collect data with taxpayer money and this, in turn, places the data back in the hands of the public taxpayer. In SA, this is regulated through the Promotion of Access to Information Act, which is the legal mechanism whereby right of access to information held by the state is executed, as promised by the Constitution of SA. Similarly, the United States of America enacted the Freedom of Information Act, which bounds Federal Agencies such as NASA, to make available any information requested by the public. While these laws were designed to promote transparency in government, they had the positive effect of some agents of government taking the liberty to make selected data readily available to the public without request.

3.2.1   Solar Resource

Solar resource maps can be created from two main sources: Ground measurements and satellite-derived data. The latter typically covers very large areas and are therefore suitable to exploratory work. The drawback is both a coarse resolution and inaccuracies due to how the light attenuation through the atmosphere is derived. Sources of such attenuation include ozone, air molecules, aerosols, clouds and water vapour. For each of these either an estimated or measurement derived compensation needs to be included, which can dramatically affect the final result. Indeed, Winkler et al. (2012) found that inadequate compensation for irradiation losses due to atmospheric aerosol concentrations lead to inaccurately derived results. On the positive side, the datasets available usually offer much longer time periods, which alleviates at least some investor fears, as solar plants are built to be operated for decades.

For ground measurements, a measurement device, such as a pyranometer or a pyrheliometer is required. Both these devices are extremely accurate, but as a consequence also very expensive. They also require to be deployed in many locations, as the results recorded by the device is only relevant to the area immediately surrounding the measurement position. Combined with the high cost, it is then understandable why Fluri (2009) reported only 11 weather stations, managed by the South African Weather Services (SAWS) in the whole of SA equipped with such devices. This does not cover a large footprint and ss a consequence, there are currently no solar resource maps of SA, publicly available, created from such measurements. The decision to use satellite-derived data is then not only a conscious one but a necessary one due to a lack of alternatives. This leads to the question of which satellite-derived map to use. The available solar resource maps for

SA are discussed by three papers (Zawilska et al. 2012; Fluri 2009; Winkler et al. 2012) in the local context and a summary of these are given in Table 3.1 below. For each option, the last column indicates in what form the data is available and the second last column indicates whether the data is freely available or restricted in some manner.

Table 3.1: Various sources of solar maps relevant to local conditions. Source: Fluri (2009), (Winkler et al. 2012), (Zawilska et al. 2012)

Map Name	Author	Year	Availability	Data Source
Solar radiation data handbook for Southern Africa	-	1990	-	Ground measurements
South African Renewable Energy Resource Database	CSIR, Eskom, DME	1999	Public	Ground measurements
Solar and Wind Energy Resource Assessment (SWERA)	USDE, NREL & DLR	2006	Public	Satellite
Surface Meteorology and Solar Energy (NASA SMSE or SSE)	NASA	2008	Public	Satellite
Photovoltaic Geographical Information System	European Commission Joint Research Centre	2007	Public	Satellite
Meteonorm	Meteotest AG	2009	Commercial	Satellite & ground measurements
SolarGIS	Solargis	-	Commercial	-
3Tier	Vaisala	-	Commercial	-

This shows that there are four sources that would satisfy the objective to be freely available to promote reproducibility. However, the majority of these could only be obtained in the format of a high-resolution image file, which is not sufficient for the purpose of this research. In the end, only the NASA dataset was found to grant access to the full raw dataset before any variables are included through manipulations in an unknown software environment. The NASA dataset was obtained from the OpenEI organization in the form of a 57 megabyte (MB) Environmental Systems Research Institute (ESRI) Shapefile containing a grid of 64 800 square polygons covering the entire surface of planet earth. A more detailed

description of this shapefile layout and subsequent computations done to reduce it follows in Chapter 3.6 of this paper.

3.2.2   Mining Operations

According to Regulation 2, sub-section (2), paragraph (f) of the Mineral and Petroleum Resources Development Regulations, as enacted by die Mineral and Petroleum Resources Development Act (2002), to obtain a mining permit the application should include a plan that includes "the boundaries of the land to which the application relates." Contacting the Department of Mineral Resources (DMR) to obtain these maps has, to date resulted in no resolution. However, within data sourced from the Northern Cape branch of the Directorate: National Geo-Spatial Information (NGI), a branch of Department of Rural Development and Land Reform, shapefiles were included which contain areas with mineral rights. It is assumed that these shapefiles were the verified product of the above-mentioned requirement from DMR, processed and formalised by NGI. This happens to be the preferred dataset, as areas with approved mining permits are subsets of areas with mineral rights, which in turn are subsets of areas where exploration rights were granted (Department of Mineral Resources 2002; Hall 2012), as illustrated by Figure 3.1 below.
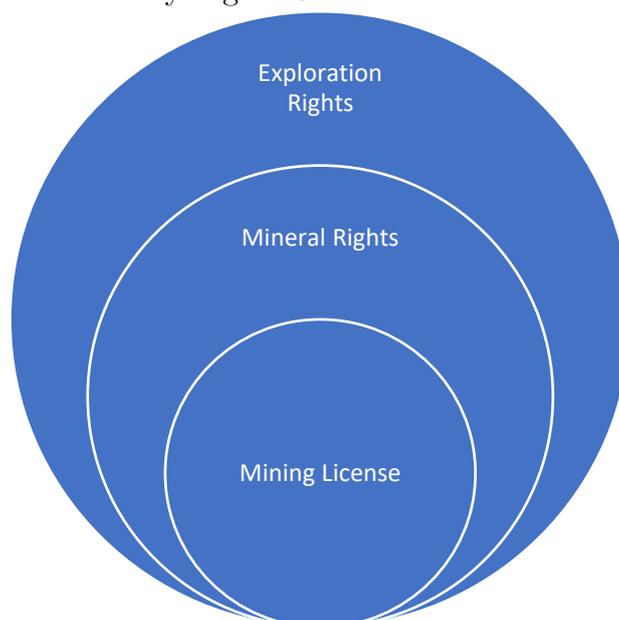
Figure 3.1: The relationship between the rights and licenses granted by the Department of Mineral Resources indicates that a database of mineral rights must include areas with valid mining licenses.

The dataset itself consists of polygons saved in the ESRI Shapefile format and has a negligible size of 386 kilobytes (kB). The data is ordered by a field referenced as its GID and consists of 147 GID's and it can be

deduced from this that there are as many areas in the Northern Cape holding valid mineral rights. A validation of this will be discussed in Chapter 3.8 while a more detailed description of this shapefile layout and subsequent computations follows in Chapter 3.6 of this paper.

3.3.3 Aerial Photography

While software such as Google Earth have numbed the general admiration for the task of aerial surveillance, finding a credible source of high quality, properly catalogued and projected aerial photography has highlighted the mammoth effort behind this task. Once again, the NGI Head Office in the Western Cape was the preferred point of contact and supplied these at zero cost. Figure 3.2 shows how the aerial photography for the entire country is organised, with the colour of each block indicating the latest year of survey (Department of Rural Development and Land Reform: National Geo-Spatial Information n.d.).
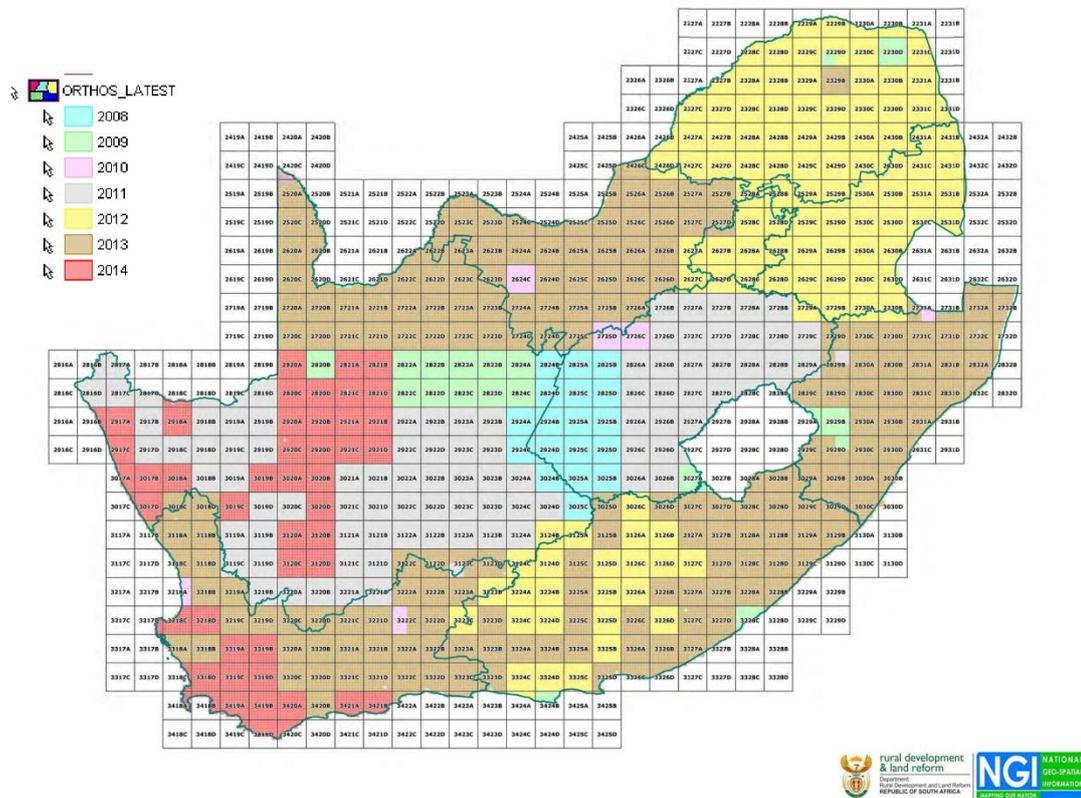
Figure 3.2: Grid showing the division of NGI datasets. The legend colour indicates the latest year of photography. Source: (Department of Rural Development and Land Reform: National Geo-Spatial Information n.d.)

As a starting point, all the data had to be collected that were inside the borders or touching the borders, of the Northern Cape province. This

represents a mammoth 1.32 terabyte (TB) of data. Each file is encoded as a GeoTIFF file and accompanying TIFF World File (.tfw), which is a sidecar file developed by ESRI to accompany GeoTIFF files. It describes the location, scale and rotation of a raster on a map, as used by GIS systems. The GeoTIFF itself is also encoded with information such as projection, coordinate reference system, ellipsoid and datum, while still remaining usable to software only making use of the standard TIFF format.

The NGI does not provide an index of how image files are organised and this had to be explored empirically. Figure 3.2 provided a good clue and eventually, the system was unravelled and illustrated here in Figure 3.3 below. The location of each file is contained within the file name and the system consists of nested squares. The outer reference is provided by the first integer latitude to the North of the current location and the first integer longitude to the West. These numbers arranged sequentially without spaces or special characters provide the first four characters in the file name. The next level of resolution is provided by dividing the spaces between the current and the next latitude/longitude pair into four squares. These are named A, B, C and D, starting at the top left (North West) and moving

clockwise. The next level of resolution (children) is then referenced by the current block name (parent) and a suffix with the same logic as described for the parent level i.e. AA, AB, AC and AD in the case of parent block A. Lastly, at the point of maximum resolution, each block is divided into 25 (arranged five by five) blocks, starting at the top left (North West), but this time numbered sequentially from left to right until the end of the line, before continuing on the left of the next line. An example of such a filename would then be *2917CA_06.jpg* which shows that the first alphabetical level of resolution (A,B,C,D) is implied and not included in the encoding of the name.

Integer Longitude			
A		B	
AA	AB	BA	BB
AC	AD	BC	BD
CA	CB	DA	DB
CC	CD	DC	(grid 1–25)
C		D	

Integer Latitude (left vertical axis label)

The DC column contains a 5×5 grid:

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

Figure 3.3: Reference framework of how NGI data is stored

3.3.4   Topographical Data

Reliable raw topographical data presents the same problem as with aerial photography. Again, the NGI Head Office supplied the required data at zero cost. The NGI topographical data is sampled at 10m intervals in rural areas and 5m in built-up areas. For this study area, the former was found to be the most prevalent type. The reference system used is the same as described for the aerial photography in Chapter 3.2.3 above. The main difference is that the topographical data is stored as ESRI Shapefiles consisting of multiple polylines. The consequence is that the data size on disk is much smaller and therefore the parent blocks are not subdivided into so many child blocks. The exception is located on borders with neighbouring countries and the ocean, where the reference system nests down to the first alphabetical level. A second consequence is that parameters such as projection, ellipsoid and datum are encoded within each file, as this is a basic requirement of ESRI encoded shapefiles.

40

3.3.5   Cleaning Data

In a perfect world, it would be easy to trust the source of data and continue with the manipulation of the data without considering the possibility of incorrect or corrupt entries in the data. In reality, provision has to be made to clean data by removing invalid entries and correcting formatting and any other errors which might occur. For the acquired data discussed in the chapters above, the aerial photography contained no errors, but a number of errors were contained within the shapefiles describing the geographical areas with mining rights.

To investigate these errors, an R library named *cleangeo*, created and maintained by Emmanuel Blondel (2017b) was imported. This library includes the functionality to provide a summary report of all the errors found within the input data. Table 3.2 shows the result of running this on the mineral rights dataset.

Table 3.2: Summaries of polygon dataset before and after clening data.

Validity errors	3	0
Orphaned holes	3	0
Logical mode errors	6	0

Orphaned hole errors are directly related to the structure of a shapefile. A hole in a shapefile is defined as a polygon wholly contained within another polygon. The purpose of this would be, for example, to describe a piece of land geometry, but without including bodies of water lying within said land in the total land area. The structure of a shapefile deals with this in a two-stage process. Firstly, a logical value in the header of each polygon entry, aptly named *hole*, indicates whether a specific polygon is classified as a hole or not. A polygon is referenced to by its *ID* entry in the header. Secondly, further up in the shapefile hierarchy a field named *plotOrder*, containing a list of IDs dictates the order in which polygons are plotted. In its most basic form then, an orphaned hole error occurs if, for some reason, the *plotOrder* field does not include all the IDs of all the polygons within the shapefile, and the excluded polygon has a *hole* value of TRUE.

Validity errors are more general and can include a whole array of reasons which qualify as validity errors. The most common validity error found in the data is due to self-intersecting polygons. The lines that polygons consist of, by definition, cannot cross each other. A shapefile holds a number of coordinate sets which define all the corners of the polygons. The lines are created by plotting from each coordinate set to the following until the last set is reached. To complete the polygon, the last set of coordinates have to

41

match with the first. In some cases, for unknown reasons, the last digits of these coordinates differ slightly. Not enough to be visible when viewing the extent of the polygon, but enough so that zooming in sufficiently will show different start and end coordinates. Conceptually this error is similar to the way in which two seemingly equal real numbers are not equal when enough decimal places are included in the comparison.

As shown in Table 3.2, *cleangeo* managed to identify and rectify each of these errors, however, these are only the types of data errors that become highly visible once trying to manipulate the data. Other errors are not as visible and have more to do with the assumptions placed on the global structure of the data. In the case of the mineral rights shapefile, this assumption was that the regions represented by the polygons in the shapefile cannot be contained or overlapping with each other. This discovery leads to a considerable manual effort to compensate for a structure in the data which doesn't allow for the easy discovery of these features. The resolution of this problem involved plotting all the mineral rights areas and manually iterating through the set of 147 polygons and interpreting the relation each polygon has with the other polygons in the set. Table 3.3 summarises the types of relations found and the number of each case.

Table 3.3: Types of encountered data problems and their respective occurrence.

Type of data problem	Number of affected regions
Fully contained (within another area) areas	27
Region part of a supergroup with a parent region which represents the sum of the group	6
Region part of a supergroup with a parent region (parts do not equal the sum)	13
Overlapping regions with no parent region	2
Exact duplicates	2

The simplest and most prevalent case is where a region is fully contained within another region. This makes the contained region irrelevant to the overall objective and creates the danger of overexaggerating the contribution of the containing area if left in the computations. To deal with this, a variable was created with the IDs of all the contained regions. This variable was routinely included in the R scripts that performed calculations and manipulations on the data to ensure their exclusion.

A second case is where a region is split into multiple sub-regions which do not overlap each other but does touch at the points where they

42

meet, and the sum of the regions equals the original region. This is not similar to the first case, but the resolution is the same and therefore the IDs of these regions were also included in the variable which stores the exceptions.

The third case is similar to the second case but differs only in the last requirement where the sum of the regions equals the original region is not met. Even with this requirement not met, the effect of these areas is the same as with the second case and are therefore dealt with in the same manner.

The last case is the most difficult to deal with, but only one such case was discovered. In this case, the regions follow the shape of the original polygons, as with the second and third cases, but the sub-regions overlap each other and therefore the sum of the sub-regions are greater than the original region. Because there was only one such case discovered, the decision was taken to deal with this case using the methods that were developed to identify and define excluded areas, as will be explained in Chapter 3.8.2, and define the overlapping area as an exclusion. The objective of the excluded areas is to be subtracted from the main investigated areas and thus, even though the motivation differs in this instance, the result yielded would satisfy the required objective.

3.4 Scripting Code Versus Manual Effort

Since the advent of modern computing, or possibly even the industrial revolution, users of machines have dealt with the decision of when to automate versus continuing to perform manual tasks. In the well-defined case of performing repetitive tasks, the decision can be simplified by considering the graph shown in Figure 3.4 below. This shows that the output of performing repetitive tasks manually yields results in a linear fashion. That is, a single task requires a unit of time to complete and completing two tasks would require roughly two units of time. If the tasks are relatively equal in size, then the amount of time to complete all of the tasks can be easily extrapolated. This stands in contrast with the second line on the graph showing the output of automating the process of completing repetitive tasks. In this case, there is a high cost involved at the beginning, where zero output is delivered while time is spent on developing the system of automation. Once this system is put to work, however, tasks are completed at a much higher rate than compared to performing them manually. The cost, in units of time, spent developing the system is quickly recovered and from the point where the lines on the graph meet, a profit is being made.

This concept is a common theme that will be referred to during Chapter 3.7 and 3.8 to justify the R scripts that will be used to perform repetitive tasks. Although, it will not be formally analysed as such, but rather as a decision based on experience.
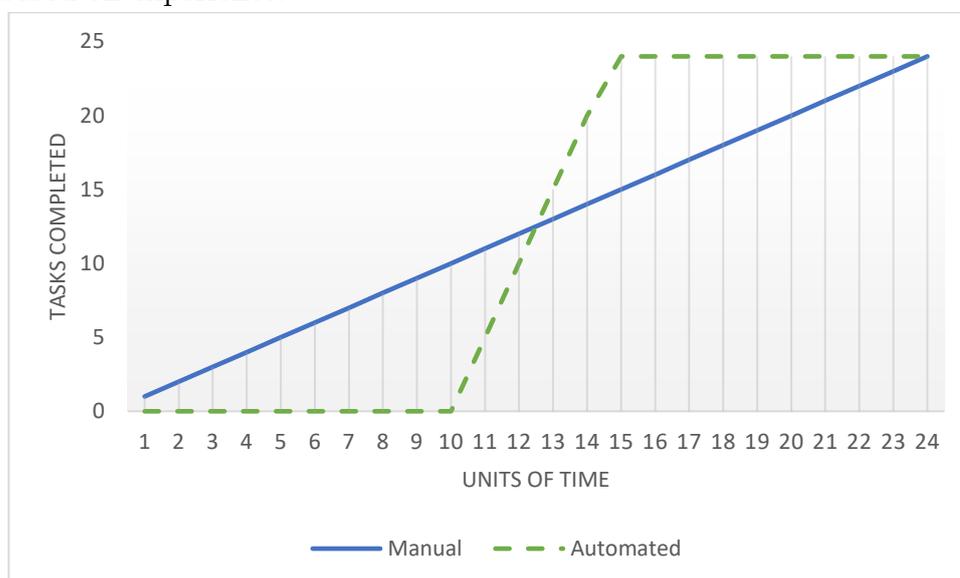
Figure 3.4: Simplified graph explaining the difference in time spent during manual versus automated processes.

3.5   Overview of R as Simulation Tool

One of the minor objectives of this paper is to promote reproducibility through accessibility, thus open-source options were an obvious direction, as the seat-cost for GIS work in the academic community is prohibitive (Bivand & Gebhardt 2000). At least since the start of the millennium, the uptake of R (R Core Team 2015) in the underfunded teaching and research sphere of developing countries has been noticeably high (Bivand 2006). Research on using R in an academic environment to perform GIS functions only revealed outdated papers. In a paper by Bivand (2006), it is reported that a search on the R website for the term "spatial" yielded 1 219 results, which was considerably more than the 447 results in the year 2002. Repeating this in 2018 yielded 7 280 results, showing that the initial increase in momentum did not cease.

At the beginning of the research, it was not exactly known what would be required and if any chosen GIS software package would be able to perform these tasks. A paper by Endel & Filzmoser (2012) indicated that R had support for GIS and allowed for easy reproduction of steps through the use of script files, which was seen as an important factor at that stage because of the many areas under investigation and the possibility of repeating many operations on each. More importantly, it was found that R had a big and vibrant online community and this was a comforting factor

that any shortcoming could be overcome through discussion and engaging the community of experts.

On recommendation by Endel & Filzmoser (2012) RStudio was selected as the preferred graphical user interface. The basic R distribution is very capable of computing mathematical and statistical data functions (Grunsky 2002), but to perform true GIS operations extra libraries need to be installed. A summary of these is given in Table 3.4 below.

Table 3.4: Summary of all additional libraries added to the base R

Library	Author
cleangeo	(Blondel 2017a)
dplyr	(Wickham & Francois 2016)
gstat	(Graler et al. 2016)
maptools	(Bivand & Lewin-Koh 2017)
sp	(Pebesma & Bivand 2005)
raster	(Hijmans 2016)
rgdal	(Bivand et al. 2016)
rgeos	(Bivand & Rundel 2017)

3.6 Preparing data

3.6.1 Filtering and Selecting Data for a Subset of Aerial Photography

As mentioned in Chapter 3.2.3, the decision was taken to collect aerial photographs of the entire Northern Cape province as a starting point. This amounted to terabytes of data which is not only cumbersome to store but slows down any computation. It would, therefore, hold an advantage to first reduce the size of this dataset to a subset that approximates the actual size of the study area better.

The following is pseudo-code for the R script that was created to perform the objective stated above.

1. Read in the shapefile containing all the mineral rights defined areas using the readOGR() function contained in the rgdal() library.
2. Assign the correct coordinate reference system to the shapefile, now stored in memory.
3. Create a plot of the entire shapefile to use as a simple visual aid to monitor progress.

4. Create a list of all the folders in which the aerial photos are stored. The original dataset is sorted according to latitude and longitude and stored in folder names similar to the description given in Chapter 3.2.3 relating to the file names.

5. Begin a loop that iterates through the list created in 4.
    a. Set the current folder to the ith index of the list.
    b. Create a list of all the aerial photographs found in this folder.
    c. If the list created in b is not empty, start a loop that iterates through this list.
        i. Load the jth image in the list from disk and temporarily store in memory as a raster using the raster() library.
        ii. Check to see whether this image overlaps with, or is contained within, any of the regions contained within the mineral rights shapefile using the gContains() and gOverlaps() functions from the rgeos() library.
        iii. If the check in *ii* yielded any true values, then the TIFF file containing the aerial photograph as well as its accompanying TIFF World File is copied to a new folder containing all the relevant aerial photos.
        iv. Plot an outline of the current image onto the map plotted in 3 to provide a means of visual feedback to the user as an indication of progress.

6. Write all collected errors and warnings to a log.

Given the size of each aerial photograph ranges between 70MB to 80MB, just the task of reading a single photograph to memory takes a while, whether done in R or any other program that can open TIFF files. This implies that even if a manual comparison could have been done relatively quickly, the entire task could have taken an extremely long time to complete if done manually through repetitive tasks. The fact that a computer eventually completed this filtering task in just under 48 hours of uninterrupted computing justifies the amount of time spent on writing the R script when considered in the context of Chapter 3.4 above.

The complete set of R code contained in the script as described above can be found as an addendum to this document.

3.6.2 Filtering and Selecting Data for a Subset of Solar Insolation Data

As mentioned in Chapter 3.2.2 above, the dataset acquired for solar insolation, contained data for the entire planet organised in a grid of 64800 cells and saved in an ESRI encoded shapefile. In order to reduce the size of this dataset to something more manageable, it was compared to the extent of the Northern Cape province and all cells that fell outside this boundary were discarded. The decision was made to use the Northern Cape boundaries instead of the extent of the shapefile containing the entire group of areas containing mineral rights for the following reasons:

- The resolution of the NASA solar insolation data is quite large. Each cell spans one-degree latitude by one-degree longitude.
- Whatever the extent of the entire mineral rights shapefile could be, it would have to be rounded up to the next adjacent latitude and longitude regardless, given the point above.
- Visual inspection of the mineral rights shapefile revealed that areas containing mineral rights are scattered from the coastline on the West to the next province on the East and from nearly the Northernmost point of the Northern Cape to the next province in the South.
- The resulting reduced dataset would not differ greatly in size if it included one-degree latitude and/or one-degree longitude of data, because of the characteristics of shapefiles.

The above meant that even if it was little, a small amount of computational time would be saved if the extent of the entire Northern Cape province is used. The method employed made use of two newly created R functions and an R script that performed some basic tasks and made use of these two functions. A pseudocode description of these follows in the paragraphs below.

The first function is very simple and takes a single input parameter, being a matrix of coordinates. It assumes the coordinate matrix to be of the form:

$$
\left\{
\begin{array}{ll}
\text{longmin,} & \text{longmax} \\
\\
\text{latmin,} & \text{latmax}
\end{array}
\right\}
$$

because this is how it was found to be stored in the Northern Cape shapefile. The function then rounds these numbers up or down, depending on the requirements for each. The function then produces an output matrix in the

same form as above. The rounding is performed in the following way using the base R functions *floor* and *ceiling*:

1. longmin is rounded down.
2. longmax is rounded up.
3. latmin is rounded down.
4. latmax is rounded up.

The second function performs a slightly more intricate computation. The purpose is to translate the physical location of the Northern Cape boundaries to the grid reference used to catalogue the NASA data in a 64800 cell grid saved within a shapefile. Figure 3.5 shows the grid layout, where the coordinate [-90,-180] corresponds to the bottom left corner of the bottom left cell, numbered as one and the top right corner of the top right cell, numbered 64800, corresponds to the coordinate [90,180]. Continuing with this logic, it can be seen that each row of cells fall within the same degree of latitude, while each column of cells falls within the same longitude. The function expects two sets of coordinates, describing the extent of an area, as inputs.
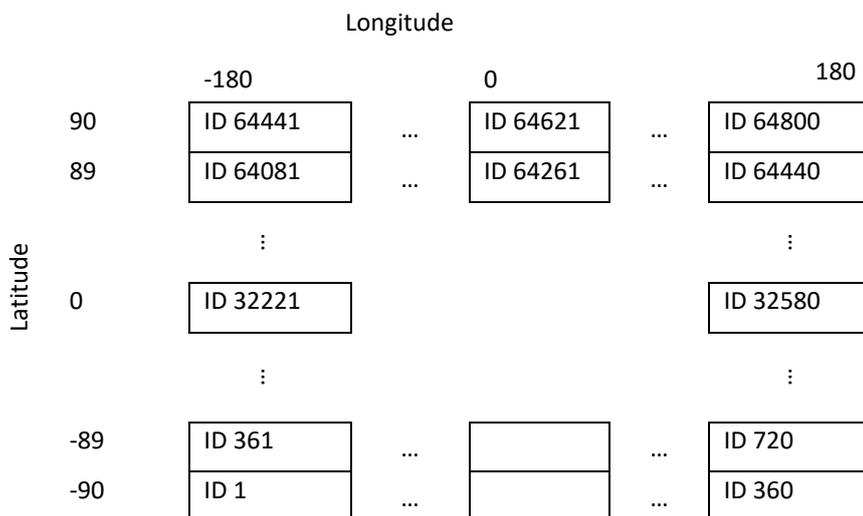
Figure 3.5: Reference system of the NASA solar radiation data

The following pseudo-code describes how the translation between the two reference systems is achieved:

1. Start a *for* loop at the latitude minimum and ending at the latitude maximum.
   a. Start a nested *for* loop at the longitude minimum an ending at the latitude maximum
      i. The translated reference is then calculated using the formula:

$$r \quad = \quad |\text{-}90 \quad \text{-} \quad i| \quad \times \quad 360 \quad + \quad |\text{-}180 \quad \text{-} \quad j|$$

where i is the current latitude of the outer *for* loop and j is the current longitude of the inner *for* loop.
  ii. Store this reference along with all the other references in a vector.

The last step is to combine these functions into a single R script that selects the appropriate subset of data and discards the rest. The pseudo-code for this script follows below.

1. Read the NASA data into a variable using the *readOGR()* function contained in the *rgdal()* library.
2. Read the shapefile describing the boundaries of the Northern Cape using the *readOGR()* function contained in the *rgdal()* library.
3. Extract the extent of the Northern Cape shapefile, from step 2, using the *bbox()* function from the *sp()* library.
4. Round the limits, from step 3, using the custom function described in the above paragraphs.
5. Translate the limits, from step 4, using the custom function described in the above paragraphs, to obtain a vector containing the corresponding references of the relevant subset in the NASA dataset.
6. Use the vector, from step 5, to select and store the relevant subset in a temporary variable.
7. Write the subset generated in step 6 in a shapefile to disk, using the *writeOGR()* function from the *rgdal()* library.

The complete set of R code contained in the script and the functions as described above can be found as an addendum to this document.

3.6.3 Filtering and Selecting Data for a Subset of Topographical Data

As mentioned in Chapter 3.3.4, the topographical data is stored in shapefiles. These are very efficient because it only stores data for areas where there is data and the areas in between require no extra storage or computation. Even so, the topographical shapefiles each contain the data covering one-degree latitude by one-degree longitude. This represents a length of 111km, in the case of latitude, which is a very large dataset considering the resolution is 10m. This chapter only deals with how the dataset comprising of all topographical data for the Northern Cape province was reduced to only the blocks relevant to the mineral rights areas.

In this case, the time relationship discussed in chapter 3.4 is nearly at the break-even point. However, there is one consideration that hasn't been discussed until now, and that is the level of expertise that increases over time. When writing more scripts and exploring more methods of dealing with GIS problems in R, it also increases the level of experience and with that the speed at which new scripts are created also increases. This also ties in with the main objectives and it was therefore decided to, once again, write an automated script to perform this sorting task.

The topographical shapefiles exist on disk within many other shapefiles, all sorted within sub-directories according to their latitude and longitude. Therefore, the basic strategy employed is to iterate through each sub-directory, identify the topographical shapefiles, compare it to the mineral rights areas and copy the shapefile to a different location if relevant. The following pseudo-code describes the R script that performs this task

1. Create a variable with all the names of the sub-directories containing the topographical data.
2. Read the mineral rights data into a variable using the *readOGR()* function contained in the *rgdal()* library.
3. Start a *for* loop at the beginning of the list created in step 1 and continuing to the end.
   a. Set the current working directory to the ith entry in the list from step 1.
   b. Search the current directory for all shapefiles containing *"HYPS_ELEVATION_LINES"* in the name and store in a list.
   c. Start a *for* loop at the beginning of the list created in step b and continuing to the end.
      i. Read the jth shapefile into a variable using the *readOGR()* function contained in the *rgdal()* library.
      ii. Compare the shapefile from step i to the entire mineral rights shapefile from step 2, checking for any overlaps.
      iii. If any of the comparisons of step ii yields TRUE, copy the current shapefile to a new destination directory.
      iv. If step iii is performed, then also copy the corresponding *.dbf* and *.shx* files.
4. Write all errors and warning to a log file.

The complete set of R code contained in the script as described above can be found as an addendum to this document.

3.6.4 Structuring Refined Data

Keeping large sets of data from different sources organised was found to be key in eventually producing a reliable result. Often, in the process of discovering the limitations of a certain type of data or the result of a certain calculation, it reveals a new problem that was not thought of initially. A well-organised dataset allows the grouping of associated datasets and sub-datasets together as well as indexing data to allow for the tracking of exceptions.

Here the decision was taken to use the unique ID, called *GID*, of each mineral rights area as the index to all other sub-datasets which are the result of calculations made on the main data sources. The *GID* was briefly mentioned in chapter 3.2.2 with the introduction of mineral rights shapefile supplied by the NGI. A directory structure was created with these *GIDs* and within each, there was sub-directories which acted as a placeholder for the result of future calculations. Figure 3.6 shows the generalised structure of the directory tree. *GID1, GID2* and *GID3* represent three of the 147 *GIDs* contained within the mineral rights shapefile forming the basis of the index. Each of the sub-directories provides a space for the calculated inclusions and exclusions. These are duplicated within each *GID*. The creation of these directories and sub-directories are a good example of where scripts trump manual effort. A small R script was written in a short amount of time that

created the entire directory structure in seconds. The R code contained in this script can be found as an addendum to this document.

GID1	GID2	GID3
Aerial	Aerial	Aerial
AspectSlope	AspectSlope	AspectSlope
DEM	DEM	DEM
Exclusion Objects	Exclusion Objects	Exclusion Objects
Solar	Solar	Solar
Topo	Topo	Topo
Plots	Plots	Plots
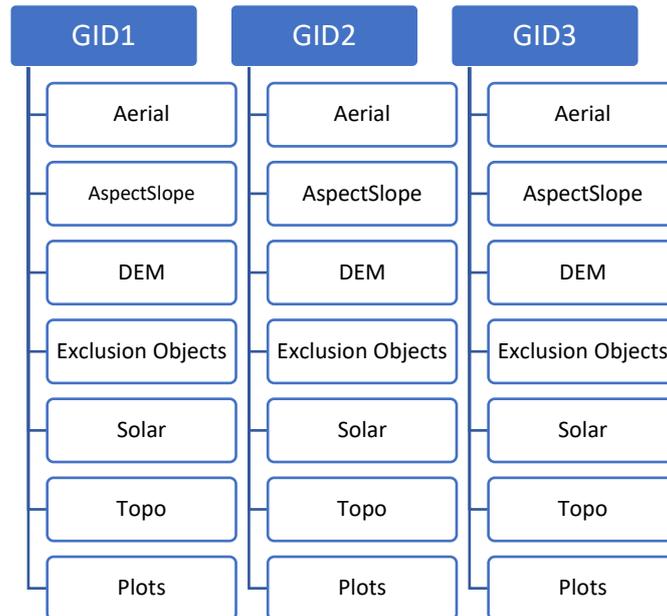
Figure 3.6: Directory structure of the refined data

3.6.5 Selecting, Sorting and Cropping Aerial Photography per Mine

While the filtering process described in chapter 3.7.1 reduced the size of the aerial photography data from 1.32 TB down to 312 aerial photos to the sum of 22.9 GB, the data is still unsorted and only referenced by its filename, as described in chapter 3.3.3. This chapter describes how aerial photos were selected based on its location, then sorted according to the sub-directory structure, as described in chapter 3.7.4, and lastly cropped to the extent of the relevant mineral rights area.

Selection is based on the mineral rights shapefile and the 147 *GIDs* contained in it. This requires a spatial comparison of an image and a shapefile to be conducted. Practically, this is performed by reading the image as a raster and performing the computations on the raster. Rasters have been discussed in chapter 3.1, but it is worthwhile mentioning that rasters are generally much larger than shapefiles and computations between the two formats were not easily achieved in R.

The second action, sorting, is achieved by implementing the directory structure from the previous chapter. This is done by merely copying an aerial photograph from its original location to the directory corresponding to the *GID* of the currently investigated area during the selection process. R also has the built-in capability to manipulate files in the host operating system.

Lastly, cropping is achieved by comparing each aerial photograph overlapping a certain *GID* and cropping the containing raster with the extent of the corresponding polygon. It should be noted that the cropping action was limited to the extent of a polygon and not on the boundary of the polygon. The latter is a computationally much more expensive action to perform and provides no additional advantages, bar being visually more appealing, and negatively affects interpretation through a loss of context at the borders of polygons.

The below pseudo-code briefly describes how the above actions are performed.

1. Read the mineral rights data into a variable using the *readOGR()* function contained in the *rgdal()* library.
2. Create a list of all the TIFF aerial photographs in the filtered directory.
3. Start a *for* loop at the beginning of the list created in step 2 and continuing to the end.
    a. Read the ith aerial photograph into a variable using the *stack()* function from the *raster()* package.
    b. Determine the extent of the raster image using the *projectExtent()* function from the *raster()* package.
    c. Coerce the extent to a polygon using the base R *as()* function.
    d. Start a *for* loop at the beginning of the list created in step 1 and continuing to the end.
        i. Isolate the jth *GID* polygon and store it in a temporary variable.
        ii. Check whether the polygon from step i intersects with the extent polygon from step c.
        iii. If step ii yields a TRUE then crop the raster with the outline of the polygon using the *intersect()* function from the *raster()* package.
            1. Change the current working directory to the corresponding *GID* and save the cropped raster as a new TIFF file using the *writeRaster()* function from the *raster()* package.
4. Write any errors and warnings to a log file.

One important aspect of the script design is the order of nesting of the *for* loops. If the *for* loop from step 3 and the *for* loop from step d were reversed, the execution time of the script as a whole would have increased dramatically. This is because loading the images to a raster is a computationally expensive operation. In fact, even with the order as is, a time-saving improvement was introduced with steps b and c. The comparison made during step ii proved to be computationally expensive and a big saving was made by extracting the extent of the investigated raster and comparing this to the mineral rights polygon rather than using the investigated raster itself. This time-saving effect this had is unknown but given that the total computational time for this process nearly reached three days, any savings were welcomed.

(a)   (b)
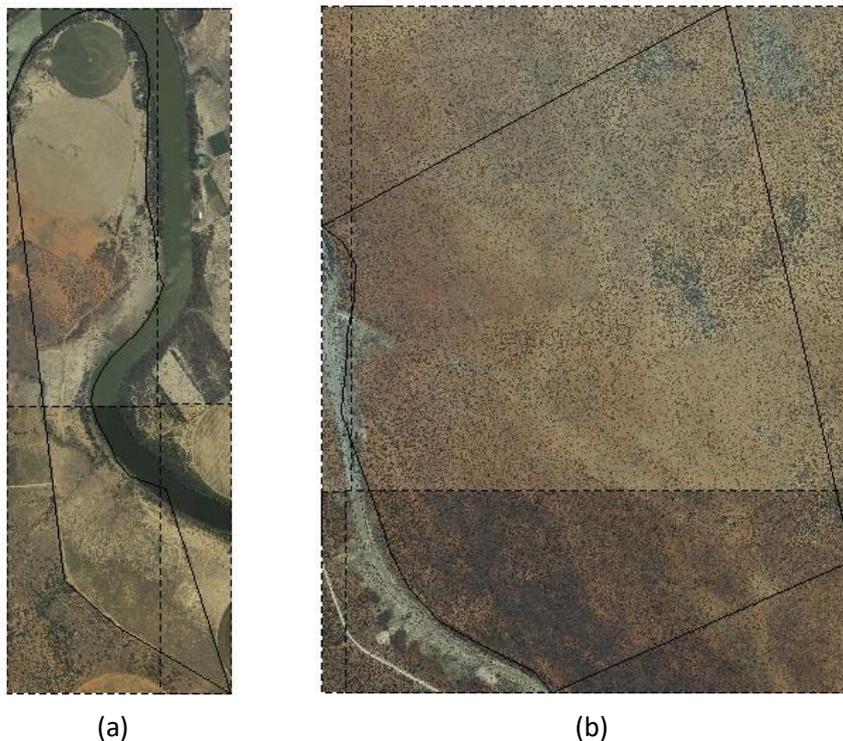
Figure 3.7: Plots of the mineral rights areas corresponding to GID 1710351 (a) and 1710356 (b) with the outlines of the cropped aerial photos shown with dashed lines.

Figure 3.7 shows an example of the result of this process. For clarity, the extent of each cropped aerial photograph is included with dashed lines, but this is not a necessary part of the result. Each consists of four cropped aerial photographs. Using Figure 3.7 (a) as an example, the top left cropped aerial photo represents the bottom right corner cropped out of a much larger aerial photograph which protruded into the extent boundary of the mineral rights area. As can be seen, the extent is a rectangular bounding box.

3.6.6 Cropping Solar Insolation Data per Mine

The process followed to achieve this is very similar to the process followed in the previous chapter, however much simpler. In this instance, the NASA solar data is stored in a shapefile and the mining data is also stored in a shapefile with polygons. This means the comparison is much more generic and therefore much simpler to include in a script. It also means that the computations performed are much simpler and therefore executed in much less time. The following pseudo-code describes the high-level process followed.

1. Read the mineral rights data into a variable using the *readOGR()* function contained in the *rgdal()* library.
2. Read the NASA insolation data into a variable using the *readOGR()* function contained in the *rgdal()* library.
3. Start a *for* loop at the beginning of the list created in step 1 and continuing to the end.
    a. Start a *for* loop at the beginning of the list created in step 2 and continuing to the end.
        i. Compare the ith mineral rights area to the jth NASA insolation area using the *gIntersects()* function from the *rgeos()* library.
        ii. If step i yields a TRUE result, then clip the relevant NASA insolation area with the outline of the current mineral rights area using the *intersect()* function from the *raster()* library.
            1. Change the current working directory to the corresponding *GID* and save the cropped shape as a new shapefile using the *writeOGR()* function from the *rgdal()* package.
4. Write any errors and warnings to a log file.

In step ii, the *intersect()* function from the *raster()* library was used. This was not the only option available to perform this operation. Another library that was extensively used in the course of this project is the *rgeos()* library which includes a similar function, *gIntersection()*. The reason that the former was chosen, is because, in that function, the attributes of both input shapes are merged in the output, whereas the latter function outputs only the result of the intersection and passes none of the attributes down to the output. This is not an acceptable outcome, as the main purpose of

performing this function is to obtain the NASA insolation attributes contained within the input and pass it on to the output.

(a)   (b)
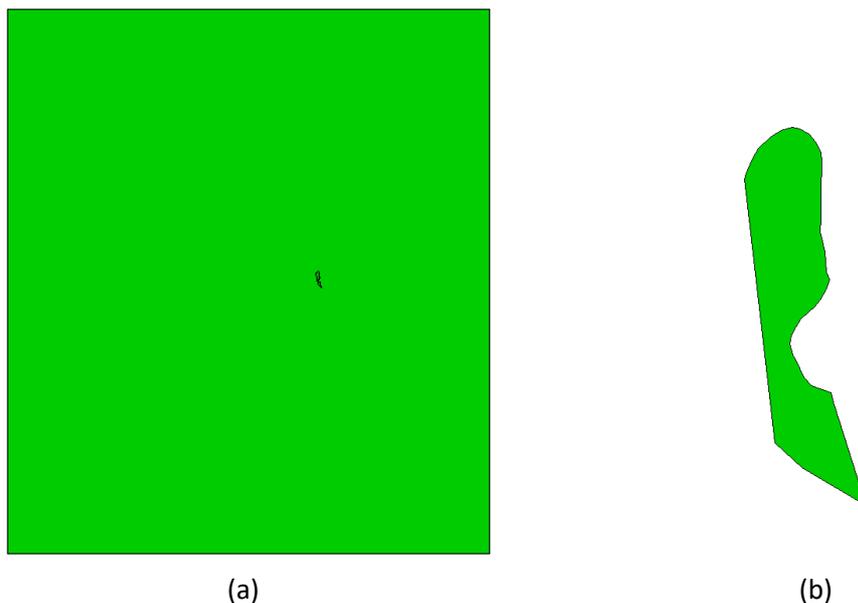
Figure 3.8: One-degree by one-degree solar radiation data before crop (a) and after cropping (b).

Figure 3.8 (a) shows an example of the NASA insolation data for a one-degree latitude by one-degree longitude area with the polygon representing a mineral rights area plotted on top of it, while Figure 3.8 (b) shows the result of the cropping. Note that the mineral rights area in (a) is barely visible at the scale of the original data. The colour of the plot is coloured green to better illustrate the cropping action, but in reality, the data is merely a polygon.

3.6.7   Cropping Topographical Data per Mine

The process of cropping introduced an interesting nuance that illustrated how, despite the best effort, it is nearly impossible to think of all scenarios when attempting something for the first time. In this case, it was brought about by the characteristics of the Northern Cape landscape. In this area, it is not very uncommon to find vast expanses of land that has no, or very little, elevation change. Combine this with some of the defined mining areas being very small and it can easily happen that an entire mine is not crossed by a single topographical line. Figure 3.9 shows an example of the plot of a topographical dataset and the polygon representing a small mine on the same plot. Being located in the area between two topographical lines means that when a crop action is performed, the result yields an empty set.

This causes problems later on when calculating the aspect and slope of an area and the input values are assumed to be non-empty.
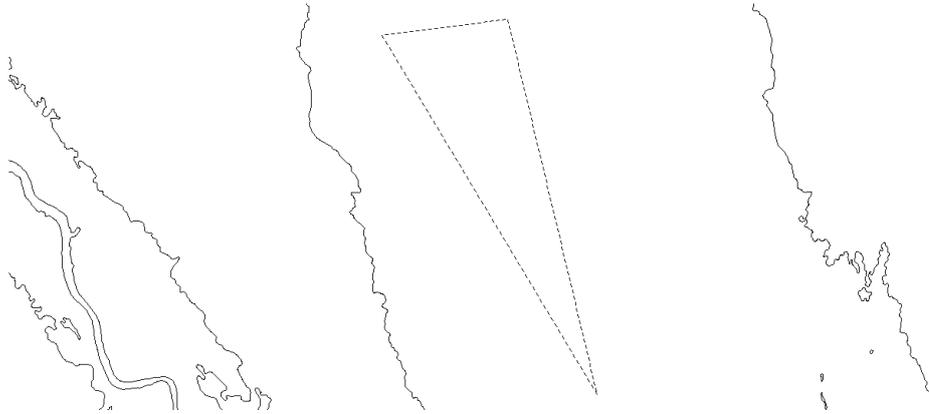
Figure 3.9: Plot of an area with mineral rights (dashed) between two topographical lines whilst never touching.

The solution could thus have been implemented on either side. Either at this stage, while dealing with the cropping of topographical data or later while calculating the aspect and slope of each area. In this case, it was chosen to go with the former option. While it would have been relatively easy to deal with a list of exceptions while calculating the aspect and slope, this calculation in itself is already a complex piece of R script and adding more complexity was not deemed optimal. Dealing with the problem as an exception would also lead to it being dealt with as an exception in any downstream calculations. The solution implemented relied on the fact that the absolute elevation value is not of concern to any of the objectives, merely the change in elevation. For this reason, whenever the above case was encountered, a dummy topographical line would be created with a polyline following the exact shape of the area under investigation. A dummy value of 1000 was then assigned to it as its elevation across the entire area.

The pseudo-code that deals with both cases, the standard case and the exceptions, are given below. The standard case was calculated in much the same fashion as described in the two previous chapters. After performing those calculations, a tedious manual process was used to search for all the exceptions. This involved plotting the output topographical data for each area and visually inspecting it. A list was then kept of all the special cases. Only these were then considered when performing calculations to create the dummy data for the exceptions.

1. Read the mineral rights data into a variable using the *readOGR()* function contained in the *rgdal()* library.

2. Create a list of all the topographical data in the filtered directory.
3. Start a *for* loop at the beginning of the list created in step 2 and continuing to the end.
    a. Read the ith topographical shapefile into a variable using the *readOGR()* function contained in the *rgdal()* library.
    b. Start a *for* loop at the first *GID* of the shapefile from step 1 and continuing to the last.
        i. Use the *gCrosses()*, *gIntersects()* and *gOverlaps()* functions from the *rgeos()* library to check for the many cases in which a mining area can co-exist with the topographical data.
        ii. If any of the outputs of step i produces a TRUE value, then use the *crop()* function from the *raster()* library to crop the topographical data with the outline of the mining area.
            1. Change the current working directory to the corresponding *GID* and save the cropped shape as a new shapefile using the *writeOGR()* function from the *rgdal()* package.
4. Write any errors and warnings in a log file.

Figure 3.10 (a) shows an example plot of a mining area and the topographical data on top of each other. The result of the cropping computations can be seen in Figure 3.10 (b). It is important to note that the outline of the

<div align="center">(a)   (b)</div>
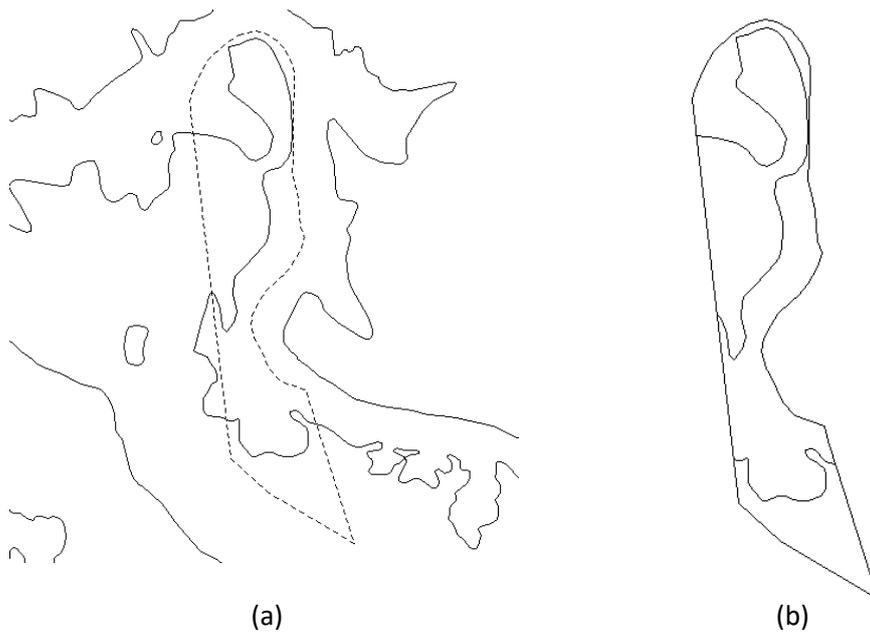
Figure 3.10: Topographical data with dashed outline of mineral rights area (a) and the product of cropping the topographical data with the outline of the mineral rights area (b).

As was the case in the previous chapter, here a choice was also made to use one function above another with similar functionality. The *intersect()* function from the *raster()* library would have equally performed the same if used to perform the crop action in step ii, except for the class of the output. With both inputs being of the class *SpatialPolygons*, it is natural that the output is of the same class. However, to save the output using the *writeOGR()* function from the *rgdal()* library requires the data to be of the class *SpatialPolygonsDataframe*. Fortunately, the *crop()* function of the *raster()* library performs the same computation, but with an output of the correct class. This was not a problem in the previous chapter when cropping the solar insolation data with the mining areas, because both the inputs were of the same class, *SpatialPolygonsDataframe*, and therefore the output was of the same class. The topographical data cropped in this chapter is of class *SpatialLinesDataframe* and therefore the output class is the nearest common denominator between the two input classes.

Once the exception areas were identified, dealing with them was slightly easier than the standard case. It is also definitely a computationally less expensive operation and the script created to perform this executed in minutes. The shapefile containing the mining areas originally had many variables, many of which are of no consequence to the computational objectives in this paper. The easiest path was thus to purely rename one of the existing variables to mimic the *HEIGHT* variable found in the topographical data and fill this variable with the dummy value.

1. Create a variable list with all the *GIDs* of the exception cases.
2. Read the mineral rights data into a variable using the *readOGR()* function contained in the *rgdal()* library.
3. Start a *for* loop at the beginning of the list created in step 2 and continuing to the end.
   a. If the ith *GID* is in the list created in step 1 then set the current working directory to the sub-directory matching with the current *GID*.
      i. Copy all the data from the mineral rights data relevant to the current *GID* to a temporary variable.
      ii. Change the name of one of the variables in the temporary variable from step i to *HEIGHT*.
      iii. Assign a value of 1000 to this variable.
      iv. Write the temporary variable as a new shapefile using the *writeOGR()* function from the *rgdal()* package with an appropriate name.

Figure 3.11 (a) shows an example of a plot of a mining area which looks remarkably similar to the plot (b) of the topographical lines for this area. The main difference being that the polygon in (b) has an extra attribute containing the dummy height for the area. In this case, different from the normal case shown in Figure 3.10, the outline of the area is also the cropped topographical data manufactured to replace the lack of topographical data inside the area.

(a)   (b)
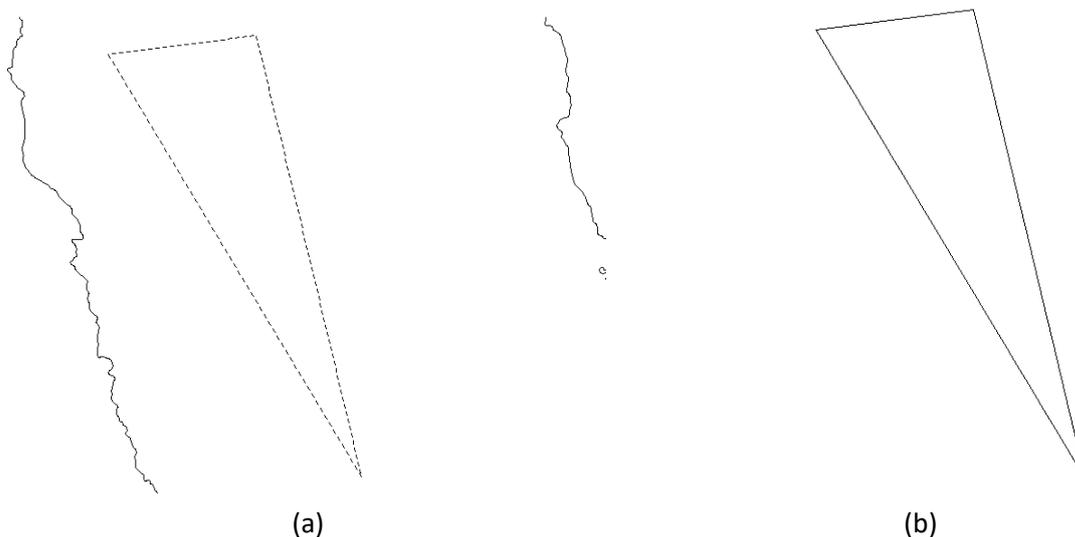
Figure 3.11: Topographical data with dashed outline of mineral rights area between two topographical lines (a) and the product of cropping the topographical data with the outline of the mineral rights area (b).

3.6.8   Calculating Brut Solar Energy per Mine

Up until now, projections and coordinate reference systems have not been brought into the equation when discussing the calculations done on the different datasets to filter them. All the calculations performed until now merely required a projection to be present but wasn't a requirement in any of the calculations. When calculating the brut solar energy per mine, this becomes relevant because it is dependent on the land area of each mine. This was discussed in Chapter 3.1 and a decision on a suitable projection for calculating land area was made.

The basic preparation for this calculation was done in chapter 3.7.6, where the NASA solar insolation was cropped to the outline of each mining area. The structure of the required script is therefore made less complex and the only remaining complexity is how to deal with cases where a mining area would straddle an integer longitude or latitude. Luckily the solution to this problem is made easier by the choice of how the refined data is stored, as discussed in chapter 3.7.4. During the cropping process, the result was stored in a sub-directory with any other results for the same area and only that area. This implies that the straddling problem would be resolved by merely scanning the sub-directory containing the solar insolation data of a specific *GID* for multiple shapefiles before continuing to perform the calculations.
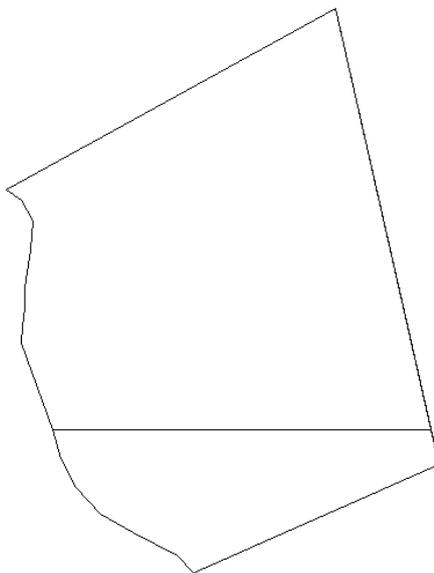
Figure 3.12: The result of cropping solar radiation data with the mineral rights area. The mineral rights area corresponding to GID 1710356 straddles two NASA solar radiation cells, with the horizontal line section indicating where the two polygons touch.

Figure 3.12 shows an example of such a case. Here it can be seen that during the process of cropping, two separate shapefiles were stored whose outside perimeter together equal the outline of the area investigated. The straight

horizontal line is the edge of the data from the original NASA insolation data where it changed from one integer longitude to the next.

The only other considerations are to change the projection of the shapefiles to the appropriate equal-area projection and also then in what form to store the results. Concerning the projection, it is important to note that merely changing the projection is not sufficient but transforming from one projection to another is what is required. To achieve this, the *spTransform()* function from the *sp()* library was used. For the storage of the results, it was decided to use a *dataframe*, which is a class of variable internal to R but is easily written to disk as a *comma-separated values* (CSV) file. This would make it easier to access and manipulate the results outside of R. The pseudo-code for all the operations discussed above follows here.

1. Create an index file from the *GIDs* in the mineral rights shapefile.
2. Create an empty *dataframe* with the base R function, *data.frame()*.
3. Start a *for* loop traversing the index created in step 1.
   a. Scan the sub-directory corresponding to the *GID* of the ith index for shapefiles and create an index of these files.
   b. Start a *for* loop traversing the index created in step a.
      i. Store the shapefile referenced by the jth index in a temporary variable by reading it with the *readOGR()* function from the *rgdal()* library.
      ii. Transform the shapefile to an equal area projection as described in the previous paragraph.
      iii. Calculate the area of the shape from step i using the gArea() function from the *rgeos()* library.
      iv. Calculate the kWh of insolation received per day.
      v. Prepare a new *dataframe* entry that includes *GID*, Area, Insolation and kWh per day and adds this to the global *dataframe*.
4. Sort the final *dataframe* using the *arrange()* function from the *dplyr()* library.
5. Write the sorted *dataframe* to disk as a CSV file using the *write.csv()* function from the *utils()* library.

Calculating the area of a polygon seems like a basic function, but even here there is an important nuance that requires knowledge of the inner workings of the function and library selected to perform the task. As explained in chapter 3.1, shapefiles have the ability to define and store holes. The question then becomes what happens to these holes when the area is

calculated for the polygon that contains them. In the context of areas with mineral rights, it was assumed that a hole in a polygon indicates an area which does not have the permitted mineral rights as the area containing it, for whatever reason. Based on this assumption, it was convenient to use the *gArea()* function from the *rgeos()* library, rather than using the *areaPolygon()* function from the *geosphere()* library, as it automatically calculates and subtracts the area of the holes, which *areaPolygon()* does not do.

Calculating the kWh per day for the area is a simple multiplication equation. The value given in the NASA solar insolation shapefile for each area is defined as the annual average kWh per square meter per day at that point. This average is based on satellite-derived data over a 20 year period. Therefore, multiplying this value with the total square meterage value for a region yields the total kWh per day. Table 3.4 shows the daily kWh data, along with the other columns stored in the CSV for the first 10 *GIDs*. The full table is available as an addendum to this document.

Table 3.5: Sample of CSV output file for first ten brut insolation entries

	Area GID	Area Size [m^2]	Insolation [kWh/m^2/day]	kWh/day
1	1710351	1566856	5,77	9040759
2	1710352	1099000	5,84	6418159
3	1710354	20667666	5,83	120492491
4	1710355	11139893	5,83	64945577
5	1710356	2882098	5,83	16802633
6	1710356	10280358	5,83	59934488
7	1710357	19054264	5,83	111086359
8	1710358	12126568	5,83	70697894
9	1710360	30551776	5,77	176283746
10	1710372	39844285	5,74	228706197

3.7 Exclusions

Chapter 3.7 concerned itself with finding the most optimistic value of solar insolation per area, while this chapter sets out to carve away at that value as much as possible within the limits of what can be justified. This can only be done where the required information is available or the necessary data is available to manipulate into the information that is required.

3.7.1   Converting Topographical Data into Aspect and Slope Data

A common element discovered in the literature study was the use of ground slopes as a qualifier to exclude certain areas from being included in the set of areas suitable for solar development. This creates the need for datasets that include data for both aspect and slope. While it cannot be said with absolute certainty that datasets which include these two types are very rare, this type of dataset was never encountered during the broader search for data for this research. Luckily aspect and slope can be calculated from topographical data, which are readily available. It does, however, brings with it a lot of complexity and the results obtained can vary based on the decisions that are made during the process of computing it.

The R library *raster()*, allows for the calculation of aspect and slope with relative ease using the *terrain()* function, however, it comes with a caveat. It requires the input to be in the form of a Digital Elevation Model (DEM) with all cells populated with values. A DEM is merely a raster object with each cell containing a single value; height. In contrast, topographical data consists of lines connecting points of the same height. It can be argued that topographical maps are optimized to be viewed and interpreted by humans and therefore a lower resolution might be favoured because it would appear less congested. The drawback is that the exact values of areas between two topographical lines cannot be obtained with certainty, but it can only be said to be anywhere between the respective values of each line and not exceeding them.

Interpolation provides a way of assigning values to these areas that exist between two topographical lines. In general, nearby data is selected with a weighted average to calculate the missing values at a given location. According to Babak & Deutsch (2008), there are two main branches of interpolation. Either a statistical criterion can be used to select the nearby weighted values or a simpler deterministic method can be used. For the former, techniques such as kriging (including but not limited to: simple-, ordinary-, universal- and simple cokriging) can be implemented, while for the latter, inverse distance weighted (IDW) interpolation is the most commonly implemented technique. IDW especially takes advantage of the principle of spatial autocorrelation, which assumes that values closer to the missing value should be more similar to it than values which are far away (Setianto & Triandini 2013). Simply stated, when taking one step away from your current position on earth, you never really find yourself in a very different state (with the exception of stepping over a cliff). The implementation of this is based on assigning a higher weight to points which

are closer (Babak & Deutsch 2008) or as the name suggests, using the inverse distance. IDW is implemented in R using the combination of the *gstat* function from the library with the same name and the *interpolate* function from the *raster* library. The most critical parameter to set here is the inverse distance power (*idp*), which controls how tightly around the predicted point the weighted values are chosen. The most common value that is usually assigned to the *idp* is usually two (Babak & Deutsch 2008) and this was the value which was implemented to compute each DEM.

Calculating the DEMs were the most time expensive calculation of this paper by quite a margin. The script written to perform these tasks ran for numerous days uninterrupted. Compared to this, the calculation of the aspect and slope data was relatively simple. The *raster* package, which was relied on heavily until now, was once again utilised. The function *terrain* calculated both the required values at once. The function requires the user to choose between two techniques, namely Flemming & Hoffer or Horn. A thorough comparison of these two techniques was made by Jones (1998) which found that Flemming & Hoffer performed better for smooth surfaces and Horn performed better for noisy surfaces. Jones (1998) warns that when topographical lines are digitised and converted to a DEM through interpolation, that it might create step artefacts. As this is exactly what was done, it was assumed that the "noisy" option would be better and thus Horn was selected.

Preceding all these steps, the last aspect under discussion is how to prepare the topographical shapefiles in order to apply the interpolation and the subsequent slope and aspect calculations. This step requires the polylines within the shapefiles to be converted into points. The ability of R to coerce data makes this step relatively easy and can be performed with only a few lines of code in a script. The only complexity here is introduced with the initial cropping of the topographical data, due to the issue discussed in chapter 3.7.7 regarding the mining areas that fall completely within two topographical lines. In that chapter, it was explained that dummy topographical data was created for these exceptions. The class of the dummy data and that of the normal cropped topographical data differs slightly and requires one intermediary coercion to be performed. This is solved with a simple *if* statement and vindicates the decision taken to create dummy data. This implies that from this point forward, the exception areas will be indistinguishable from the normal areas and no more exceptions would have to be made in downstream calculations. The pseudo-code that describes the script which performs all the tasks discussed here follows below.

1. Create an index file from the *GIDs* in the mineral rights shapefile.

2. Start a *for* loop that traverses the index created in step 1.

    a. Set the current directory to the one that corresponds with the ith *GID.*

    b. Create an index file of all the topographical shapefiles.

    c. If the list created in step b is not empty, then continue to the next step.

    d. Start a *for* loop that traverses the index created in step b.

        i. Read the topographical shapefile using the readOGR() function from the rgdal() library.

        ii. Use the boundary box of the current topographical file to specify the dimensions of an empty raster.

        iii. Set the resolution of the raster. Note that the resolution is set using the same units as the bounding box i.e. degrees.

        iv. If the class of the topographical data is *SpatialPolygonsDataframe* (that corresponds to the exception areas) first coerce to *SpatialLinesDataFrame* before coercing to *SpatialPointsDataFrame.*

        v. If the class of the topographical data is *SpatialLinesDataframe* coerce directly to *SpatialPointsDataFrame.*

        vi. Create an interpolation model using the *gstat()* library.

        vii. Create a DEM using the *interpolate*() function from the *raster()* library combined with the empty raster from steps ii & iii and the interpolation model from step vi.

        viii. Use the DEM from step vii as the input to the *terrain()* function from the *raster()* library in order to calculate aspect and slope.

        ix. Save the DEM to disk using the *writeRaster()* function from the *raster()* library.

        x. Save the aspect and slope data to disk using the *writeRaster()* function from the *raster()* library.

Figure 3.13 shows the intermediate steps toward calculating the aspect and slope data. In (a) and (b) the original cropped topographical data is shown next to the same data that has been converted from lines to points. Below that (c) shows what the result of the interpolation process looks like. The

(a)

(b)

(c)

(d)

(e)
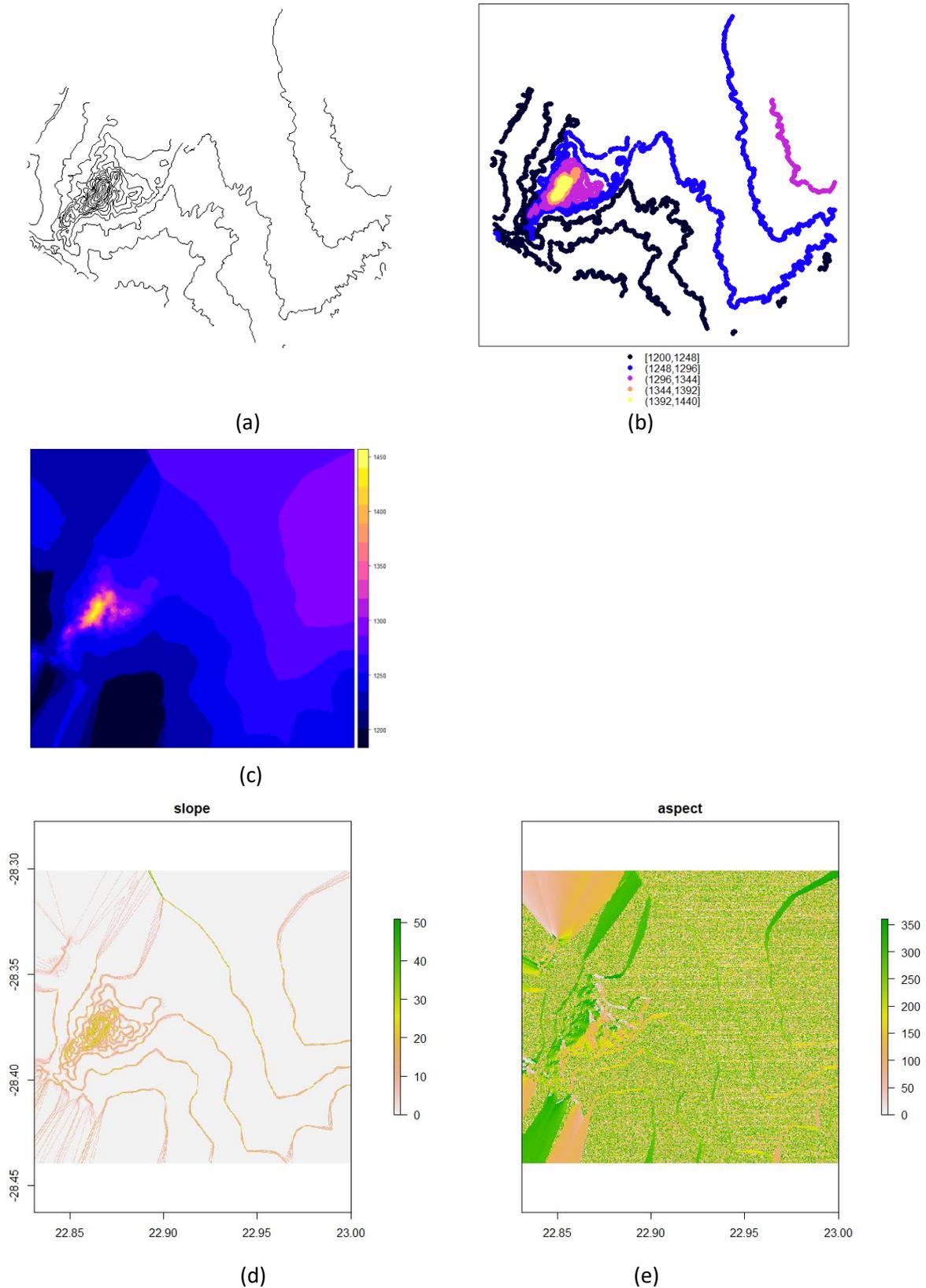
Figure 3.13: Original topographical data (a), converted to point, or raster, data (b), result of interpolation (c) and the slope and aspect data in (d) and (e) respectively.

aspect and slope plots are shown in (d) and (e) respectively. Note how the

values for aspect are seemingly random in areas that are not close to the original topographical lines. This will be discussed further in more detail in chapter 3.7.3 when the data is used to perform a selection of exclusion areas.

3.7.2 Finding and Selecting Other Qualifying Exclusion Areas

Not all exclusions are as well defined as those that depend on aspect and slope. There is a whole barrage of exceptions, some whom may only apply once in the entire dataset. These tend to be based loosely on what is known to be acceptable truths. It is, for instance, not advisable to build solar infrastructure in a large body of water[2] or on a public road. Some of these types of exclusion even have datasets that were received with the data received from NGI, such as National Roads and high voltage power lines. Problematically, these datasets are very sporadically populated. Some areas will have the power lines included, while in other areas there are no power lines included in places where the aerial photos clearly show power lines. This lead to the decision to manually identify these exclusions. Further motivation for this decision is based on the fact that there would inevitably be some exclusions, such as open groove mines, that will have no data anywhere and would have to be identified manually. A system would have to be developed and coded to be able to do this and once such a system exists, it might as well be used to identify the exclusions that are defined sporadically. At the very least it adds to the overall project by bringing some consistency.

The overall strategy that was developed was based on human input. This required drawing polygons around objects that could be positively identified and were part of a pre-determined list of exclusions. Being based on human input, this removed automation as an option, but this did not mean that an R script could not be created to make it as easy as possible. For this, it required to understand how R interacts with the operating system and in particular mouse clicks. For this purpose the *drawPoly()* function from the *raster()* library was implemented. It uses the current plot area in R as input combined with user-defined mouse clicks. The coordinates of the clicks on the screen are translated to the coordinates as they exist on the raster image. This allows the function to output a polygon which is

---

[2] Retrospectively, right after completing all calculations, it was announced in the media that in China the world's largest floating PV plant was built (Lant 2017).

already of the class *SpatialPolygons*, which means it can readily be plotted with any of the other datasets for each area.

It was foreseen that creating the polygon exclusions for the entire dataset would not be done in a single sitting. Therefore, it was decided to build in the functionality to start, stop and resume the process at any given point. This was achieved by prompting the user after creating the polygons for a specific region and then using the feedback to either store the current process or continue. When it was chosen to store the current process, the index of one of the internal *for* loops is stored in a file on disk. This file is automatically opened and queried when the script is started again and the *for* loop limits are subsequently changed to start at the stored index and not from the beginning. This and the rest of the R script is described in the following pseudo-code.

1. Store a list of all the *GID*s excluded from the overall calculation due to being contained within other regions already included.
2. Open the file on disk containing the saved progress index.
3. Create an index variable from the *GIDs* in the mineral rights shapefile.
4. Start a *for* loop at the point recovered in step 2 until the end of the list of the index created in step 3.
   a. Create an index variable of all the cropped aerial photos stored in the directory corresponding with the current *GID*.
   b. If the index created in step a is not empty and the current *GID* is not in the list created in step 1, then continue to step c.
   c. Load the first aerial photo from the list created in step a using the *stack()* function from the *raster()* library.
   d. Load the shape of the mining area that corresponds with the current *GID*.
   e. Use the CRS of the aerial photo loaded in step c to transform the projection of the shapefile loaded in step d to the same projection.
   f. Plot the current mining area to utilise the full extent of the plot window in R.
   g. Start a *for* loop that traverses the index created in step a.
      i. Load each aerial photo from the list created in step a using the *stack()* function from the *raster()* library.
      ii. Plot the current aerial photo with the other aerial photos.

    h. Start a *while* loop that continues looping if the user indicates that another polygon must be created.

        i. Create a new polygon using the *drawPoly()* function from the *raster()* library.

        ii. Assign a CRS to the polygon.

        iii. Embed the polygon in a *SpatialPolygonsDataFrame*.

        iv. Write the result of step iii as a shapefile to disk using the *writeOGR()* function from the *rgdal()* library.

        v. Prompt the user to continue to the next polygon or not with the use of the *readline()* function from base R.

    i. Prompt the user to continue to the next mining area or not with the use of the *readline()* function from base R.

    j. If the input from step i indicates that the user does not want to continue, continue with step k.

    k. Save the current index of the *for* loop from step 4 to disk.

    l. Call a *break()* function to interrupt the current loop.

To identify potential exclusions, it was decided to use a system where positive identification was required to include an exclusion. This meant that possible exclusions were not marked with polygons if it was unclear what its identity is. To be positively identified, it should be clearly visible and identifiable as one of the following:

- Rivers (not including dry riverbeds)
- Areas where the defined mining area included a portion of the ocean along a coastline.
- Farm sheds
- Farm housing
- Mining housing
- Mining machine sheds
- Open grooved mines
- Mine earth dumps (if discernible)
- National and regional tarred roads
- Airfields
- Railway tracks

Mine earth dumps posed a considerable challenge as these become very difficult to identify, especially if native plants have established themselves on the dump and if the photo angle is very close to perpendicular. The latter makes it more difficult to see the vertical drop at the edge of the dump. On

the other side of the spectrum are items that can also be positively identified but were deliberately not included as exclusions. These are:

- Gravel/sand roads
- Agricultural activity

For the purpose of this investigation, gravel and sand roads were seen as non-permanent structures that can be rerouted or removed without a significant cost compared to the overall cost of development. Much the same was attributed to visible farming activities. Since being located within the borders of an area defined as having mineral rights, it was assumed that the agricultural fields were of a temporary nature. Also, as agriculture doesn't alter or change the landscape dramatically, it would not be something that poses a great limitation to a developer. In fact, the opposite might be true.

Figure 3.14: GID 97373 outline in black and the identified exclusion areas in red.

71

The result of this process yielded 485 positively identified exclusions which took approximately two weeks to identify. Figure 3.14 shows an example of the identified exclusions for one specific mining area.

3.7.3 Relating Exclusion Areas to Solar Insolation

At this point, a lot of effort has gone into identifying exclusion areas, but these only serve as an input to the next level of calculations. This chapter deals with correlating each exclusion area to the corresponding insolation value found at that area and then calculating the total insolation loss. The first part of the chapter will deal with the exclusion polygons that were created, as discussed in chapter 3.8.2. The second half of this chapter will then discuss how the aspect and slope data that was calculated and discussed in chapter 3.8.1 will be used to select exclusion areas and calculate their corresponding insolation values.

Because the NASA insolation data spans one-degree latitude by one-degree longitude, it is very unlikely that a mining area will have more than one insolation value across its land area. Even so, this means that there are two basic cases that can occur and must be dealt with in the R script. The first, and easiest case, is when a mining area is only covered by a single insolation value. The second case is when there is any number of insolation values greater than one within a single mining area. Dealing with these cases were greatly simplified with the data structure chosen and discussed in chapter 3.7.4 and the method of cropping the insolation data as discussed in chapter 3.7.6. Even with these simplifications, it still required 120 lines of code to complete the task. Each mining areas has within its own sub-directory structure a set of shapefiles that store the desired exclusions, as well as a set of shapefiles that store the insolation values for the entire mining area, cropped to the shape of the mining area. Importantly also, these insolation shapefiles do not overlap each other. This means that dealing with the second, more difficult case, would require making a list of the insolation shapefiles, which would be more than one, and then comparing the exclusion shapefiles to it one by one to establish whether it is contained within that insolation area or not. If it is, then the insolation value associated with that exclusion area has been found, and so on. The pseudo-code dealing with both these cases are given below.

1. Store a list of all the *GID*s excluded from the overall calculation due to being contained within other regions already included.
2. Create an index variable from the *GIDs* in the mineral rights shapefile.

3. Create an empty *dataframe* with the base R function, *data.frame()* to store the results in.
4. Start a *for* loop traversing the index created in step 2.
   a. If the current *GID* is not in the list created in step 1, continue to step b.
   b. Create a list of all the exclusion shapefiles in the sub-directory of the parent directory corresponding to the current *GID*.
   c. Create a list of all the insolation shapefiles in the sub-directory of the parent directory corresponding to the current *GID*.
   d. If the length of the list created in step c is more than one, then continue to step e (this is the difficult case), otherwise, continue to step f (this is the easy case).
   e. Start a *for* loop that traverses the list created in step c.
      i. Open the ith insolation shapefile using the *readOGR()* function from the *rgdal()* library.
      ii. Transform the shapefile projection to an equal area projection using the *spTransform()* function from the *rgdal()* library.
      iii. Start a *for* loop that traverses the list created in step b.
         1. Open the jth exclusion shapefile using the *readOGR()* function from the *rgdal()* library.
         2. Transform the shapefile projection to an equal area projection using the *spTransform()* function from the *rgdal()* library.
         3. Use the *gContains()* function from the *rgeos()* library to establish whether the ith insolation polygon and the jth exclusion polygon overlap.
         4. If step 3 resulted in a TRUE expression, calculate the land area of the exclusion polygon with the *gArea()* function from the *rgeos()* library and continue to step 5.
         5. Prepare a new *dataframe* entry that includes *GID*, ExclusionID, Area, Insolation and kWh per day and adds this to the global *dataframe*.

73

f. Perform step i and step ii and then continue to step g.

g. Calculate the land area of the exclusion polygon with the *gArea()* function from the *rgeos()* library.

h. Prepare a new *dataframe* entry that includes *GID*, ExclusionID, Area, Insolation and kWh per day and adds this to the global *dataframe*.

5. Sort the final *dataframe* using the *arrange()* function from the *dplyr()* library.

6. Write the sorted *dataframe* to disk as a CSV file using the *write.csv()* function from the *utils()* library.

The result of the above pseudo-code is a CSV file with the same columns as the headers specified in step h. This is similar to the result of the CSV that was created for gross solar energy as discussed in chapter 3.7.8 and means the process has now finally reached the point where final numbers can be calculated. Table 3.5 shows a sample consisting of the first 10 entries of how the data found in the CSV file looks and the similarity with Table 3.4 should also be apparent. The "Exclusion ID" column is the actual file name stored on disk where the convention was chosen to place the GID at the beginning of the file name and suffix it with the number of the exclusion as they were defined in order to have multiple exclusions per area.

Table 3.6: Sample of CSV output file for first ten exclusion entries

	Area GID	Exclusion ID	Area Size [m²]	Insolation [kWh/m²/day]	kWh/day
1	1710351	1710351_Exclusion_1.shp	5887	5,77	33969
2	1710360	1710360_Exclusion_1.shp	1838514	5,77	10608226
3	1710372	1710372_Exclusion_3.shp	775905	5,74	4453697
4	1710372	1710372_Exclusion_1.shp	50719	5,77	292646
5	1710372	1710372_Exclusion_2.shp	158901	5,77	916862
6	1710372	1710372_Exclusion_3.shp	775905	5,77	4476974
7	1710372	1710372_Exclusion_4.shp	228699	5,77	1319594
8	1710390	1710390_Exclusion_3_1.shp	10045	5,77	57959
9	1710392	1710392_Exclusion_1.shp	142245	5,77	820751
10	1710394	1710394_Exclusion_1.shp	5554	5,77	32045

As mentioned at the beginning of this chapter, the next few paragraphs will deal with the aspect and slope data and how these relate to identified exclusion areas. The parameters that steer the selection process has been discussed in chapter 2.5, but as a reminder, the basic principle is to eliminate

all areas with a southernly directed slope above a certain degree threshold as well as areas with any direction of the slope that slope too much. The slope and aspect data are stored in separate files, thus making selections based on these would require having direct access to both in an identical reference framework or at least one where there is a one to one relationship between the data in both. The *stack()* function from the *raster()* library makes this task much easier by being able to load both datasets as two layers of the same raster. Thereafter it is merely a process of selecting the data that must be retained and discarding the rest, after which the remaining calculations can be done. This process is described in the pseudo-code below.

1. Store a list of all the *GID*s excluded from the overall calculation due to being contained within other regions already included.
2. Create an index variable from the *GIDs* in the mineral rights shapefile.
3. Create an empty *dataframe* with the base R function, *data.frame()* to store the results in.
4. Start a *for* loop traversing the index created in step 2.
   a. If the current *GID* is not in the list created in step 1, continue to step b.
   b. Set the current working directory to the sub-directory corresponding to where the aspect and slope data is stored for the ith *GID* in the *for* loop.
   c. Load both the aspect and slope datasets using the *stack()* function from the *raster()* library.
   d. Use the *mask()* function from the *raster() library* to mask the points in the raster that do not comply with the selection criteria.
   e. The *area()* function from the *raster()* library is used to calculated areas of the unmasked data. Important to note is that this function does not compute the area of the entire raster, but rather calculates the area which each cell represents. This leads to an extra step, step f.
   f. Calculated the sum of all the cells in the raster calculated in step e using the base R function *sum()*.
   g. Prepare a new *dataframe* entry that includes *GID* and Area and adds this to the global *dataframe*.
5. Sort the final *dataframe* using the *arrange()* function from the *dplyr()* library.

6. Write the sorted *dataframe* to disk as a CSV file using the *write.csv()* function from the *utils()* library.

Another important note about step e relates to area calculations and projections. The documentation for the function is quick to point out that the answer is an approximation because the width of the cells decreases in length at the poleward side compared to the equator-ward side. This problem is negated by the action taken when these datasets were created, as described in section 3.6 earlier in the chapter.

While performing this action, the insolation values and the resulting kWh per day for each exclusion was not included in the CSV file that was created in the end. This is because the average insolation value for each region was already extracted and saved to a CSV file as part of the process that was followed in chapter 3.8.2 and therefore that dataset can merely be re-used. Figure 3.15 (a) and (b) shows the original datasets containing the aspect and slope data for an example area. Below that, Figure 3.15 (c) and (d) shows what the result of the selection criteria in step d is. By inspection, it can be seen that these dots correspond to the areas in (a) and (b) that intuitively seemed to have large slopes.

(a)

(b)
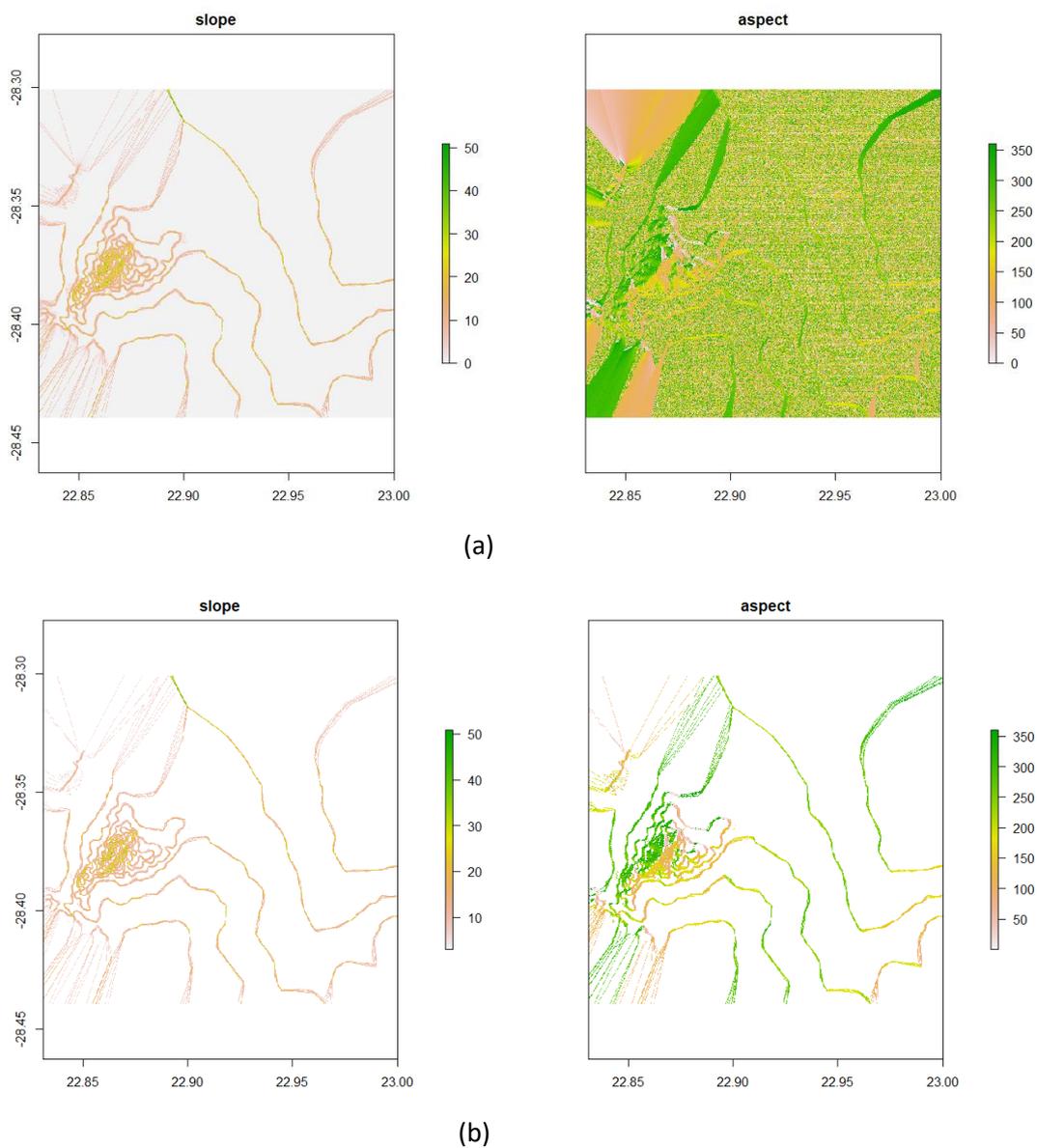
Figure 3.15: Original slope and aspect dataset (a) and the subset (b) that was selected for exclusion based on the slope selection criteria.

The exclusion criteria were selected based on Table 2.1 compiled during the literature study. Many of the authors themselves noted that, unlike for CSP, there is no accepted standard to use for slope exclusions. This brings into question the entire premise of the decision-making methods used and whether an area classified as "less preferential" is truly less preferential than one classified as "preferred," but this is beyond the scope of this paper. Despite this, a decision has to be made regarding this. Many of the authors chose a blanket cut-off and five per cent was a very prevalent choice exercised.

There is a problem with this approach and that is the influence of latitude. Solar panels tilted towards the sun generally perform better than panels lying horizontally on a flat surface. A general rule of thumb is that fixed-tilt solar panels should be tilted with a number of degrees that approach the latitude at which it is installed, in order to better face the sun. This crude rule compares well to the findings of Le Roux (2016) studying the optimal tilt angle for local South African conditions, which varied between 26° to 29° depending on where in the country. This is also the premise on which LTI solar maps are created. The practical outcome of this is that a solar panel installed flat on a slight North aspect incline of six degrees at a position with latitude six degrees South, is favoured above a panel installed flat at the same latitude. However, with the slope cut-off favoured in the literature study, this area would have been dismissed from the "preferred" category.

Another fact to be aware of is the tendency to work with a slope as a percentage. This is common in the architecture and civil roadworks industries, but in GIS mapping as encountered within R the tendency is more toward units in degrees. Therefore, the decision here was first taken by considering the way it was found in the literature study, then converted to degrees for implementation. A cut-off of three per cent was chosen, while for slopes with a Northern aspect (being defined as from 315° to 45°) the limit is increased to five per cent. The conversion is explained with the help of Figure 3.16 below. As can be seen in (a) the per cent-slope is a ratio of the rise of a slope compared to the distance travelled while rising, expressed as a percentage. In comparison, the degree-slope (b) is expressed as the angle of a triangle, related to the same two components as for the percentage slope, given by the tangent of the degrees. Rearranging these equations and setting them equal, it can be easily shown that converting from percentage-slop to degree slope requires the formula

$$\emptyset = \tan^{-1}\left(\frac{slope(\%)}{100}\right)$$

which was used to set the two limits discussed above as 1.72° and 2.86°, respectively. These are the limits that were implemented in the code found in Addendum A and the result of which was shown in Figure 3.15.

$$Slope(\%) = \left(\frac{b}{a}\right) * 100 \qquad\qquad tan(ø) = \left(\frac{b}{a}\right)$$
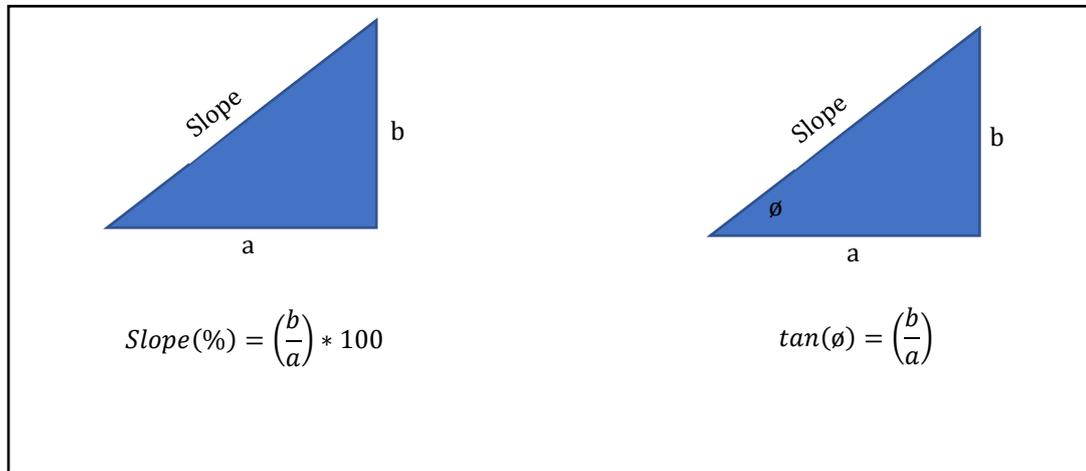
Figure 3.16: Comparison of percent-slope and degree-slope

Chapter 4

4.   Simulation Results

The following sections will purely show the results obtained after all the calculations were completed. This is essentially a single number per *GID* that indicates the amount of MWh of solar insolation received per day and summed to calculate the totals. This is chosen as the starting point, as it represents the latest point in the process that provides a meaningful number to consider before any further assumptions or interpretations are added.

The first thing to note is the number of *GIDs* in the results. At the starting point, there were 147 regions in the shapefile containing the areas with valid mineral rights, as received from the NGI. After compensating for data errors and the unique interpretation cases found in Table 3.3, there were 113 valid cases left. A summary of the results is given in Table 4.1 and the full table of results for each individual area is given in Addendum B. The final numbers are given in GWh instead of kWh to make the numbers a bit more palatable. The total of the area under investigation was calculated as being 2 572km^2, which represents 0.69% of the total land area of the Northern Cape province.

Table 4.1: Summary of calculations

Number of valid GID areas	113
Total area of valid GID areas [km^2]	2 572
Brut insolation [GWh/day]	14 882
Insolation excluded from general list of exclusions [GWh/day]	1 227
Insolation excluded due to slope and aspect [GWh/day]	1 598
Nett insolation [GWh/day]	**12 057**

The final result, and the pinnacle towards where this paper strived is the sum of the nett solar insolation for the areas which were investigated, with this number sitting at 12 057GWh/day. This is what remained of the brut 14 882GWh/day that was calculated to fall on these areas from the raw NASA data before any subsequent calculations were made. What chipped away at this number was the totals of the two main types of exclusions that were dealt with. The first type is the combined exclusions identified in section 3.8.2 and defined by 485 manually drawn polygons,

which amongst others, included water bodies, mining and farming structures, open-grooved mines as well as road and rail networks. The total combined number for these exclusions were calculated as a loss of 1 227GWh/day. The second type of exclusion was all land sloped more than 3°, except all North facing slopes, which were excluded when more than 5° in slope. The total exclusion due to this was calculated as 1 598GWh/day.

Sections 4.2 and 4.3 will delve a bit deeper into these single total number which, on their own, does not provide much insight other than revealing what a small percentage of the total land area is actually required to make a significant difference to the electricity production source profile of South Africa.

4.1   Data and Result Validation

The most prominent data sources in this thesis are:
- Aerial photographs
- Topographical shapefiles
- Solar insolation shapefile
- Shapefile containing the mineral rights assigned to areas in the Northern Cape.

Each of these requires a different method of validation. The easiest form of validation would be if another data source could be found and used in a comparison. This was the basis of testing the aerial photographs. Google Earth Pro (Google LLC 2018) provided an easy and readily available source of many curated aerial photographs. The first step was to plot a sample of an aerial photograph in R and then extracting a set of coordinates using the *locator()* function from the *graphics* package in the base R (R Core Team 2015) repository. This was then converted to a spatial object using *SpatialPoints()* from the *sp* (Pebesma & Bivand 2005) package and plotted on the aerial image, visible as a red crosshair in a circle in Figure 4.1 (a). This extracted coordinate was then manually entered into Google Earth Pro and the result of this is shown in Figure 4.1 (b). This, at least, provides some verification that the coordinate reference system stored within the aerial photographs are correct and by extension, all the subsequent transformations, plotting and extracting commands in R must have yielded the correct results along the way.

(a)   (b)
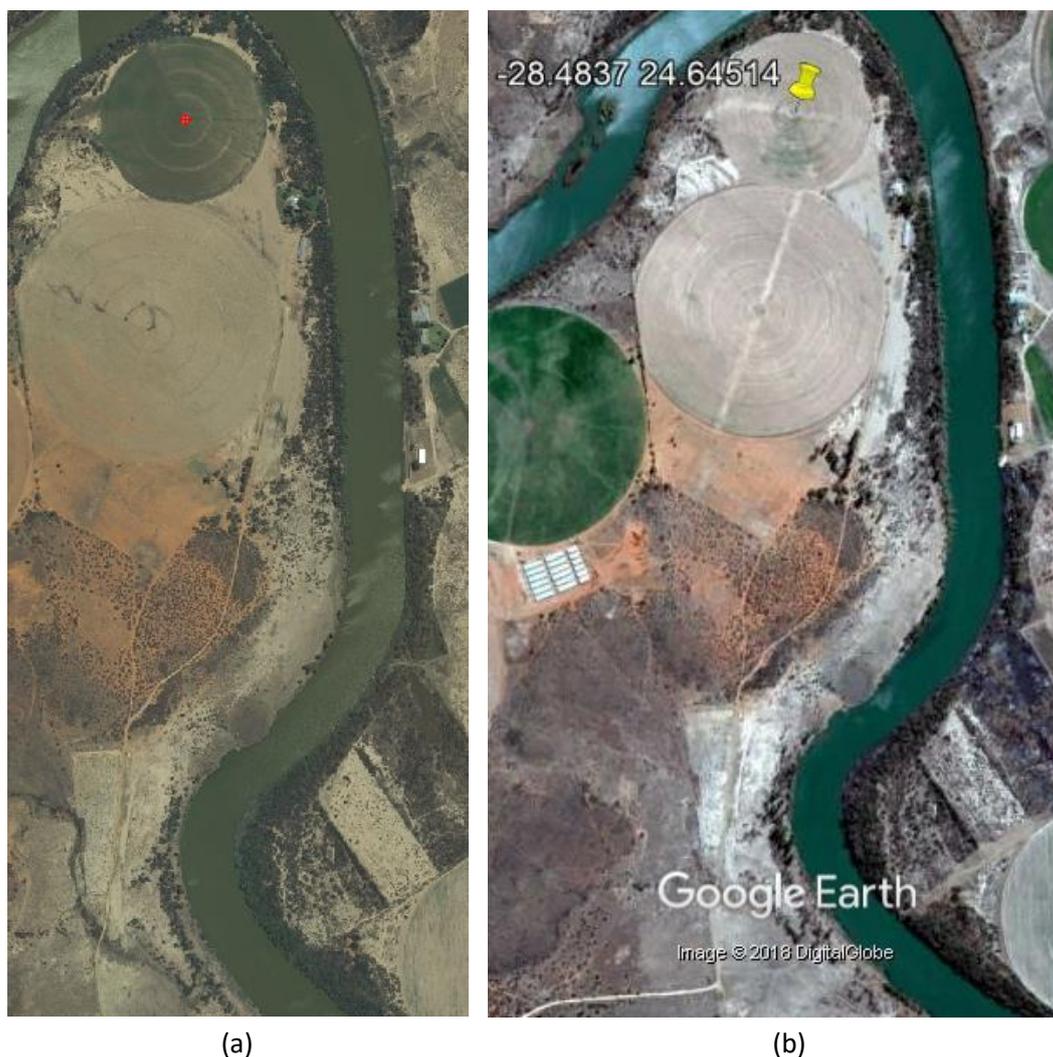
Figure 4.1: Comparison of R plotted aerial photograph and resultant Google aerial photograph at the same coordinate

With the R plotting of raster images verified, it removed the need to use further third-party software for the rest of the verification processes. As was the case for verifying the topographical data. The topographical data's actual height values could not be verified, but the general accuracy of the shapefiles in terms of plotting positions could be visually inspected. The most effective method for this is to match the topographical data of a naturally inclined area which is identifiable in aerial photos, such as a hill or mountain. The surrounding area of GID 2035415, shown in Figure 4.2, was selected to illustrate here. Firstly, the uncropped aerial file from the source data was plotted and then the topographical data for the area was added to the plot as an extra layer on top of this. Visually, while not entirely

without uncertainty, the topographical data matches exactly with what can be interpreted from the underlying aerial photograph. This can then, at least, verify that the coordinate reference system stored within the topographical data is correct and that the data represents reality.

Figure 4.2: Visual verification of topographical data

The solar radiation is verified with a completely different tactic, namely *blind trust*. The source of the original dataset is NASA and OpenEI is the keeper of the publicly available dataset. Both these organisations have a larger degree of institutional trust accompanying their reputations. While this might be enough to accept the validity of the dataset as a whole, it is still helpful to perform a sanity check. To do this, recall Figure 1.3 from Chapter 1. On this SolarGIS map of South Africa, the city of Upington in the Northern Cape is in an area which, by visual inspection, receives solar radiation somewhere between 2 200 and 2 300 kWh/m² per annum. When the coordinates of the centre of Upington is plotted on the NASA solar dataset, the value returned is 5,87 kWh/m² per day over an average of a year, which equates to 2144 kWh/m² per annum, which is in the same ballpark with ease.

The last sanity check is related to the calculation of land area. There is a large reliance on these types of calculations in this thesis and therefore a small test to verify these techniques would be welcome. The total land area size of Northern Cape is 372 889km² (Lehohla 2011) according to a report by Statistics South Africa. When using the same methods that were

implemented in Chapter 3, R yields a result of 373 396km^2, which compares very favourably at this scale.

4.2 Solar Potential per Mine

The solar potential presented in this section will be dealt with in two manners. The first part will purely report on the results of the calculations done up until this point in order to obtain the solar insolation values for each individual area investigated. The second part will build on these values with a few assumptions to reach more real-world results. Only five of the studies in the literature study went beyond the concept of land suitability and brought in the idea of power plant output and therefore it was not considered to be part of the scope of this paper. However, an attempt will be made for the sake of completeness, but concepts such as system performance ratio (sometimes called annual performance ratio), inverter efficiency and temperature efficiency were not included.

Because the assessed areas vary wildly in size, the results also vary as such. The area with the most insolation received value stands at a healthy 1 609 358 MWh/day, while the minimum is a mere 2MWh/day. Because of this range, the results shown in Table 4.2 are summarised according to its order of magnitude in a pseudo-logarithmic style. The majority of investigated areas fall in the 10-100 GWh/day category and the broader 1-100 GWh/day category holds the outright majority share.

Table 4.2: Number of areas and its respective solar insolation received value

0-10 MWh/day	3
10-100 MWh/day	7
100-1000 MWh/day	18
1-10 GWh/day	27
10-100 GWh/day	33
100-1000 GWh/day	21
1-10 TWh/day	4
Total	113

While knowing the absolute value of an area is a necessary result, this might be less applicable to the owner of the land given that the purpose of the land is assigned to mining. A more relevant number to know would be how much of the current land is excluded from solar energy production, as this would be indicative of the freedom to place a solar plant in an area

less preferred for its current purpose. Table 4.3 expresses this, but given the length of such a list, if provided for each area, it is expressed as a summary noting the number of areas which fall into ranges of percentage. Expressed in words, 68 of the 113 areas investigated only has 0%-25% of its land area currently classified as an excluded area. Therefore, for these 68 areas, which represents the outright majority, at least 75% of their land area is available to place a solar PV plant.

Table 4.3: Number of areas and the corresponding percentage of land area which is classified as excluded areas.

0%-25%	25%-50%	50%-75%	75%-100%	Total areas
68	24	9	12	113

In the previous section, the results of all the calculations were deliberately presented as the pure insolation value received per area, because this was dictated by the literature study outcome. This section will dare to build on this with a few assumptions on solar PV plant operation and panel efficiency to bring this value more in line with real-world expectations of the peak power that can be expected. The factors that have the biggest effect on how much of the received solar insolation is transformed into electrical power on the grid is panel efficiency and area factor. The authors of the literature study whom included panel efficiency vary on this topic, with (Abed & Nagasaka 2017) reporting 8.8% to 37.9%, (Charabi & Gastli 2011) reported 7.9% to 13.1%, Merrouni et al. (2016) reported 11.89% and Yushchenko et al. (2018) reported a host of ranges. These ranges were mostly referenced to meta-studies on the subject at hand and can therefore be seen as the most reliable. Abbreviating the result, delivers efficiencies of between 13.3% to 22% for monocrystalline cells and 12% to 15.67% for polycrystalline cells. The simplest case would be to assume a *high* and *low* case with values set at 12% and 22% respectively.

Area factors are an indication of what fraction of a specific area can be covered with panels since it is inevitable that space has to be left open between rows and for service roads. The only paper explicitly including this is Gastli & Charabi (2010), which included an area factor of 0.7 when calculating electric power generation potential. The formula implemented is similar to that implemented by Carrión et al. (2008) and the adaptation used in this paper is given by Equation 1:

$$E = G \times A \times \varepsilon$$

where:

E = Electric power generation per day (MWh/day)

G = Solar radiation received (MWh/day)

A = Area factor

ε = Panel efficiency

Table 4.4 summarises the effect of Equation 1 by comparing the resultant values of the *low* and *high* cases with the original pure insolation values from Table 4.2 above. The total generation from the beginning of this chapter also decreases from 12,06 TWh/day to 1,01 TWh/day and 1,86 TWh/day for the *low* and the *high* cases respectively. All these numbers are in Table 4.5 together with the original values from which they were derived at the beginning of the chapter.

Table 4.4: High and Low case after performing Equation 1 on Table 4.2

	From Table 4.2	*Low* Case	*High* Case
0-10 MWh/day	3	10	6
10-100 MWh/day	7	21	17
100-1000 MWh/day	18	24	28
1-10 GWh/day	27	34	29
10-100 GWh/day	33	23	27
100-1000 GWh/day	21	1	6
1-10 TWh/day	4	0	0
Total	113	113	113

Table 4.5: Summary of headline values in this chapter

Total insolation – Gross [TWh/day]	Total insolation – minus exclusions [TWh/day]	Energy production – *High* case [TWh/day]	Energy production – *Low* case TWh/day]
14,88	12,06	1,86	1,01

4.3   Sensitivity Analysis

The results shown in Table 4.5 show a *low* case and a *high* case. The breadth of difference between these two results reveals the importance of accurately setting the values of input variables and the impact it has on the output. This, at the core, is what encompasses a sensitivity analysis and will be explored further in this chapter.

A look at the studies in the literature study shows that of the 16 investigated, only the studies by Aly et al. (2017), Carrión et al. (2008), Sánchez-lozano et al. (2013) and Yushchenko et al. (2018) mention that it

would be possible to perform a sensitivity analysis and only Garni & Awasthi (2017) seems to go as far as implementing a sensitivity analysis. This is a surprising result, given that the majority of the MCDA techniques used by the studies in the literature study use some form of weighted variable to construct a decision matrix. In such cases, it would be prudent to perform at least an uncertainty analysis or sensitivity analysis or both, to quantify the effect of each weight as well as that of each variable.

In this study, such weights were not present and this greatly simplifies the sensitivity analysis design. Firstly, the sensitivity of a variable was defined with Equation 2 as:

$$\epsilon = \frac{\delta}{\gamma}$$

where:

ϵ = Sensitivity (Dimensionless)

δ = Percentage change observed in output compared to the base case (%)

γ = Percentage change effected in input compared to the base case (%)

Secondly, a base case needs to be selected. The *low* case result of the previous chapter was arbitrarily selected and set as the base case. This selection implies that the output variable being monitored is the total energy production potential per day. The input selection, on the other hand, was not as simple and in some instances, base variables were selected and in other cases, the intermediate results of specific processes (outputs that were used as inputs in sequential calculations) were selected. The selected input variables were:

- o Insolation
- o Slope
- o Area factor
- o Panel efficiency
- o Exclusion areas

The first four variables are simply adjusted by varying the numeric value representing each variable respectively. The last variable is the result of the manual process of selecting exclusion areas with polygons and therefore tests the model's sensitivity with respect to the accuracy of this process. All the variables that led to the creation of each area's DEM were purposefully excluded. This is because the adjustment of a variable such as the *inverse distance power* would require each DEM to be recalculated for

each area. This is only one variable and the computational cost for adjusting all DEM input variables could potentially be weeks. It is also noted, as mentioned in section 3.7.1, that comparing interpolation techniques is a field of study on its own and therefore beyond the scope of this sensitivity analysis.

The amount by which the input variables are varied were iteratively explored by increasing one variable to a value which can be reasonably considered to be the hard-maximum for that variable. That increase was then applied to the remaining variables to observe whether any of them would exceed their own maximum values. For example, if the increase in panel efficiency from the *low* case to the *high* case is applied to other variables, then the *area factor* variable would increase to a value above one, which is beyond the defined limits for that variable. It was found that an increase of 42.86% would satisfy all variables.

Some of the results of the sensitivity analysis can be theoretically deduced. Area factor and panel efficiency are two of the multiplication terms found on the RHS of Equation 1. The product on the LHS of Equation 1 is the selected output variable, therefore it can be concluded that the sensitivity of these input variables is exactly one, according to the definition for sensitivity as defined in Equation 2. Nevertheless, this was tested empirically by changing the parameters of the model and found to be exactly one, shown in Figure 4.3, as predicted theoretically.
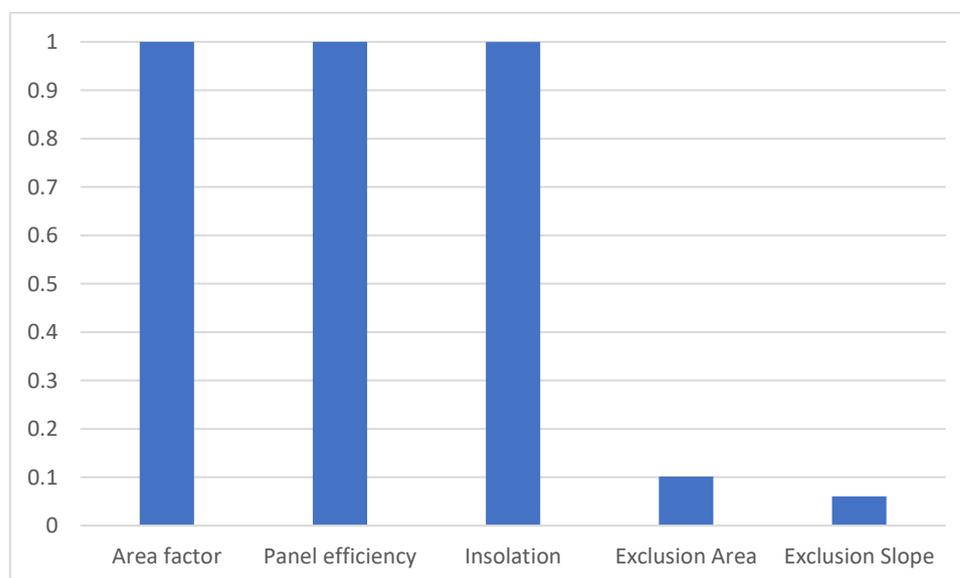
Figure 4.3: Result of sensitivity analysis.

The sensitivity of *insolation* could also have been predicted theoretically. When calculating the nett energy for any localised area, the

result is obtained by first calculating the gross energy received at that area and then subtracting the energy representing the exlusions in that area. If the area considered is small enough, then the insolation value for both terms of the summation would be the same, therefore becoming a common denominator. The outcome is that the nett energy would increase or decrease directly proportional to insolation.

Chapter 5

5. Conclusions and Recommendations

Chapter 4 presented the results after all calculations were completed. These results were only presented as pure numbers without inferring too much from the data. To satisfy the objective of using data to inform stakeholders, the following sections will infer possible routes of action by combining the data results with the known policy landscape. The data results were obtained by using a simplified multi-criteria decision system based on Boolean logic, applied within the RStudio software environment as a series of scripts. The inputs to these scripts were unaltered data from various sources. These steps realised the sub-objectives set out in Chapter 1, which combined together, reached the ultimate goal of calculating the total solar PV generation potential of land located within the boundaries of licensed mines in the Northern Cape province of South Africa.

5.1 Important Findings

The whole of South Africa uses over 220 TWh of electricity annually (Krupa & Burch 2011). This number includes only electricity consumption and does not represent the total energy usage with other sources, such as liquid fuels used in the transport sector. In the previous chapter, it was found that the total annual generation potential on currently licensed mines in the Northern Cape province lies roughly between a high of 4 402 TWh and a low of 369 TWh (converted from 12.06 and 1.01 TWh/day respectively). This implies that even on a worst-case base estimation, using only the identified land to settle utility-scale solar PV plants, would satisfy the entire country's electricity demand requirements. As the research focussed purely on generation potential, elements such as dispatchability were not included and realistically the number can be much lower than this.

The main factors that influence dispatchability, outside of policy limitations, are related to how the national grid is organised as well as utilised from a demand-side perspective. Figure 5.1 shows how the grid is currently organised. From this, it is visible that the majority of generation and consumption reside intertwined in the northern parts of the country. This is another coincidental symptom of the *minerals-energy complex,* as geographically, where cheap coal used to be found in the same region where the economy rapidly grew around the minerals extraction and beneficiation

industries. Naturally, energy had to be transported to the south and east where these coal resources, and a lack of other resources to competitively generate electricity, were not naturally occurring. Figure 5.1 also shows the location of where a broad selection of the REIPP procurement programme project is located. It is clear that solar and wind resources are located in areas which were previously nett receivers of energy, with the Northern Cape province under-developed in terms of large-scale grid infrastructure. This disconnect between the locations of generation and consumption, in essence, is the reason why solar-generated power will not be able to satisfy the entire country's electricity needs, even if the resource exceeds the demand. To achieve this, the national grid would have to be reorganised. The fact that the national grid operator is also the single largest generator of coal-derived electricity is a major conflict of interest in achieving this goal and again highlights the importance of the need for the ISMO bill to be promulgated.
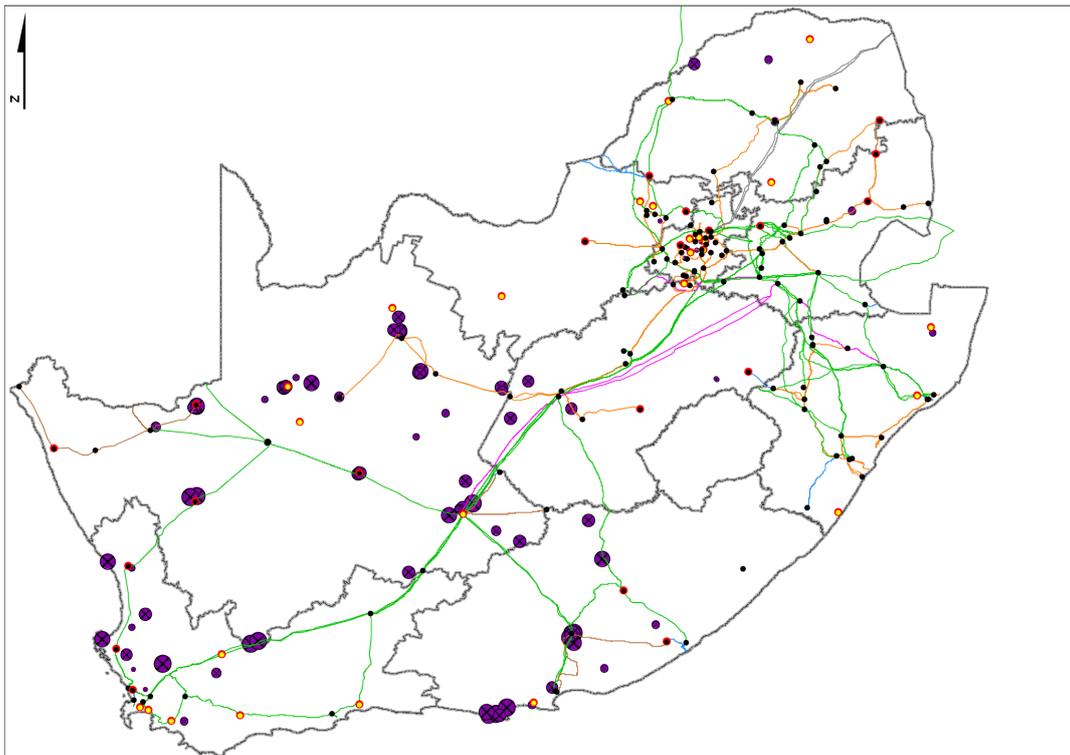
Figure 5.1: National electricity grid of South Africa. Solid lines show transmission lines of different carrying capacity. Black, red and yellow dots show current Eskom infrastructure (sub-stations) and purple dots around crosshairs show REIPP procurement program projects. Source: (Eskom Transmission Division 2015)

Solar photovoltaic derived electricity has always had one Achilles heel related to the utilisation of the energy, or demand-side profile. Figure 5.2 illustrates this problem. The typical daily demand curve for electricity has

two peaks; a big early evening peak and a less severe morning peak which becomes less prominent in summer. Comparing this profile with the typical solar PV daily generation curve shows an almost inverse relationship to the demand profile. Peak generation overlaps with a period of time in the day when usage slumps, while generation drop-off times overlap with those of the demand peaks. The result is that, without some form of energy storage, solar PV would never be able to supply hundred percent of the country's electricity demand.
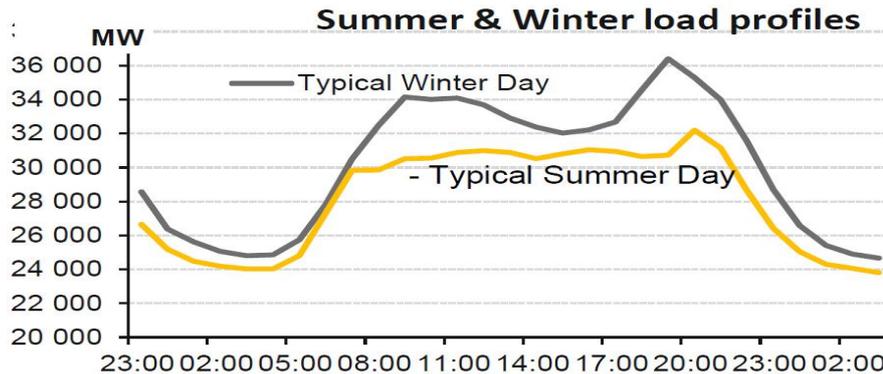
(a)
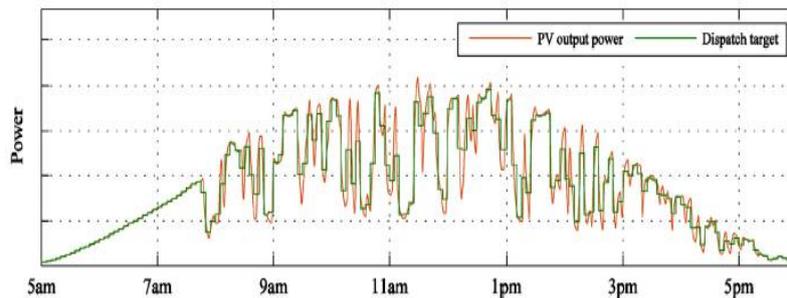
(b)

Figure 5.2 Typical load profile (a). Source: (Matona 2014) and a typical PV generation profile (b). Source: (Wang et al. 2015)

5.2   Recommendations to Stakeholders

In previous chapters, three stakeholders were identified, namely: Mining companies, solar energy developers and policy makers/government. These parties have an intertwined role in the case of energy generation. It is therefore that none of these entities works in silos, independently from each other. The aim of this section is to illustrate the dependence of each entity on the others and the opportunities that can be realised if small changes are made to the current status.

Mining companies. Mining companies, like all companies, are profit driven. Therefore, if there is a business case to develop solar PV plants on

mines and there are no large legislative stumbling blocks, it is reasonable to think that it will proceed. Two business cases for this development exists, being an own-consumption model or as a type of IPP trading in power generated. The own-consumption model is by far the simplest and presents the least number of variables to analyse in terms of its business sense. This decision is for each individual mine to make, but possible variables can relate to the cost of obtaining a grid connection to a previously unconnected area or to avoid loss of services by the grid utility. Whatever the outcome of such an analysis might be, the first question should be whether enough reliable solar power can be generated. The average annual electricity consumption per mine varied from 4.2 GWh to 2752 GWh (Votteler & Brent 2017b) in the sample of their study. This paper found that the annual electricity generation potential varied between 36.5 GWh and 3 650 GWh (100 MWh – 10GWh per day) for the majority of the mines analysed. This overlaps very favourably with the consumption figures and thus leaves only one matter which this paper can comment on, which is whether it is allowed. Chapter 1 covered some of the Acts regulating the mining industry and it was found that mining rights/permit holders have the freedom to bring onto the land being mined any equipment and erecting any structure which is used for, or incidental to, mining activities. While the final decision would be in the hands of the Minister, it can be stated that there should be no reason insofar as generation potential or policy blockades, that would prevent solar PV plant development for own-consumption.

Solar energy developers. While mining companies will be the ultimate beneficiary of a solar PV plant, it will in all likelihood be a solar energy developer responsible for the design and build program. Mining companies' expertise lies in mining and not solar PV development and this presents a business opportunity for solar energy developers. Many of the current solar energy developers have experience with developing environmental protection programmes as well as social development schemes, all of which overlaps with the requirements imposed on mining companies in order to obtain mining rights/permits. Their role would not necessarily have to be limited to developing for the own-consumption model. Chapter 1 illustrated that, depending on interpretation, mines could be allowed to export generated electricity after mine closure as part of the policies that govern environmental rehabilitation of a previously mined area. This would open various permutations of co-ownership and commercial relationships between mining companies and solar energy developers. An example would be for a mining company to supply the capital to establish a solar PV plant for own-consumption while the developer stays on in an operational role. Once the

mine closes, majority ownership is transferred to the developer which has the necessary skills to negotiate wheeling the energy on the open market, while the mining company retains a minor stake in ownership.

Policymakers. Excluding the own-consumption model, any other options discussed above are dependent on either small policy changes or leniency in execution. It would also require inter-departmental cooperation. This has already been proven successful between the Departments of Energy, Environmental Affairs and Water Affairs during the development and execution of the REIPP procurement program and would only require a single department, the Department of Mineral Resources, to be aligned. For the Department of Energy, the REIPP procurement program has been a lauded success, but the uptake of the small-scale generation program within it has been lacklustre. This paper presents an avenue of development which would require minimum effort and play to the strengths of all the established stakeholders to mitigate this. Beyond the scope of the small-scale generation program, it was shown earlier that the investigated area has the potential to satiate the entire country's demand, but limitations were identified regarding dispatchability. Policy can be a very powerful and effective tool and, in this instance, it is imperative that policymakers finalise the ISMO bill for promulgation and that government passes it into law. This would not directly solve the design imbalance of the national grid but would create the incubator from where solutions can be created. Agile policy could also play a big role in load-generation matching by shifting or flattening peak load times through mechanisms such as time-dependent pricing strategies on the demand-side. On the generation-side, incorporating a favourable pricing strategy in the negotiated PPA for supply during current peak times could allow the financial impetus required to make incorporated energy storage financially viable for energy developers, as in the widely publicised Tesla battery story in southern Australia (Krieger 2018). In addition, investing in further research into storage technologies, such as currently the case between government and private sector on a vanadium redox flow battery project (James 2018), could help alleviate peaking supply problems and stimulate GDP growth by making South Africa a serious player in a global market where demand for storage solutions will only get bigger in the future.

5.3 Simulation Limitations

Throughout the course of this research R and the relevant imported libraries, as accessed through RStudio, proved to be a very capable and

versatile GIS package. While the learning curve was steep, the code-based interface allowed small adaptations to be easily applied to previously executed code. This is also advantageous when many similar manipulations are required to large amounts of data, as these can be semi-automated. The inherent difficult learning curve also forced down a better understanding of the subject matter at hand, as this is required for fault-finding in unexpected calculation results. Fault-finding itself is relatively convenient, as code-based interfaces such as those found in most programming languages have well-established processes to do so. It can be concluded, in line with one of the initial objectives of this paper, that the use of R in the academic realm for the use of GIS research, is a viable and cost-effective alternative to commercial GIS software packages.

The fact that the user has to become competent in the subject at hand in order to write meaningful code, implies that a large amount of time and effort has to be spent on various subjects in order to complete a task such as this research. Within a limited time-budget, it implies that some subjects have to be neglected and, in this research, that particular subject is the one of interpolation. Interpolation was not considered as part of the scope of the research objective, but it was a necessary step in achieving those objectives. Only the simplest methods, the Kriging family of statistical interpolation and the IDW deterministic methods were considered. Either of these performs perfectly well on point data representing elevation, but when that point data is obtained by digitizing topographical lines, it introduces an artefact in the form of sharp ridges at the position where the topographical lines use to reside. This exact warning was issued by Jones (1998) in a paper comparing interpolation methods.

5.4 Suggested Future Research

A factor that stood out when comparing the studies of the literature study with this paper, is the amount of data which is readily available to include in other studies, especially of European origin. While the NGI perform a laudable job, it can only be assumed that a shortage of funding requires prioritization to lean toward projects which would not overlap with the private sector or possibly even academia. This presents an opportunity for energy research specifically to capture data which can enhance the ability to draw conclusions from the performed studies.

One such example would be the capture of lower voltage electricity lines. While GIS sources exist for transmission lines, these sources typically only include very high voltage transmission lines and not the final

distribution lines found beyond the last or second last transformer. Logging the capacity, length and the approximate position of these lines in an expansive data capturing project as part of the future research would grant greater insight into the ability of mines in the role of energy generators, as opposed to only the self-consumption model.

Combined with the above, an audit on the energy usage patterns of mines in the Northern Cape province might also reveal a nuance specific to these mines. Gold- and platinum mines feature in many research articles, but the lesser valuable ore typically mined in the sands of the Northern Cape province might probably not share the same energy usage characteristics as those mines. This, in turn, might strengthen or weaken either argument supporting a self-consumption or nett energy export model.

As part of the requirements set out in the Minerals and Petroleum Resources Development Act, mines are required to submit detailed site plans as well as environmental protection and rehabilitation programmes. These documents would be a great resource to provide insight into the level of environmental accountability of the mines, as signed off by the Minister. The result could be critical in establishing whether the "pre-determined state" clause is frequently implemented and to what degree, as this would set the established precedent for potential solar plant development on previously mined land.

List of References

Abed, M. & Nagasaka, K., 2017. Utility-scale implementable potential of wind and solar energies for Afghanistan using GIS multi-criteria decision analysis. *Renewable and Sustainable Energy Reviews*, 71(April 2016), pp.150–160.

Alami, A., Elwali, F. & Mezrhab, A., 2018. Large scale PV sites selection by combining GIS and Analytical Hierarchy Process . Case study : Eastern Morocco. *Renewable Energy*, 119, pp.863–873.

Aly, A., Jensen, S.S. & Pedersen, A.B., 2017. Solar power potential of Tanzania : Identifying CSP and PV hot spots through a GIS multicriteria decision making analysis. *Renewable Energy*, 113, pp.159–175.

Arán Carrión, J. et al., 2008. Environmental decision-support systems for evaluating the carrying capacity of land areas: Optimal site selection for grid-connected photovoltaic power plants. *Renewable and Sustainable Energy Reviews*.

Babak, O. & Deutsch, C. V., 2008. Statistical approach to inverse distance interpolation. *Stochastic Environmental Research and Risk Assessment*, (July).

Baker, L., 2011. Governing electricity in South Africa: wind, coal and power struggles. , (July), pp.1–27.

Van den Berg, J., 2013. *Submission to NERSA : Eskom MYPD3 application*,

Bivand, R., 2006. Implementing Spatial Data Analysis Software Tools in R. *Geographical Analysis*, 38(1), pp.23–40.

Bivand, R. & Gebhardt, A., 2000. Implementing functions for spatial statistical analysis using the R language. *Journal of Geographical Systems*, 2(3), pp.307–317.

Bivand, R., Keitt, T. & Rowlingson, B., 2016. rgdal: Bindings for the Geospatial Data Abstraction Library.

Bivand, R. & Lewin-Koh, N., 2017. maptools: Tools for Reading and Handling Spatial Objects.

Bivand, R. & Rundel, C., 2017. rgeos: Interface to Geometry Engine -

Open Source (GEOS).

Blondel, E., 2017a. cleangeo: Cleaning Geometries from Spatial Objects.

Blondel, E., 2017b. Cleaning Geometries from Spatial Objects.

Borgogno Mondino, E., Fabrizio, E. & Chiabrando, R., 2014. Site Selection of Large Ground-Mounted Photovoltaic Plants: A GIS Decision Support System and an Application to Italy. *International Journal of Green Energy*, 12(5), pp.515–525.

Boyse, F. et al., 2014. *Sunshine for mines: Implementing renewable energy for off-grid operations*, Washington DC.

Brito, M.C. et al., 2012. Photovoltaic potential in a Lisbon suburb using LiDAR data. *Solar Energy*, 86(1), pp.283–288.

BusinessTech, 2018. How SA's new Carbon Tax will affect petrol and electricity prices. *BusinessTech*. Available at: https://businesstech.co.za/news/energy/218691/how-sas-new-carbon-tax-will-affect-petrol-and-electricity-prices/ [Accessed July 22, 2018].

C. Parrado; A. Girard; F. Simon; E. Fuentealba, 2016. 2050 LCOE projection for a hybrid PV-CSP plant in the Atacama Desert, Chile. *Energy*, 94, pp.422–430.

Carrión, J.A. et al., 2008. The electricity production capacity of photovoltaic power plants and the selection of solar energy sites in Andalusia (Spain). *Renewable Energy*, 33(4), pp.545–552.

Charabi, Y. & Gastli, A., 2011. PV site suitability analysis using GIS-based spatial fuzzy multi-criteria evaluation. *Renewable Energy*, 36(9), pp.2554–2561.

Council for Scientific and Industrial Research (CSIR), 2016. Wind and Solar PV Resource Aggregation Study for South Africa. , (March).

Dekker, J. et al., 2012. Investigating the effects of solar modelling using different solar irradiation data sets and sources within South Africa. *Solar Energy*, 86(9), pp.2354–2365.

Department of Energy, 2011. *Integrated Resource Plan for Electricity 2010-2030*, Pretoria.

Department of Energy, 2015. *State of Renewable Energy in South Africa*,

Department of Mineral Resources, 2002. *A Guideline for a Mining Work Programme to be Submitted for a Applications for a Mining Right in Terms of the Minerals and Petroleum Resources Development Act.*,

Republic of South Africa.

Department of Mineral Resources, 2015. List of operating mines. Available at: www.dmr.gov.za [Accessed November 20, 2015].

Department of Minerals and Energy, 2008. Electricity Pricing Policy of the South African Electricity Supply Industry. *Government Gazette*, (31741).

Department of Minerals and Energy, 2003. *White Paper on Renewable Energy*, Pretoria.

Department of Minerals and Energy, 1998. *White Paper on the Energy Policy of the Republic of South Africa*, Pretoria.

Department of Rural Development and Land Reform: National Geo-Spatial Information, Colour Digital Aerial Imagery at 0.5m GSD (2008-2016) and 0.25 GSD (2017-current). Available at: http://www.ngi.gov.za/index.php/what-we-do/aerial-photography-and-imagery/35-colour-digital-aerial-imagery-at-0-5m-gsd-2008-2016-and-0-25m-gsd-2017-current [Accessed July 30, 2015].

Diakoulaki, D. & Karangelis, F., 2007. Multi-criteria decision analysis and cost-benefit analysis of alternative scenarios for the power generation sector in Greece. *Renewable and Sustainable Energy Reviews*.

Domínguez Bravo, J., García Casals, X. & Pinedo Pascua, I., 2007. GIS approach to the definition of capacity and generation ceilings of renewable energy technologies. *Energy Policy*, 35(10), pp.4879–4892.

Eberhard, A. & Gratwick, K.N., 2011. IPPs in Sub-Saharan Africa: Determinants of success. *Energy Policy*, 39(9), pp.5541–5549.

Edkins, M.T., 2012. World Wind Energy Conference. Conference topic 2: Local, national and regional policies, barriers and incentives. In *Reviewing Wind Policy Risk in South Africa*. Bonn, pp. 1–11.

EIUG, 2018. EIUG – Energy Intensive User Group of Southern Africa. Available at: http://eiug.org.za/ [Accessed July 17, 2018].

Endel, F. & Filzmoser, P., 2012. R & GIS : Geospatial Plotting FFG IFEDH project , Student at Vienna University of Technology. , pp.1–6.

Energy Blog, 2018. South Africa: Utility scale technologies per province. Available at: http://www.energy.org.za/data-and-tools/data-viz/south-african-utility-scale-technologies-per-province [Accessed July 21, 2018].

Eskom, 2017. Generation Plant mix. , p.2. Available at: http://www.eskom.co.za/AboutElectricity/FactsFigures/Documents/ GX0001GenPlantMixRev19.pdf [Accessed July 16, 2018].

Eskom Transmission Division, 2015. *Generation Connection Capacity Assessment of the 2022 Transmission Network*,

Fine, B. & Rustomjee, Z., 1996. *The political economy of South Africa : from minerals-energy complex to industrialisation*, Witwatersrand University Press.

Fischer-Kowalski, M. et al., 2011. *Decoupling natural resource use and environmental impacts from economic growth, A Report of the Working Group on Decoupling to the International Resource Panel.*,

Fluri, T., 2009. Solar Resource Mapping in South Africa. , (March), p.29.

Fraunhofer Institute, 2012. *Photovoltaics Report*, Freiburg.

Garni, H.Z. Al & Awasthi, A., 2017. Solar PV power plant site selection using a GIS-AHP based approach with application in Saudi Arabia. *Applied Energy*, 206(October), pp.1225–1240.

Gastli, A. & Charabi, Y., 2010. Solar electricity prospects in Oman using GIS-based solar radiation maps. *Renewable and Sustainable Energy Reviews*, 14(2), pp.790–797.

Google LLC, 2018. Google Earth Pro.

Gorsevski, P. V. et al., 2013. A group-based spatial decision support system for wind farm site selection in Northwest Ohio. *Energy Policy*, 55, pp.374–385.

Graler, B., Pebesma, E. & Heuvelink, G., 2016. Spatio-Temporal Interpolation using gstat. *The R Journal*, 8(1), pp.204–218.

Gratwick, K.N. & Eberhard, A., 2008. Demise of the standard model for power sector reform and the emergence of hybrid power markets. *Energy Policy*, 36(10), pp.3948–3960.

Grunsky, E.C., 2002. R: a data analysis and statistical programming environment - an emerging tool for the geosciences. *Computers & Geosciences*, 28, pp.1219–1222.

Hall, H., 2012. Mineral and Petroleum Resources Development Regulations. , 45056(513), pp.1–67.

Hijmans, R.J., 2016. raster: Geographic Data Analysis and Modeling.

Independent Power Producer Office, 2017a. *Independent Power Producers*

Procurement Programme (IPPPP): An Overview, Centurion.

Independent Power Producer Office, 2017b. *REIPPPP focus on Northern Cape: Provincial Report Volume 1*, Centurion.

Independent Power Producer Office, 2015. *REIPPPP focus on Northern Cape*, Centurion.

Jahangiri, M. et al., 2016. Finding the best locations for establishment of solar-wind power stations in Middle-East using GIS : A review. *Renewable and Sustainable Energy Reviews*, 66, pp.38–52.

Jakubiec, J.A. & Reinhart, C.F., 2013. A method for predicting city-wide electricity gains from photovoltaic panels based on LiDAR and GIS data combined with hourly Daysim simulations. *Solar Energy*, 93, pp.127–143.

James, N., 2018. South Africa to pilot first utility-scale vanadium redox flow battery. *Engineering News*. Available at: http://www.engineeringnews.co.za/article/lessons-from-the-deployment-of-first-utility-scale-redox-battery-2018-02-23 [Accessed October 19, 2018].

Janke, J.R., 2010. Multicriteria GIS modeling of wind and solar farms in Colorado. , 35, pp.2228–2234.

Jones, K.H., 1998. A Comparison of Two Approaches to Ranking Algorithms Used to Compute Hill Slopes. *GeoInformatica*, 2(3), pp.235–256.

Krieger, N., 2018. After One Year of Operation, Tesla's Australian Mega Battery Is Doing Just Fine. *engineering.com*. Available at: https://www.engineering.com/ElectronicsDesign/ElectronicsDesignArt icles/ArticleID/17746/After-One-Year-of-Operation-Teslas-Australian-Mega-Battery-Is-Doing-Just-Fine.aspx [Accessed October 20, 2018].

Krupa, J. & Burch, S., 2011. A new energy future for South Africa: The political ecology of South African renewable energy. *Energy Policy*, 39(10), pp.6254–6261.

Lant, K., 2017. The World's Largest Floating Solar Plant Is Finally Online. *Futurism*. Available at: https://www.sciencealert.com/the-world-s-largest-floating-solar-plant-is-finally-online?perpetual=yes&limitstart=1 [Accessed August 19, 2018].

Latinopoulos, D. & Kechagia, K., 2015. A GIS-based multi-criteria evaluation for wind farm site selection . A regional scale application in

Greece. , 78, pp.550–560.

Lehohla, P., 2011. *Provincial profile: Northern Cape - Census 2011*,

Lukač, N. et al., 2013. Rating of roofs' surfaces regarding their solar potential and suitability for PV systems, based on LiDAR data. *Applied Energy*, 102, pp.803–812.

Malczewski, J., 2006. GIS-based multicriteria decision analysis: A survey of the literature. *International Journal of Geographical Information Science.*

Merrouni, A.A., Mezrhab, A. & Mezrhab, A., 2016. PV sites suitability analysis in the Eastern region of Morocco. *Sustainable Energy Technologies and Assessments*, 18, pp.6–15.

Minister of Finance, 2017. Draft Carbon Tax Bill.

Montmasson-Clair, G. & Ryan, G., 2014. Lessons from South Africa's Renewable Energy Regulatory and Procurement Experience. *Journal of Economic and Financial Sciences*, 7(1), pp.507–526.

Nakumuryango, A. & Inglesi-Lotz, R., 2016. South Africa's performance on renewable energy and its relative position against the OECD countries and the rest of Africa. *Renewable and Sustainable Energy Reviews*, 56, pp.999–1007.

National Academy of Sciences, 2017. Project: Reproducibility and Replicability in Science. *National Academy of Sciences.* Available at: https://www8.nationalacademies.org/pa/projectview.aspx?key=49906 [Accessed August 16, 2018].

van Niekerk, J.L., 2014. Renewable Energy in South Africa - Recent growth and future prospects with a focus on CSP. , (March).

Nkambule, N.P. & Blignaut, J.N., 2017. Externality costs of the coal-fuel cycle: The case of Kusile Power Station. *South African Journal of Science*, 113(9–10), pp.72–80.

Noorollahi, E. et al., 2016. Land Suitability Analysis for Solar Farms Exploitation Using GIS and Fuzzy Analytic Hierarchy Process (FAHP)—A Case Study of Iran. *Energies*, 9(8), p.643.

Oxford Analytica, 2013. SOUTH AFRICA: Pretoria eyes energy mix beyond coal. , pp.1–4.

Pebesma, E.J. & Bivand, R., 2005. Classes and methods for spatial data in R.

Petticrew, M. & Roberts, H., 2006. *Systematic Reviews in the Social Sciences: A Practical Guide*, Oxford: Blackwell.

PR Newswire Europe Including UK Disclose, 2016. South Africa to Export Its IPP Renewable Energy Programme in Collaboration with 11 African Nations. *PR Newswire Association LLC.*

R Core Team, 2015. R: A language and environment for statistical computing.

Reed, D. et al., 2003. *Towards a just South Africa: The political economy of natural resource wealth*,

Republic of South Africa, 1996. Constitution of the Republic of South Africa no. 108 of 1996.

Republic of South Africa, 2006. *Electricity Regulation Act (No. 4 of 2006) as ammended*,

Republic of South Africa, 1994. Government Gazette. *Government Gazette*, 354(16149).

Republic of South Africa, 2011. *Independent System and Market Operator Bill*, Pretoria: National Assembly.

Republic of South Africa, 2016. Integrated Energy Plan. *Government Gazette*, 617(40445).

Republic of South Africa, 2002. Mineral and Petroleum Resources Development Act. *Government Gazette*, 448(23922).

Rizzoli, A.E. & Young, W.J., 1997. Delivering environmental decision support systems: Software tools and techniques. *Environmental Modelling and Software.*

Le Roux, W.G., 2016. Optimum tilt and azimuth angles for fixed solar collectors in South Africa using measured data. *Renewable Energy*, 96, pp.603–612.

Sánchez-lozano, J.M. et al., 2013. Geographical Information Systems (GIS) and Multi-Criteria Decision Making (MCDM) methods for the evaluation of solar farms locations : Case study in south-eastern Spain. , 24, pp.544–556.

Sánchez-lozano, J.M. et al., 2014. GIS-based photovoltaic solar farms site selection using ELECTRE-TRI : Evaluating the case for Torre Pacheco , Murcia , Southeast of Spain. , 66, pp.478–494.

Science as fact, 2018. Science is facing a serious replication crisis, but there

are solutions. *Science as fact*. Available at: https://www.scienceaf.com/science-is-facing-a-serious-reproducibility-crisis-but-there-are-solutions [Accessed August 16, 2018].

Setianto, A. & Triandini, T., 2013. Comparison of Kriging and Inverse Distance Weighted (IDW) interpolation methods in lineament extraction and analysis. *Journal of Southeast Asian Applied Geology*, 5(1), pp.21–29.

Sing, M. et al., 2016. Estimation of Hong Kong ' s solar energy potential using GIS and remote sensing technologies. *Renewable Energy*, 99, pp.325–335.

SolarGIS, 2018. Solar resource maps of Germany | Solargis. Available at: https://solargis.com/maps-and-gis-data/download/germany/ [Accessed July 21, 2018].

Stellenbosch University, GeoSUN Africa & GeoModel Solar, 2014. SolarGIS GHI Map of South Africa. Available at: http://www.crses.sun.ac.za/files/research/publications/SolarGIS_GHI _South_Africa_width15cm_300dpi.png [Accessed July 21, 2018].

Suri, M., Cebecauer, T. & Skoczek, A., 2014. Cloud Cover Impact on Photovoltaic Power Production in South Africa. *2nd Southern African Solar Energy Conference*.

Tahri, M., Hakdaoui, M. & Maanan, M., 2015. The evaluation of solar farm locations applying Geographic Information System and Multi-Criteria Decision-Making methods : Case study in southern Morocco. , 51, pp.1354–1362.

Urban-Econ, 2013. *Northern Cape Renewable Energy Strategy*,

Uyan, M., 2013. GIS-based solar farms site selection using analytic hierarchy process (AHP) in Karapinar region Konya/Turkey. *Renewable and Sustainable Energy Reviews*, 28, pp.11–17.

Votteler, R.G., 2016. *A mining perspective on the potential of renewable electricity sources for operations in South Africa.*

Votteler, R.G. & Brent, A.C., 2017a. A mining perspective on the potential of renewable electricity sources for operations in South Africa: Part 1 - The research approach and internal evaluation process. *Journal of Southern African Institute of Mining and Metallurgy*, 117(3), pp.285–298.

Votteler, R.G. & Brent, A.C., 2017b. A mining perspective on the

potential of renewable electricity sources for operations in South Africa: Part 2 - A multi-criteria decision assessment. *Journal of Energy in Southern Africa*, 117(3), pp.299–312.

Walker, S., 2015. Mine Power Options. *Engineering and Mining Journal*, (July), pp.34–40.

Wang, G., Ciobotaru, M. & Agelidis, V.G., 2015. Optimal capacity design for hybrid energy storage system supporting dispatch of large-scale photovoltaic power plant. *Journal of Energy Storage*, 3, pp.25–35.

Wickham, H. & Francois, R., 2016. dplyr: A Grammar of Data Manipulation.

Winkler, H. et al., 2012. How applicable are standard Solar Irradiation Maps ? Insights from Case Studies. *Proceedings of ISES-Europe Solar Conference*, p.8.

World Bank, 2018. Electricity production from coal sources (% of total) | Data. Available at: https://data.worldbank.org/indicator/EG.ELC.COAL.ZS?locations=ZA [Accessed July 17, 2018].

Yushchenko, A. et al., 2018. GIS-based assessment of photovoltaic (PV) and concentrated solar power (CSP) generation potential in West Africa. *Renewable and Sustainable Energy Reviews*, 81(April 2016), pp.2088–2103.

Zawilska, E., Brooks, M.J. & Meyer, A.J., 2012. A review of solar resource assessment initiatives in South Africa: The case for a national network. *World Renewable Energy Forum, WREF 2012, Including World Renewable Energy Congress XII and Colorado Renewable Energy Society (CRES) Annual Conferen*, 1(1), pp.1–7.

Addendum A

A.1 Code Repository: Reading and filtering solar insolation data

```r
library(rgdal)

#Lees NASA solar data- en Distrik shapefiles in
  Noordkaap.shape<-readOGR(dsn = "E:/Raw data/NC Shapefiles",layer =
  "District Municipality")
  Solar.shape<-readOGR(dsn = "E:/Raw data/Solar Data/Global Horizontal
  Irradiance/nasaghi278",layer = "GHI_NASA_low")

#Funksie neem vector met koordinaat limiete en rond minima af en maxima op
tot naaste heeltal.
  limiet_ronding <- function(input_vector){

    input_vector[1,1] <- floor(input_vector[1,1])
    input_vector[1,2] <- ceiling(input_vector[1,2])
    input_vector[2,1] <- floor(input_vector[2,1])
    input_vector[2,2] <- ceiling(input_vector[2,2])

    return(input_vector) }

#Funksie wat Noordkaap limiete vertaal na dieselfde verwysingsraamwerk
as NASA data en in vektor stoor
#Nasa data word in vektor gestoor genommer van links onder (ID1) tot
regs bo (ID64800) en dit stem oreen met
# koordinate (-90,-180) en (90,180) onderskeidelik.
  solar_data_filter_refs <- function(limiete){
    datarefvector <- c()

    for (i in limiete[2,1]:(limiete[2,2]-1)){ for (j in
      (limiete[1,1]+1):limiete[1,2]){ dataref <- (abs((-90)-i))*360 +
      (abs((-180)-j)) datarefvector <- c(datarefvector,dataref) }
    }
    return(datarefvector) }


#Vind maksimum en minimum lat/longs van Noordkaap en rond op/af tot naaste
heel tal wat alles insluit.
  limiete <- bbox(Noordkaap.shape) limiete <- limiet_ronding(limiete)

#Genereer verwysings vektor en gebruik dit om subset van NASA data te maak
beperk to relevant area datarefvector <- solar_data_filter_refs(limiete)
  Solar.shape.reduced <- Solar.shape[Solar.shape$ID %in% datarefvector ,]
```

```
#Skryf verkleinde shapefile, want subset bevat nog nulle in
oorspronklike posisies, dus word nog dieselfde hoeveelheid geheue
gebruik writeOGR(Solar.shape.reduced,"E:/Filtered Raw data/Solar
Data","GHI_NASA_NConly",
"ESRI Shapefile",overwrite_layer = TRUE)
Solar.shape<-readOGR(dsn = "E:/Filtered Raw Data/Solar Data", layer =
"GHI_NASA_NConly")
writeOGR(Solar.shape.reduced,"E:/Filtered Raw data/Solar
Data","GHI_NASA_NConly",
"ESRI Shapefile",overwrite_layer = TRUE)
```

A.2 Code Repository: Reading and filtering aerial photos

```r
library(rgdal)
library(raster)
library(rgeos)
library(sp)

#Implementering

  #Lees shapefile in wat grense van raster data (lugfotos) gaan bepaal.
  Noordkaap is net vir verwysing
  Mineralrights.shape<-readOGR(dsn = "E:/Raw data/NC Shapefiles",layer =
  "Mineral Rights Clean")
  Noordkaap.shape<-readOGR(dsn = "E:/Raw data/NC Shapefiles",layer =
  "District Municipality")

  #Transformeer projeksies na standaard WGS84 projeksie en ellips (in
  proj4 string formaat)
  Mineralrights.shape@proj4string <- CRS("+proj=longlat +ellps=WGS84
  +towgs84=0,0,0,0,0,0,0 +no_defs")
  Noordkaap.shape@proj4string <- CRS("+proj=longlat +ellps=WGS84
  +towgs84=0,0,0,0,0,0,0 +no_defs")

  #Skep plot as basis van visuele terugvoer om vordering te toon en skep
  error log plot(Noordkaap.shape)
  plot(Mineralrights.shape,col = "red",add = TRUE)
  Errorlog <- data.frame(ID=character(),Errormessage=character())

  #Lugfotos word in blokke groepeer wat in unieke subdirectories gestoor
  word. Hierdie skep 'n lys van daardie directories.
  setwd("E:/Raw Data/Aerial")
  subdirlist <- list.dirs(recursive = FALSE)

  #Hierdie lus itereer deur lys van subdirectories, maak 'n lys van tiff
  files in die huidige subdir en begin dan nog 'n lus van iterasies wat
  elke tiff file ondersoek om te bepaal of dit relevant is deur te
  vergelyk met die mineraal regte shapefile.
  for (i in 1:length(subdirlist)){ setwd(paste0("E:/Raw
    Data/Aerial",as.character(subdirlist[i]))) rasterfilelist <-
    list.files(pattern = ".tif",include.dirs = FALSE,recursive =
    FALSE,ignore.case = TRUE)

    #Laai elke raster om die beurt. Ontgin punte op hoeke wat die
    boundingbox voorstel. Kyk of die huidige raster gedeeltelik
    (gOverlaps) of geheel (gContains) enige van die polygons in
    Mineralrights bevat. TryCatch keer dat
    #R nie stop op enige foute nie, net die fout vertoon en aangaan.
    if (length(rasterfilelist)!=0){ for (j in 1:length(rasterfilelist)){
      tryCatch({ rastertocheck <-
```

```r
        raster(paste0(getwd(),"/",rasterfilelist[j])) raam <-
        projectExtent(rastertocheck,"+proj=longlat +ellps=WGS84
            +towgs84=0,0,0,0,0,0,0 +no_defs")
            raam.shape <- as(raam@extent, 'SpatialPoints')
            proj4string(raam.shape) <- "+proj=longlat +ellps=WGS84
            +towgs84=0,0,0,0,0,0,0
            +no_defs"
             resultvectorcontains <-
             gContains(Mineralrights.shape,raam.shape,byid = TRUE)
             resultvectoroverlaps <-
             gOverlaps(Mineralrights.shape,raam.shape,byid = TRUE)

        #Funksie wat foute probeer vang en vertoon sowel as stoor
        },error=function(e){ cat("ERROR :",conditionMessage(e), "\n")
            newerrorentry
            <data.frame(ID=as.character(rasterfilelist[j]),Errormessage=p
            aste(conditionMessage( e)))
            Errorlog <- rbind(Errorlog,newerrorentry) }
        )

        #Indien enige gOverlap of gContains TRUE toestand lewer, word
        die relevante raster kopieer na 'n ander bestemming.
        if ((any(resultvectorcontains,na.rm = TRUE)) ||
        (any(resultvectoroverlaps,na.rm = TRUE))){ sourcefilepath <-
        paste0(getwd(),"/",as.character(rasterfilelist[j]))
          destinationfilepath <- "E:/Filtered Raw Data/Aerial"
          file.copy(sourcefilepath,destinationfilepath,overwrite = TRUE)
          #Kopieer ook elke raster se ooreenstemmende .tfw file
          substring(sourcefilepath, (nchar(sourcefilepath)-3),
          nchar(sourcefilepath)) <-
          ".tfw" file.copy(sourcefilepath,destinationfilepath,overwrite =
          TRUE)
        } plot(raam.shape,pch = 0, col = "blue",add = TRUE) }
    }
  setwd("E:/Filtered Raw Data/Aerial") write.csv2(Errorlog,"Errorlog-
Aerialbroadfilter",sep = ",") }
```

A.3 Code Repository: Reading and filtering topographical data

```r
library(sp)
library(rgdal)
library(raster)
library(rgeos)

#Implementering

  #Lees shapefile in wat grense van raster data (lugfotos) gaan bepaal.
  Noordkaap is net vir verwysing
  Mineralrights.shape<-readOGR(dsn = "E:/Raw data/NC Shapefiles",layer =
  "Mineral Rights Clean")
  Noordkaap.shape<-readOGR(dsn = "E:/Raw data/NC Shapefiles",layer =
  "District Municipality")

  #Transformeer projeksies na standaard WGS84 projeksie en ellips (in
  proj4 string formaat)
  Mineralrights.shape@proj4string <- CRS("+proj=longlat +ellps=WGS84
  +towgs84=0,0,0,0,0,0,0 +no_defs")
  Noordkaap.shape@proj4string <- CRS("+proj=longlat +ellps=WGS84
  +towgs84=0,0,0,0,0,0,0 +no_defs")

  #Skep plot as basis van visuele terugvoer om vordering te toon en skep
  error log plot(Noordkaap.shape)
  plot(Mineralrights.shape,col = "red",add = TRUE)
  Errorlog <- data.frame(ID=character(),Errormessage=character())

  #Skep 'n lys van al die subdirectories waarin die topografiese files
  gestoor is.
  setwd("E:/Raw Data/Topo Shapefiles") subdirlist <- list.dirs(recursive =
  FALSE)

  #Hierdie lus itereer deur lys van subdirectories, maak 'n lys van
  topografies files in die huidige subdir en begin dan nog 'n lus van
  iterasies wat elke topografiese file ondersoek om te bepaal of dit
  relevant is deur te vergelyk met die mineraal regte shapefile.
  Topografiese shapefiles word gekenmerk deur "HYPS_ELEATION_LINES" in
  die file naam.
  for (i in 1:length(subdirlist)){ setwd(paste0("E:/Raw Data/Topo
    Shapefiles",subdirlist[i])) shapefilelist <- list.files(pattern =
    ".shp",include.dirs = FALSE,recursive = FALSE,ignore.case = TRUE)
    shapefilelist <-
    shapefilelist[grep("HYPS_ELEVATION_LINES",shapefilelist)]

    #Laai elke shapefile om die beurt. if (length(shapefilelist)!=0){ for
    (j in 1:length(shapefilelist)){ tryCatch({ shapefiletocheck <-
    readOGR(dsn = paste0(getwd()),layer =
```

```r
        substring(shapefilelist[j],1,(nchar(shapefilelist[j])-4)))
        proj4string(shapefiletocheck) <- "+proj=longlat +ellps=WGS84
            +towgs84=0,0,0,0,0,0,0 +no_defs"
        resultvectorcontains <-
        gContains(Mineralrights.shape,shapefiletocheck,byid = TRUE)
        resultvectoroverlaps <-
        gOverlaps(Mineralrights.shape,shapefiletocheck,byid = TRUE)

            #Funksie wat foute probeer vang en vertoon sowel as stoor
            },error=function(e){ cat("ERROR :",conditionMessage(e), "\n")
                newerrorentry <-
                data.frame(ID=as.character(rasterfilelist[j]),Errormessage=pa
                ste(conditionMessage( e)))
                Errorlog <- rbind(Errorlog,newerrorentry) }
            )

            #Indien enige gOverlap of gContains TRUE toestand lewer, word
            die relevante shapefile kopieer na 'n ander bestemming.
            if ((any(resultvectorcontains,na.rm = TRUE)) ||
            (any(resultvectoroverlaps,na.rm =
            TRUE))){ sourcefilepath <- paste0(getwd(),"/",shapefilelist[j])
                destinationfilepath <- "E:/Filtered Raw Data/Topo Shapefiles"
                file.copy(sourcefilepath,destinationfilepath,overwrite = TRUE)
                #Kopieer ook elke shapefile se ooreenstemmende .dbf en .shx
                file substring(sourcefilepath, (nchar(sourcefilepath)-3),
                nchar(sourcefilepath)) <".dbf"
                file.copy(sourcefilepath,destinationfilepath,overwrite = TRUE)
                substring(sourcefilepath, (nchar(sourcefilepath)-3),
                nchar(sourcefilepath)) <-
                ".shx" file.copy(sourcefilepath,destinationfilepath,overwrite =
                TRUE)
            } plot(shapefiletocheck,pch = 0, col = "blue",add = TRUE)
        }
    }
setwd("E:/Filtered Raw Data/Topo Shapefiles")
write.csv2(Errorlog,"Errorlog-Topobroadfilter",sep = ",") }
```

111

A.4 Code Repository: Clipping of solar insolation data with mine extents

```r
library(sp)
library(rgdal)
library(raster)
library(rgeos)

#Implementering

  #Lees shapefile in wat grense van solar data gaan bepaal. Skep index
  baseer op die unieke ID van elke area in die Mineral Rights shapefile
  wat vervolgens as die unieke ID van elke area in alle berekeninge
  gebruik sal word. NASA solar data kry ook 'n index sodat die aparte
  elemente van beide shapefiles vergelyk kan word.
  Mineralrights.shape<-readOGR(dsn = "E:/Raw Data/NC Shapefiles",layer =
  "Mineral Rights Clean")
  MineralrightsIndex <- Mineralrights.shape$GID
  NASAsolar.shape<-readOGR(dsn = "E:/Filtered Raw Data/Solar Data",layer =
  "GHI_NASA_NConly") NASAsolarIndex <- NASAsolar.shape$ID

  #Transformeer projeksies na standaard WGS84 projeksie en ellips (in
  proj4 string formaat)
  Mineralrights.shape@proj4string <- CRS("+proj=longlat +ellps=WGS84
  +towgs84=0,0,0,0,0,0,0 +no_defs")
  Mineralrights.shape <- spTransform(Mineralrights.shape, "+proj=longlat
  +ellps=WGS84 +towgs84=0,0,0,0,0,0,0 +no_defs")
  NASAsolar.shape <- spTransform(NASAsolar.shape, "+proj=longlat
  +ellps=WGS84
  +towgs84=0,0,0,0,0,0,0 +no_defs")
  #Versekering dat hulle identies is. gIntersects is baie picky hieroor
  projection(NASAsolar.shape) <- projection(Mineralrights.shape)

  #Skep plot as basis van visuele terugvoer om vordering te toon en skep
  error log plot(NASAsolar.shape)
  plot(Mineralrights.shape,add = TRUE)
  Errorlog <- data.frame(ID=character(),AreaSize=character())

  #Elke aparte polygon in die Mineralrights shapefile word om die beurt
  met elke aparte polygon in die NASA solar data shapefile vergelyk met
  gIntersects. As TRUE, word daardie spesifieke polygon in die NASA
  solar data gecrop tot die
  #buitelyne van die spesifieke Mineralrights area en dan gestoor in 'n
  aparte shapefile met die unieke ID as voorvoegsel.
  for (i in 1:length(MineralrightsIndex)){

    for (j in 1:length(NASAsolarIndex)){ tryCatch({ resultintersects <-
      gIntersects(Mineralrights.shape[Mineralrights.shape$GID ==
        MineralrightsIndex[i],],NASAsolar.shape[NASAsolar.shape$ID ==
```

```r
      NASAsolarIndex[j],],byid = TRUE)

      if (resultintersects){ clip <-
        intersect(NASAsolar.shape[NASAsolar.shape$ID ==
        NASAsolarIndex[j],],
        Mineralrights.shape[Mineralrights.shape$GID ==
        MineralrightsIndex[i],]) setwd(paste0("E:/Refined
        Data/",MineralrightsIndex[i],"/Solar"))
        writeOGR(obj = clip,dsn = getwd(),layer =
        paste0(MineralrightsIndex[i],"_",j),driver = "ESRI Shapefile",
        overwrite_layer = TRUE)
      }
    },error=function(e){ cat("ERROR :",conditionMessage(e), "\n")
      newerrorentry
      <data.frame(ID=as.character(NASAsolarIndex[j]),Errormessage=paste
      (conditionMessage(e))) Errorlog <- rbind(Errorlog,newerrorentry)
      }
    )
  }
}
setwd("E:/Filtered Raw Data/Solar Data") write.csv2(Errorlog,"Errorlog-
solar clipping",sep = ",")
```

113

A.5 Code Repository: Clipping of topographical data with mine extents - dealing with non-flat areas

```
library(sp)
library(rgdal)
library(raster)
library(rgeos)

#Implementering

  #Lees betrokke files en maak indekse
  #EPSG Projection 4148 - Hartebeesthoek94 is: "+proj=longlat +ellps=WGS84
  +towgs84=0,0,0,0,0,0,0 +no_defs "
  Mineralrights.shape<-readOGR(dsn = "E:/Raw Data/NC Shapefiles",layer =
  "Mineral Rights Clean")
  Mineralrights.shape@proj4string <- CRS("+proj=longlat +ellps=WGS84
  +towgs84=0,0,0,0,0,0,0 +no_defs")
  MineralrightsIndex <- Mineralrights.shape$GID setwd("E:/Filtered Raw
  Data/Topo Shapefiles")
  TopofileIndex <- list.files(pattern = ".shp",include.dirs =
  FALSE,recursive = FALSE)

  #Plots to have visual tracing en skep error log
  plot(Mineralrights.shape)
  ihavefound <- 0
  found <- 1
  Errorlog <- data.frame(ID=character(),Errormessage=character())

  for (i in 1:length(TopofileIndex)){
    Topofiletocheck <- readOGR(dsn = paste0(getwd()),layer =
    substring(TopofileIndex[i],1,(nchar(TopofileIndex[i])-4)))
    Topofiletocheck@proj4string <- CRS("+proj=longlat +ellps=WGS84
    +towgs84=0,0,0,0,0,0,0 +no_defs")

    #Create a less memory intensive version of Topofiletocheck to plot
    progress Temptopotoplot <-
    as(extent(Topofiletocheck@bbox),'SpatialPolygons')
    proj4string(Temptopotoplot) <- "+proj=longlat +ellps=WGS84
    +towgs84=0,0,0,0,0,0,0
    +no_defs"
    plot(Temptopotoplot,col = "red",add = TRUE)

      for (j in 1:length(MineralrightsIndex)){ tryCatch({

            resultvectorintersects <-
            gIntersects(Mineralrights.shape[Mineralrights.shape$GID
```

```r
                      == MineralrightsIndex[j],],Topofiletocheck,byid = TRUE)
                      resultvectorcrosses <-
                      gCrosses(Mineralrights.shape[Mineralrights.shape$GID ==
                      MineralrightsIndex[j],],Topofiletocheck,byid = TRUE)
                      resultvectoroverlaps <-
                      gOverlaps(Mineralrights.shape[Mineralrights.shape$GID ==
                      MineralrightsIndex[j],],Topofiletocheck,byid = TRUE)

                      if
                      (any(resultvectorintersects,resultvectorcrosses,resultvectorover
                      laps, na.rm = TRUE)){
                        ihavefound[found] <- MineralrightsIndex[j]
                        found <- found + 1
                        clip <-
                        crop(Topofiletocheck,Mineralrights.shape[Mineralrights.shape$
                        GID == MineralrightsIndex[j],])
                        ##Below old implementation was required because
                        rgeos::gIntersect did not output a dataframe, which is the
                        #required input for writeOGR. But using raster::crop solves
                        that problem and the problem of gIntersect not passing on the
                        attributes.
                        #clipdf <- data.frame(id = 1:length(clip))
                        #row.names(clipdf) <- row.names(clip) #clipdf<-
                        SpatialLinesDataFrame(clip,clipdf)
                        setwd(paste0("E:/Refined
                        Data/",MineralrightsIndex[j],"/Topo"))
                        writeOGR(obj = clip,dsn = getwd(),layer =
                        paste0(MineralrightsIndex[j],"_",found),driver = "ESRI
                        Shapefile",overwrite_layer = TRUE)

                      }
                  },error=function(e){cat("ERROR :",conditionMessage(e), "\n")
                        newerrorentry <-
                        data.frame(ID=as.character(MineralrightsIndex[j]),Errormessag
                        e=paste(conditionMess age(e)))
                        Errorlog <- rbind(Errorlog,newerrorentry)

                  })
          } setwd("E:/Filtered Raw Data/Topo Shapefiles")
    } setwd("E:/Filtered Raw Data/Topo Shapefiles")
write.csv2(Errorlog,"Errorlog-Topoclipping")
```

A.6 Code Repository: Clipping of topographical data with mine extents – dealing with flat areas

```r
library(sp)
library(rgdal)
library(raster)
library(rgeos)

#Implementering

#These shapes fall completely within and without touching any topo lines,
thus crop had no result to write.
#To write any sort of shapefile, to be used later to create DEM, plainly
write the outline polygon of the area.
exceptionGIDlist <-
c("1710355","1710358","1710361","1710412","1996632","2026792","2027403",
  "2027698","2027881","2030470","2030471","2030473","2030476","2032564",
  "2032565","2097674","2127546","2134300","2140434","2150908")
    #Lees betrokke files en maak indekse
    #EPSG Projection 4148 - Hartebeesthoek94 is: "+proj=longlat
    +ellps=WGS84
    +towgs84=0,0,0,0,0,0,0 +no_defs "
    Mineralrights.shape<-readOGR(dsn = "E:/Raw Data/NC Shapefiles",layer =
    "Mineral Rights Clean")
    Mineralrights.shape@proj4string <- CRS("+proj=longlat +ellps=WGS84
    +towgs84=0,0,0,0,0,0,0 +no_defs")
    MineralrightsIndex <- Mineralrights.shape$GID

    #Plots to have visual tracing en verander working directory
    plot(Mineralrights.shape) setwd(paste0("E:/Refined Data"))


    for (j in 1:length(MineralrightsIndex)){ if (MineralrightsIndex[j]
      %in% exceptionGIDlist){ setwd(paste0("E:/Refined Data/",
      MineralrightsIndex[j],"/Topo"))
    tempshapefile <- Mineralrights.shape[Mineralrights.shape$GID ==
    MineralrightsIndex[j],]

      #To create a DEM later, a "HEIGHT" var is required in the
      dataframe. This is a short cheat to recycle an existing unused
      var, rename it and populate with a dummy height.
      names(tempshapefile@data) <-
      c("GID","EASE_TYPE","GEOM_AREA","HEIGHT","TAG_X","TAG_Y","TAG_VAL
      UE","TAG_SIZE","TAG_A NGLE","TAG_JUST","DATE_STAMP")
      tempshapefile@data$HEIGHT <- 1000
      writeOGR(obj = tempshapefile,dsn = getwd(),layer =
      paste0(MineralrightsIndex[j],"_",j),driver = "ESRI
      Shapefile",overwrite_layer = TRUE)
```

```r
        plot(Mineralrights.shape[Mineralrights.shape$GID ==
        MineralrightsIndex[j],], col = "red", add = TRUE)

} else { plot(Mineralrights.shape[Mineralrights.shape$GID ==
MineralrightsIndex[j],], col = "blue", add = TRUE)
        }

}
```

117

```r
        plot(Mineralrights.shape[Mineralrights.shape$GID ==
        MineralrightsIndex[j],], col = "red", add = TRUE)

} else { plot(Mineralrights.shape[Mineralrights.shape$GID ==
MineralrightsIndex[j],], col = "blue", add = TRUE)
```

A.7 Code Repository: Clipping of aerial photographs with mine extents

```r
library(sp)
library(rgdal)
library(raster)
library(rgeos)

#Implementering

  #Lees betrokke files, maak indekse en transformeer na standaard WGS84
  projeksie en ellips (in proj4 string formaat)
  Mineralrights.shape <-readOGR(dsn = "E:/Raw data/NC Shapefiles",layer =
  "Mineral Rights Clean")
  Mineralrights.shape@proj4string <- CRS("+proj=longlat +ellps=WGS84
  +towgs84=0,0,0,0,0,0,0 +no_defs")
  MineralrightsIndex <- Mineralrights.shape$GID
  ##Mineralrights.shape <- spTransform(Mineralrights.shape,
  CRS("+proj=longlat +ellps=WGS84
  +towgs84=0,0,0,0,0,0,0 +no_defs"))

  setwd("E:/Filtered Raw Data/Aerial")
  AerialfileIndex <- list.files(pattern = ".tif",include.dirs =
  FALSE,recursive = FALSE)

  #plot purely for tracking progress and create error empty error log
  plot(Mineralrights.shape)
  Errorlog <- data.frame(ID=character(),AreaSize=character())

  #Elke aparte polygon in die Mineralrights shapefile word om die beurt
  met elke aparte lugfoto vergelyk met gIntersects. As TRUE, word daardie
  spesifieke lugfoto (raster) gecrop tot die buitelyne van die spesifieke
  Mineralrights area en dan gestoor in 'n aparte raster (GeoTIFF) met die
  unieke ID as voorvoegsel.
  #Dis belangrik om die lugfotos te laai in die buitenste lus en nie die
  binneste geneste lus nie, weens die tyd wat dit neem om een lugfoto te
  laai in geheue.
  for (i in 1:length(AerialfileIndex)){
    Aerialfiletocheck <- stack(paste0(getwd(),"/",AerialfileIndex[i]))
    #Makliker om polygons met polygons te vergelyk (en minder berekening
    intensief) raam <- projectExtent(Aerialfiletocheck,"+proj=longlat
    +ellps=WGS84
    +towgs84=0,0,0,0,0,0,0 +no_defs")
    raam.shape <- as(raam@extent, 'SpatialPolygons')

    #toets elke shape in Mineralrights om te kyk of hy met die huidige
    lugfoto intersect. for (j in 1:length(MineralrightsIndex)){ tempshape
    <- Mineralrights.shape[Mineralrights.shape$GID ==
    MineralrightsIndex[j],] tryCatch({ resultintersects <-
    gIntersects(tempshape,raam.shape,byid = TRUE)
```

```r
        },error=function(e){cat("ERROR :",conditionMessage(e), "\n")
            newerrorentry
            <data.frame(ID=as.character(rasterfilelist[j]),Errormessage=pas
            te(conditionMessage(e) ))
            Errorlog <- rbind(Errorlog,newerrorentry) }
        )

        #wanneer in interseksie gevind word, clip die lugfoto met die area
        wat hom intersect (in geheue), verander na die toepoaslike working
        directory en stoor die clipped raster in GTIFF formaat met die
        naam wat begin met die area se GID gevolg deur die oorspronlike
        lugfoto nommer (minus extension).
        if (any(resultintersects,na.rm = TRUE)){
          #plot die area wat tans True resultaat gelewer het in rooi bo-oor
          die huidige plot van al die areas.
          plot(tempshape,col="red",add=T)

          #transformeer die MineralRights area wat tans ondersoek word in
          dieselfde coordinaat stelsel as die huidige lugfoto om vals
          negatiewe uit te skakel. tempshape <-
          spTransform(tempshape,Aerialfiletocheck@crs)

          #Hierdie clipping vat baie berekening krag en tyd.
          clip <- raster::intersect(Aerialfiletocheck,tempshape)
          setwd(paste0("E:/Refined Data/",MineralrightsIndex[j],"/Aerial"))
          writeRaster(clip,filename=paste0(MineralrightsIndex[j],"_",
substring(AerialfileIndex[i],1,(nchar(AerialfileIndex[i])-4)),"-
clipped","_",j),format
          ="GTiff",TFW=YES,options="INTERLEAVE=BAND",overwrite=T)
          setwd("E:/Filtered Raw Data/Aerial")

      }
    }

  }

  #Skryf error log
setwd("E:/Filtered Raw Data/Aerial") write.csv2(Errorlog,"Errorlog-Aerial
clipping",sep = ",")
```

119

A.8 Code Repository: Calculating the brut solar insolation within each mine extent

```r
library(rgdal)
library(rgeos)
library(maptools)
library(dplyr)

#Belangrike inligting: Wanneer areas bereken word, moet die betrokke area
eers na 'n equal area projection transformer word. Dis omrede tipiese
projeksies optimaal is om visueel herkenbaar te wees (veral op planeet
skaal) maar lei tot onakurate berekening van area. Hoe kleiner die area,
hoe minder van 'n verskil maak dit. #SR-ORG:8550 South African Albers NGI
aerials is die gekose equal area projection.

  #These shapes fall completely within another shape and should thus be
  excluded to avoid including them again in results.
  exceptionGIDlist <-
  c("1710350","1710362","1710363","1710364","1710365","1710366","1710367","
  1710368","1710369","1710370","1710373","1710374","1710375","1710383",
  "1710384","1710385","1710386","1710393","1710395","1710404","1710409",
  "1710410","2018956","2021033","2039570","2039572","2039575","2043532",
  "2043584","2097663","2097665","2097667","2097671","2150912","2151243",
  "2151687" )
  #Lees betrokke files, maak indekse en skep lee dataframe om resultaat in
  te stoor.
  Mineralrights.shape<-readOGR(dsn = "E:/Raw data/NC Shapefiles",layer =
  "Mineral Rights Clean")
  Mineralrights.shape@proj4string <- CRS("+proj=longlat +ellps=WGS84
  +towgs84=0,0,0,0,0,0,0 +no_defs")
  MineralrightsIndex <- Mineralrights.shape$GID
  Resultdf <-
  data.frame(ID=character(),AreaSize=double(),Insolation=double(),kWhperday
  =double())

  #This is purely for watching progress. Skep error log en randwaardes.
  plot(Mineralrights.shape)
  Errorlog <- data.frame(ID=character(),AreaSize=character())
  GlobalIndex <- 1

  #Itereer deur elke area in Mineralrights, skep 'n lys van die NASA
  clipped shapefiles in die area se betrokke
  #directory structure en bereken dan daardie area se totale grootte.
  for (i in 1:length(MineralrightsIndex)){ if (!(MineralrightsIndex[i]
    %in% exceptionGIDlist)){
      #Maak 'n lys van al die relevante shapefiles (gewoonlik net een)
      setwd(paste0("E:/Refined Data/",MineralrightsIndex[i],"/Solar"))
```

```r
ShapefileIndex <- list.files(pattern = ".shp",include.dirs =
FALSE,recursive = FALSE)

#Belangrik: Hierdie for lus gaan nie die totaal vir elke area
uitwerk as daar meer as een NASA shapefile oorvleul
#met 'n area nie, daar gaan bloot net twee inskrywings met dieselfde
GID wees en verskillende area groottes.
for (j in 1: length(ShapefileIndex)){
  ##Ou implemetering van if. Ingewikkeld omdat elke area se NASA
    solar shapefiles nie apart in subdirectories gestoor was nie.
    Nuwe implementering met subdirs elimineer dit en vereenvoudig if.
    if (as.character(MineralrightsIndex[i]) ==
    unlist(strsplit(ShapefileIndex[j]," _ "))[1]){
      tryCatch({
        InvestigatedArea <- readOGR(dsn = paste0(getwd()),layer =
          substring(ShapefileIndex[j],1,(nchar(ShapefileIndex[j])-4)))
        InvestigatedArea@proj4string <- CRS("+proj=longlat
          +ellps=WGS84
          +towgs84=0,0,0,0,0,0,0 +no_defs")
        InvestigatedArea <- spTransform(InvestigatedArea, "+proj=aea
          +lat_1=24
          +lat_2=-33 +lat_0=0 +lon_0=24 +x_0=0 +y_0=0 +ellps=WGS84
          +units=m +no_defs ")

        #Gebruik gArea ipv Geosphere::areaPolygon omdat gArea reeds
          die "holes" se area aftrek
        Arearesult <- gArea(InvestigatedArea) kWhperday <-
          InvestigatedArea$annual*Arearesult

        #Skep nuwe dataframe inskrywing en stoor in die globale
          dataframe (rbind)
        newentry <-
          data.frame(ID=as.character(MineralrightsIndex[i]),AreaSize=Ar
          earesult, Insolation = InvestigatedArea$annual,
          kWhperday=kWhperday)
        Resultdf <- rbind(Resultdf,newentry)
        GlobalIndex <- GlobalIndex+1
        #Again, for watching progress
        plot(InvestigatedArea, col = "red", main = ShapefileIndex[j])
      },error=function(e){cat("ERROR :",conditionMessage(e), "\n")})

    }
  }
}
#Sorteer en stoor globale resultaat dataframe as 'n CSV
setwd("E:/Refined Data") Resultdf <- arrange(Resultdf,ID)
write.csv2(Resultdf, file = "Solar Insolation per area.csv")
#Skryf error log
write.csv2(Errorlog,"Errorlog-solar per area",sep = ",")
```

121

A.9 Code Repository: Convert topographical data to point data, interpolate to create DEM and calculate aspect and slope

```r
library(rgdal)
library(raster)
library(gstat)

#Implementering

  #Lees betrokke files, maak indekse. Geen transformasie nodig hierdie
  keer want dit is slegs om die region IDs te kry.
  Mineralrights.shape <-readOGR(dsn = "E:/Raw data/NC Shapefiles",layer =
  "Mineral Rights Clean")
  Mineralrights.shape@proj4string <- CRS("+proj=longlat +ellps=WGS84
  +towgs84=0,0,0,0,0,0,0 +no_defs")
  MineralrightsIndex <- Mineralrights.shape$GID
  for(i in 1:length(MineralrightsIndex)){
    setwd(paste0("E:/Refined Data/",
    as.character(MineralrightsIndex[i]),"/Topo")) TopofileIndex <-
    list.files(pattern = ".shp",include.dirs = FALSE,recursive = FALSE)

    #Net n check om te sien of die huidige folder wel topofiles het, om
    onnodige error te voorkom
    if (length(TopofileIndex) > 0){
        #klein kans dat daar meer as een clipped topo shapefile per
        region is
        for (j in 1:length(TopofileIndex)){

          #Lees topo data in
          Topofiletocheck <- readOGR(dsn = getwd(),layer =
          substring(TopofileIndex[j],1,(nchar(TopofileIndex[j])-4)))
          Topofiletocheck@proj4string <- CRS("+proj=longlat +ellps=WGS84
          +towgs84=0,0,0,0,0,0,0 +no_defs")

          #Skep lee Raster file. Gebruik die topofile se bounding box
          dimensies as die xlimiet en ylimiet en erf sy CRS. Die
          resolusie moet onverklaarbaar in 'n aparte roep gemaak word.
          Dit word nie spesifiseer as die hoeveelheid punte per ry/kolom
          nie, maar as die afstand wat elke punt moet verteenwoordig.
          Let op dat "afstand" in dieselfde eenheid is as waarin die
          raster definieer is m.a.w. grade.
          Resultaat.raster <- raster(xmn = Topofiletocheck@bbox[1,1],xmx
          =
          Topofiletocheck@bbox[1,2],ymn =
          Topofiletocheck@bbox[2,1],ymx =
          Topofiletocheck@bbox[2,2],crs = Topofiletocheck@proj4string)
          res(Resultaat.raster) <- 0.0001
```

```r
#Omskep kontoer lyne na Raster punte. Die plat areas wat deur
die TopoClip as uitsonderings hanteer is moet eers van
SpatialPolygonDataFrame na SpatialLinesDataFrame coerce word.
Dit kan nie direk van Polygons na points nie.
if (class(Topofiletocheck)[1] == "SpatialPolygonsDataFrame"){
  Topofiletocheck <-
  as(Topofiletocheck,'SpatialLinesDataFrame')
  Topopoints <- as(Topofiletocheck,'SpatialPointsDataFrame')
}
else {
  Topopoints <- as(Topofiletocheck,'SpatialPointsDataFrame')
}

#Skep interpolasie model. "HEIGHT" in Topopoints is die
bindende veranderlike. nmax is die maksimum omliggende punte
wat gebruik word om die ontbrekende interpolasie punt te
bepaal.
Interpolasiemodel <- gstat(id = "HEIGHT", formula =
HEIGHT~1,data = Topopoints, nmax = 7, set = list(idp=2))


#Skep interpoleerde DEM met raster:interpolate wat 'n
interpolasie funksie is wat spesifiek XY data verwag met 'n
gstat model. Vreemde eienskap is dat Resultaat.raster nooit
gevul word nie, maar bloot as 'n template gebruik word oor hoe
om die resultaat te stoor. Hierdie is 'n verskriklike hulpbron
rowende berekening wat #etlike ure per region neem. interpDEM
<- interpolate(Resultaat.raster,Interpolasiemodel)

#Bereken slope en aspect. Die neighbors parameter bepaal watse
algoritme gebruik word.
AspectSlope <- terrain(interpDEM, opt =
c('slope','aspect'),unit = 'degrees',neighbors = 8)

#Skryf Rasters na file
setwd(paste0("E:/Refined Data/",
as.character(MineralrightsIndex[i]),"/Aspect Slope"))
writeRaster(AspectSlope,
paste0(as.character(MineralrightsIndex[i]),"_AspectSlope",j,".g
rd")) setwd(paste0("E:/Refined Data/",
as.character(MineralrightsIndex[i]),"/DEM"))
writeRaster(interpDEM,
paste0(as.character(MineralrightsIndex[i]),"_DEM",j,".grd"))
setwd(paste0("E:/Refined Data/",
as.character(MineralrightsIndex[i]),"/Topo"))
    }
  }
}
```

A.10 Code Repository: Looping methods to create user specified polygons on aerial photographs to mark exclusions – Small areas

```r
library(raster)
library(rgdal)

#Onthou om Aerialprogress.txt te skep voor eerste run

#Implementation

  #These shapes fall completely within another shape and should thus be
  excluded to avoid including them again in results.
  exceptionGIDlist <-
  c("1710350","1710362","1710363","1710364","1710365","1710366","1710367","
  1710368"
    ,"1710369","1710370","1710373","1710374","1710375","1710383","1710384"
    ,"1710385","1710386","1710393","1710395","1710404","1710409","1710410"
    ,"2018956","2021033","2039570","2039572","2039575","2043532","2043584"
    ,"2097663","2097665","2097667","2097671","2150912","2151243","2151687"
)

  #Die interne for loop indeks word extern gestoor, wat dit moontlik maak
  om 1 of 2 of 10 analieses te doen, te stop en vorderring te stoor (die
  indeks) en weer die script te run en hervat waar jy opgehou het.
  setwd("E:/Filtered Raw Data/Aerial")
  globalindex <- read.table("Aerialprogressindex.txt", header = FALSE)
  globalindex <- globalindex[1,1]

  Mineralrights.shape <- readOGR(dsn = "E:/Raw data/NC Shapefiles",layer =
  "Mineral Rights Clean")
  MineralrightsIndex <- Mineralrights.shape$GID
  #Mineralrights.shape <- spTransform(Mineralrights.shape,)

  #Lus wat deur elke area van Mineralrights itereer. By elke iterasie word
  al die clipped lugfotos wat voorheen vir die betrokke area gemaak en
  gestoor is opgeroep. Alles word dan op dieselfde plek geplot.
  for(i in globalindex:length(MineralrightsIndex)){
    setwd(paste0("E:/Refined Data/",MineralrightsIndex[i],"/Aerial"))
AerialfileIndex <- list.files(pattern = ".tif",include.dirs =
FALSE,recursive = FALSE)

    #Net n check om te sien of die huidige folder wel Aerialfiles het, om
    onnodige error te voorkom
    if (length(AerialfileIndex) > 0){
      if (!(MineralrightsIndex[i] %in% exceptionGIDlist)){

          #Die eerste lugfoto word gebruik om die koordinaat stelsel van
          die area te transformeer en te plot
```

```r
Aerialfile <- stack(paste0(getwd(),"/",AerialfileIndex[1]))
CurrentRegion <- Mineralrights.shape[Mineralrights.shape$GID ==
MineralrightsIndex[i],]
CurrentRegion@proj4string <- CRS("+proj=longlat +ellps=WGS84
+towgs84=0,0,0,0,0,0,0 +no_defs")
CurrentRegion <- spTransform(CurrentRegion,Aerialfile@crs)
print(paste("Currently drawing polygons for:
",MineralrightsIndex[i]))

#Plot eers die polygon, al word hy overwrite, om die volle
extent van die plot venster te gebruik.
##dev.new(width = 15, height = 10,noRStudioGD = TRUE)
plot(CurrentRegion)

#Al die relevante lugfoto(s) word nou geplot op dieselfde plot.
for(j in 1:length(AerialfileIndex)){
  Aerialfile <-
  stack(paste0(getwd(),"/",AerialfileIndex[j]))
  plotRGB(Aerialfile,add=T)
}
plot(CurrentRegion,col = "transparent",add = T)

#set rand parameters
continue <- TRUE exclusionindex <- 1

#Gebruiker skep polygons met muis-klieke bo-op die lugfotos.
Die while loop laat toe dat meer as een polygon #geskep kan
word.
while(continue == TRUE){

  newpolygon <- drawPoly(sp = TRUE, col = "red")
  newpolygon@proj4string <- Aerialfile@crs
  newpolygon <- spTransform(newpolygon,"+proj=longlat
  +ellps=WGS84
  +towgs84=0,0,0,0,0,0,0 +no_defs")

  #Mouse-click extracted polygon's class is SpatialPolygons.
  Must be converted to SpatialPolygonsDataFrame before writeOGR
  can be used to write shapefile.
  newdf <- data.frame(ID = 1:length(newpolygon))
  row.names(newdf) <- row.names(newpolygon)
  newdf<-SpatialPolygonsDataFrame(newpolygon,newdf)
  writeOGR(obj = newdf,dsn = paste0("E:/Refined
  Data/",MineralrightsIndex[i],"/Exclusion Objects"),layer =
  paste0(MineralrightsIndex[i],"_Exclusion_",exclusionindex),"
  ESRI Shapefile",overwrite_layer = TRUE)

  #Code that requires user input leading to a T/F condition to
  continue. exclusionindex is only to use for a #file naming
  incremental suffix.
```

```
      continue <- as.logical(toupper(readline(prompt = "Do you
      want to draw another polygon? (T/F): "))) exclusionindex <-
      exclusionindex+1
   }
   dev.off()

   #set rand parameter continue <- TRUE

   #Kyk of die proses nie natuurlik tot by sy einde aangekom het
   nie (laaste area inskrywing in Mineralrights) en puls dan
   gebruiker of daar na die volgende area aanbeweeg moet word.
   if(i != length(MineralrightsIndex)){ continue <-
      as.logical(toupper(readline(prompt = "Do you want to
      continue to the next region? (T/F): ")))

      if(continue != TRUE){
         ##save globalindex+1 to txt file overwrite ##break out of
         this if and the current for loop setwd("E:/Filtered Raw
         Data/Aerial") fileconnection <-
         file("Aerialprogressindex.txt")
         writeLines(as.character(i+1),fileconnection)
         close(fileconnection) break() }
      #Hierdie else gaan dalk moeilikheid met die for indeks
      veroorsaak.
      else{
         globalindex <- i + 1
      }
   }
  }
 }
}
```

A.11 Code Repository: Looping methods to create user specified polygons on aerial photographs to mark exclusions – Large areas

```r
library(raster)
library(rgdal)

#Onthou om Aerialprogress.txt te skep voor eerste run

#Implementation

  #These shapes fall completely within another shape and should thus be
  excluded to avoid including them again in results.
  exceptionGIDlist <-
  c("1710350","1710362","1710363","1710364","1710365","1710366","1710367","
  1710368","1710369","1710370","1710373","1710374","1710375","1710383"
  ,"1710384","1710385","1710386","1710393","1710395","1710404","1710409"
  ,"1710410","2018956","2021033","2039570","2039572","2039575","2043532"
  ,"2043584","2097663","2097665","2097667","2097671","2150912","2151243"
  ,"2151687")
  #These are a list of the large areas that are the only ones that will be
  included. inclusionGIDlist <-
  c("1710390","1710371","2004518","1993763","1993798","1994010","1991064"
    ,"2097662","2097666","2097670","2113212","2113954","2137726","2146870"
    ,"2151240","2151242","2151674","2151675","2151676","2151686")

  #Die interne for loop indeks word extern gestoor, wat dit moontlik maak
  om 1 of 2 of 10 analieses te doen, te stop en vorderring te stoor (die
  indeks) en weer die script te run en hervat waar jy opgehou het.
  setwd("E:/Filtered Raw Data/Aerial")
  globalindex <- read.table("Aerialprogressindex.txt", header = FALSE)
  globalindex <- globalindex[1,1]

  Mineralrights.shape <- readOGR(dsn = "E:/Raw data/NC Shapefiles",layer =
  "Mineral Rights Clean")
  MineralrightsIndex <- Mineralrights.shape$GID
  #Mineralrights.shape <- spTransform(Mineralrights.shape,)

  #Lus wat deur elke area van Mineralrights itereer. By elke iterasie word
  al die clipped lugfotos wat voorheen vir die betrokke area gemaak en
  gestoor is opgeroep. Alles word dan op dieselfde plek geplot.
  for(i in globalindex:length(MineralrightsIndex)){
    setwd(paste0("E:/Refined Data/",MineralrightsIndex[i],"/Aerial"))
    AerialfileIndex <- list.files(pattern = ".tif",include.dirs =
    FALSE,recursive = FALSE)

    #Net n check om te sien of die huidige folder wel Aerialfiles het, om
    onnodige error te voorkom
```

```r
if (length(AerialfileIndex) > 0){ if ((MineralrightsIndex[i] %in%
   inclusionGIDlist)){

    #Die eerste lugfoto word gebruik om die koordinaat stelsel van
    die area te transformeer en te plot
    Aerialfile <- stack(paste0(getwd(),"/",AerialfileIndex[1]))
    CurrentRegion <- Mineralrights.shape[Mineralrights.shape$GID ==
    MineralrightsIndex[i],]
    CurrentRegion@proj4string <- CRS("+proj=longlat +ellps=WGS84
    +towgs84=0,0,0,0,0,0,0 +no_defs")
    CurrentRegion <- spTransform(CurrentRegion,Aerialfile@crs)
    print(paste("Currently drawing polygons for:
    ",MineralrightsIndex[i]))

    #Al die relevante lugfoto(s) word nou geplot op dieselfde plot.
    for(j in 1:length(AerialfileIndex)){

      #Plot eers die image, om die volle extent van die plot
      venster te gebruik.
      Aerialfile <- stack(paste0(getwd(),"/",AerialfileIndex[j]))
      plotRGB(Aerialfile)

      #Region gaan waarskynlik buite die skerm extents gaan, maar
      dit is presies die doelwit.
      plot(CurrentRegion,col = "transparent",add = T)

      #set rand parameters
      continue <- TRUE exclusionindex <- 1

      #Gebruiker skep polygons met muis-klieke bo-op die lugfotos.
      Die while loop laat toe dat meer as een polygon geskep kan
      word.
      while(continue == TRUE){
        newpolygon <- drawPoly(sp = TRUE, col = "red")
        newpolygon@proj4string <- Aerialfile@crs
        newpolygon <- spTransform(newpolygon,"+proj=longlat
        +ellps=WGS84
        +towgs84=0,0,0,0,0,0,0 +no_defs")

        #Mouse-click extracted polygon's class is SpatialPolygons.
        Must be converted to SpatialPolygonsDataFrame before writeOGR
        can be used to write shapefile.
        newdf <- data.frame(ID = 1:length(newpolygon))
        row.names(newdf) <- row.names(newpolygon)
        newdf<-SpatialPolygonsDataFrame(newpolygon,newdf)
        writeOGR(obj = newdf,dsn = paste0("E:/Refined
        Data/",MineralrightsIndex[i],"/Exclusion Objects"),layer =
```

128

```
      paste0(MineralrightsIndex[i],"_Exclusion_",j,"_",exclusioninde
      x),"ESRI Shapefile",overwrite_layer = TRUE)

      #Code that requires user input leading to a T/F condition to
      continue.
      exclusionindex is only to use for a #file naming
      incremental suffix.
      continue <- as.logical(toupper(readline(prompt = "Do
      you want to draw another polygon? (T/F): ")))
      exclusionindex <- exclusionindex+1
    } dev.off()
  }


    #set rand parameter
    continue <- TRUE

    #Kyk of die proses nie natuurlik tot by sy einde aangekom het nie
    (laaste area inskrywing in Mineralrights) en puls dan gebruiker
    of daar na die volgende area aanbeweeg moet word.
    if(i != length(MineralrightsIndex)){ continue <-
      as.logical(toupper(readline(prompt = "Do you want to continue
      to the next region? (T/F): ")))

      if(continue != TRUE){
          ##save globalindex+1 to txt file overwrite ##break out of
      this if and the current for loop
      setwd("E:/Filtered Raw Data/Aerial")
      fileconnection <- file("Aerialprogressindex.txt")
      writeLines(as.character(i+1),fileconnection)
      close(fileconnection) break() }
      #Hierdie else gaan dalk moeilikheid met die for indeks
      veroorsaak.
      else{
        globalindex <- i + 1
      }
    }
  }
  }
}
```

A.12 Code Repository: Calculating the solar insolation contained within each exception polygon

```r
library(rgeos)
library(rgdal)
library(dplyr)
library(raster)

#Implementation

  #Lees betrokke files, maak indekse. Geen transformasie nodig hierdie
  keer want dit is slegs om die region IDs te kry.
  Mineralrights.shape <-readOGR(dsn = "E:/Raw data/NC Shapefiles",layer =
  "Mineral Rights Clean")
  Mineralrights.shape@proj4string <- CRS("+proj=longlat +ellps=WGS84
  +towgs84=0,0,0,0,0,0,0 +no_defs")
  Mineralrights.shape <- spTransform(Mineralrights.shape, "+proj=aea
  +lat_1=24 +lat_2=-33
  +lat_0=0 +lon_0=24 +x_0=0 +y_0=0 +ellps=WGS84 +units=m +no_defs ")
  MineralrightsIndex <- Mineralrights.shape$GID

  #These shapes fall completely within another shape and should thus be
  excluded to avoid including them again in results.
  exceptionGIDlist <-
  c("1710350","1710362","1710363","1710364","1710365","1710366","1710367","
  1710368","1710369","1710370","1710373","1710374","1710375","1710383",
  "1710384","1710385","1710386","1710393","1710395","1710404","1710409",
  "1710410","2018956","2021033","2039570","2039572","2039575","2097663",
  "2097665","2097667","2097671","2150912","2151243","2151687" )
  #Dataframe waarin die resultate gestoor sal word
  Resultdf <-
  data.frame(ID=character(),ExclusionID=character(),AreaSize=double(),Insol
  ation=double(),kWhp erday=double())
  GlobalIndex <- 1

  #Itereer deur elke GID en werk uit wat die grootte van elke exclusion
  area is en gevolglik die kWh vir elke area wat afgetrek moet word van
  die totale area.
  for (i in 1:length(MineralrightsIndex)){ print(paste0("Currently
    calculating exclusion areas for region ",i,":
    ",MineralrightsIndex[i]))

    #Keer die GIDs van kleiner areas wat binne in groter areas bevat is.
    Voorkom dus dat die waarde twee keer #ingesluit word.
    if (!(MineralrightsIndex[i] %in% exceptionGIDlist)){
```

```r
#Maak 'n lys van al die exclusion shapefile areas binne die
huidige GID setwd(paste0("E:/Refined
Data/",MineralrightsIndex[i],"/Exclusion Objects"))
ExclusionIndex <- list.files(pattern = ".shp",include.dirs =
FALSE,recursive = FALSE)

#Maak 'n lys van al die NASA solar data shapefile areas binne die
huidige GID, indien meer as 1.
setwd(paste0("E:/Refined Data/",MineralrightsIndex[i],"/Solar"))
SolarIndex <- list.files(pattern = ".shp",include.dirs =
FALSE,recursive = FALSE)

#Eerste (ingewikkelde) geval: Area wat deur 2 NASA shapefiles
oorvleuel word if (length(SolarIndex) > 1){
#Itereer deur die NASA shapefiles
for (j in 1:length(SolarIndex)){ setwd(paste0("E:/Refined Data/",
MineralrightsIndex[i],"/Solar"))
NASAArea <- readOGR(dsn = getwd(),layer =
    substring(SolarIndex[j],1,(nchar(SolarIndex[j])-4)),verbose =
    FALSE)
    NASAArea@proj4string <- CRS("+proj=longlat +ellps=WGS84
    +towgs84=0,0,0,0,0,0,0 +no_defs")
    NASAArea <- spTransform(NASAArea, "+proj=aea +lat_1=24 +lat_2=-
    33 +lat_0=0
    +lon_0=24 +x_0=0 +y_0=0 +ellps=WGS84 +units=m +no_defs ")

    #Maak net seker dat daar wel exclusions is
    if (length(ExclusionIndex > 0)){
      #Gaan deur lys van exclusions en kyk of dit binne die huide
      NASA shapefile is.
      for (k in 1:length(ExclusionIndex)){
        setwd(paste0("E:/Refined
        Data/",MineralrightsIndex[i],"/Exclusion Objects"))
        ExclusionPoly <- readOGR(dsn = getwd(),layer =
        substring(ExclusionIndex[k],1,(nchar(ExclusionIndex[k])-
        4)),verbose = FALSE)
        ExclusionPoly@proj4string <- CRS("+proj=longlat
        +ellps=WGS84
        +towgs84=0,0,0,0,0,0,0 +no_defs")
        ExclusionPoly <- spTransform(ExclusionPoly, "+proj=aea
        +lat_1=24 +lat_2=-33
        +lat_0=0 +lon_0=24 +x_0=0 +y_0=0 +ellps=WGS84 +units=m
        +no_defs ")

        tryCatch({
        resultoverlaps <-
        gOverlaps(NASAArea,ExclusionPoly,checkValidity = TRUE)
```

131

```r
        resultcontains <-
        gContains(NASAArea,ExclusionPoly,checkValidity = TRUE)
        },error=function(e){cat("ERROR :",conditionMessage(e),
        "\n")})

        #Die exclusion area is wel binne die huidige NASA area
        if (resultcontains || resultoverlaps){ tryCatch({
            ExclusionArea <- gArea(ExclusionPoly)
          },error=function(e){cat("ERROR :",conditionMessage(e),
          "\n")}) kWhperday <- NASAArea$annual*ExclusionArea

          #Skep nuwe dataframe inskrywing en stoor in die globale
          dataframe (rbind)  newentry <-
          data.frame(ID=as.character(MineralrightsIndex[i]),Exclusi
          onID=
          as.character(ExclusionIndex[k]),AreaSize=ExclusionArea,In
          solation=
          NASAArea$annual,kWhperday=kW hperday)
          Resultdf <- rbind(Resultdf,newentry)
          GlobalIndex <- GlobalIndex+1
          ExclusionArea <- 0
        }
      }
    }
  }
}
#Tweede (eenvoudige) geval: area oorvleuel deur net 1 NASA
shapefile
else {
setwd(paste0("E:/Refined Data/",MineralrightsIndex[i],"/Solar"))
NASAArea <- readOGR(dsn = getwd(),layer =
  substring(SolarIndex[1],1,(nchar(SolarIndex[1])-4)))
  NASAArea@proj4string <- CRS("+proj=longlat +ellps=WGS84
  +towgs84=0,0,0,0,0,0,0 +no_defs")
  NASAArea <- spTransform(NASAArea, "+proj=aea +lat_1=24 +lat_2=-33
  +lat_0=0 +lon_0=24 +x_0=0 +y_0=0 +ellps=WGS84 +units=m +no_defs ")

  #Maak net seker dat daar wel exclusions is
  if (length(ExclusionIndex > 0)){
    for (k in 1:length(ExclusionIndex)){ setwd(paste0("E:/Refined
      Data/",MineralrightsIndex[i],"/Exclusion Objects"))
      ExclusionPoly <- readOGR(dsn = getwd(),layer =
      substring(ExclusionIndex[k],1,(nchar(ExclusionIndex[k])-
      4)),verbose = FALSE)
      ExclusionPoly@proj4string <- CRS("+proj=longlat +ellps=WGS84
      +towgs84=0,0,0,0,0,0,0 +no_defs")
```

132

```r
        ExclusionPoly <- spTransform(ExclusionPoly, "+proj=aea
        +lat_1=24 +lat_2=-33
        +lat_0=0 +lon_0=24 +x_0=0 +y_0=0 +ellps=WGS84 +units=m
        +no_defs ")

        #This is to find the single point polygons that causes errors
        tryCatch({
          ExclusionArea <- gArea(ExclusionPoly)
        },error=function(e){cat("ERROR :",conditionMessage(e),

        "\n")}) kWhperday <- NASAArea$annual*ExclusionArea

        #Skep nuwe dataframe inskrywing en stoor in die globale
        dataframe (rbind)
        newentry <-
        data.frame(ID=as.character(MineralrightsIndex[i]),
        ExclusionID=as.character(Exclusi onIndex[k]), AreaSize=
        ExclusionArea,
        Insolation=NASAArea$annual,kWhperday=kWhperday)
        Resultdf <- rbind(Resultdf,newentry)
        GlobalIndex <- GlobalIndex+1
        #Easy way to spot the erronious single point polygons in
        the result data later
        ExclusionArea <- 0

      }
    }
  }
  }
}

#Sorteer en stoor globale resultaat dataframe as 'n CSV
setwd("E:/Refined Data")
Resultdf <- arrange(Resultdf,ID)
write.csv2(Resultdf, file = "Solar Insolation of exclusions per
area.csv")
```

133

A.13 Code Repository: Calculating the solar insolation corresponding to the slope and aspect exclusion criteria

```r
library(rgdal)
library(raster)
library(rgeos)
library(dplyr)

#Implementering

  #Lees betrokke files, maak indekse. Geen transformasie nodig hierdie
  keer want dit is slegs om die region IDs te kry.
  Mineralrights.shape <-readOGR(dsn = "E:/Raw data/NC Shapefiles",layer =
  "Mineral Rights Clean")
  Mineralrights.shape@proj4string <- CRS("+proj=longlat +ellps=WGS84
  +towgs84=0,0,0,0,0,0,0 +no_defs")
  MineralrightsIndex <- Mineralrights.shape$GID

  #These shapes fall completely within another shape and should thus be
  excluded to avoid including them again in results.
  exceptionGIDlist <-
  c("1710350","1710362","1710363","1710364","1710365","1710366","1710367",
  "1710368","1710369","1710370","1710373","1710374","1710375","1710383",
  "1710384","1710385","1710386","1710393","1710395","1710404","1710409",
  "1710410","2018956","2021033","2039570","2039572","2039575","2097663",
  "2097665","2097667","2097671","2150912","2151243","2151687" )
  #Dataframe waarin die resultate gestoor sal word
  Resultdf <- data.frame(ID=character(),AreaSize=double())
  GlobalIndex <- 1

  #Itereer deur elke GID en werk uit wat die grootte van elke exclusion
  area is #en gevolglik die area wat afgetrek moet word van die totale
  area.
  for (i in 1:length(MineralrightsIndex)){ print(paste0("Currently
    calculating exclusion areas for region: ",MineralrightsIndex[i]))

    #Keer die GIDs van kleiner areas wat binne in groter areas bevat is.
    Voorkom dus dat die waarde twee keer #ingesluit word.
    if (!(MineralrightsIndex[i] %in% exceptionGIDlist)){

      #Maak 'n lys van al die Aspect/Slope files binne die huidige GID
      setwd(paste0("E:/Refined Data/",MineralrightsIndex[i],"/Aspect
      Slope"))
      AspectSlopeIndex <- list.files(pattern = ".grd",include.dirs =
      FALSE,recursive = FALSE)

      if (length(AspectSlopeIndex)>0){
```

```r
        #Vir ingeval daar meer as een .grd file per area is, maar daar
        behoort nie. for (j in 1:length(AspectSlopeIndex)){
        setwd(paste0("E:/Refined Data/",MineralrightsIndex[i],"/Aspect
        Slope")) InvestigatedArea <-
        stack(paste0(getwd(),"/",AspectSlopeIndex[j]))

          #Alle punte in die raster wat nie aan die "mask" kriterie
          voldoen nie, word weggegooi uit die datastel tans alles onder
          3 grade slope OF (tussen 3 en 5 grade slope en Noord aspect)
          InvestigatedArea <- mask(x = InvestigatedArea, mask =
          (InvestigatedArea$slope > 5) | ((InvestigatedArea$slope>3) &
          (InvestigatedArea$aspect>45) & (InvestigatedArea$aspect <315)),
          maskvalue = 0)
          #raster::area bereken die area per sel ("pixel") vir die hele
          raster
          InvestigatedArea <- area(InvestigatedArea$slope,na.rm = TRUE)

          #Haal al die NA waardes uit, tel al die area selle op en
          skaleer van km2 na m2
          InvestigatedArea <- InvestigatedArea[!is.na(InvestigatedArea)]
          InvestigatedArea <- sum(InvestigatedArea)
          InvestigatedArea <- InvestigatedArea*1000*1000

          #Skep nuwe dataframe inskrywing en stoor in die globale
          dataframe (rbind)
          newentry < data.frame(ID=as.character(MineralrightsIndex[i])
          ,AreaSize=InvestigatedArea)
          Resultdf <- rbind(Resultdf,newentry)
          GlobalIndex <- GlobalIndex+1

      }
    }
  }
}

#Sorteer en stoor globale resultaat dataframe as 'n CSV
setwd("E:/Refined Data")
Resultdf <- arrange(Resultdf,ID) write.csv2(Resultdf, file = "AreaSize
of slope exclusions per area.csv")
```

135

Addendum B

Table B.1: Result Table. Panel efficiency of 12% and 22% used for low and high case respectively and area factor of 0.7 included.

		Brut Insolation	Exclusions			After Panel Efficiency is included	
GID	Size of Area [m²]	kWh/day	Slope Exclusions [kWh/day]	Other Exclusions [kWh/day]	Nett kWh/day	Low [MWh/day]	High [MWh/day]
1710351	1566856	9040759	0	33969	9006790	757	1387
1710352	1099000	6418159	5211853	0	1206307	101	186
1710354	20667666	120492491	0	0	120492491	10121	18556
1710355	11139893	64945577	0	0	64945577	5455	10002
1710356	13162456	76737121	0	0	76737121	6446	11818
1710357	19054264,01	111086359,2	1351170,737	0	109735188,5	9218	16899
1710358	12126568,47	70697894,17	0	0	70697894,17	5939	10887
1710360	30551775,72	176283745,9	33474965,47	10608225,84	132200554,6	11105	20359
1710372	57266819,22	329234218,4	12996054,64	7006075,896	309232087,8	25975	47622
1710387	42576217,44	251199682,9	48171022,19	0	203028660,7	17054	31266
1710388	86612221,4	511012106,3	121336665,2	0	389675441,1	32733	60010
1710390	36228508,69	209038495,2	5386687,763	57958,74031	203593848,7	17102	31353
1710392	31106551,18	179484800,3	21174703,63	820751,4705	157489345,2	13229	24253
1710394	5417356,679	31258148,04	2571338,05	183275,3996	28503534,59	2394	4390
1710403	30349665,04	170565117,5	138153406,4	0	32411711,09	2723	4991
1710405	753527,6117	4475954,014	152718,4209	0	4323235,593	363	666
1710408	106044,1436	619297,7987	0	0	619297,7987	52	95
1710361	203518,416	1174301,26	0	0	1174301,26	99	181
1710371	220717037,2	1273537304	183916963,4	22872451,04	1066747890	89607	164279
1710411	13344694,91	77799571,32	0	0	77799571,32	6535	11981
1710412	7419156,406	43253681,85	0	47282,03903	43206399,81	3629	6654
1710413	6351175,083	37027350,73	0	217962,0936	36809388,64	3092	5669
95757	434037,9913	2439293,511	2168273,226	0	271020,2855	23	42
97373	450098,3881	2642077,538	357083,9951	492252,2226	1792741,321	151	276
2004518	48406197,04	276883447	54245705,49	8011237,729	214626503,8	18029	33052
2004519	42897230,69	245372159,6	15574675,83	67248409,32	162549074,4	13654	25033
2008340	586362,5529	3383311,93	354687,8751	0	3028624,055	254	466

2008341	7977153,981	46028178,47	5224228,96	924970,6473	39878978,86	3350	6141
1998245	31091239,57	179396452,3	0	20352226,19	159044226,1	13360	24493
1996615	80431584,18	460068661,5	50210494,42	179699213,1	230158954	19333	35444
1996630	88804,98619	499084,0224	0	48927,62622	450156,3961	38	69
1996631	50801,78137	285506,0113	0	112314,5928	173191,4185	15	27
1996632	6948,385346	39049,92564	0	4432,670148	34617,2555	3	5
2013806	11288547,71	66941087,9	0	23662248,84	43278839,06	3635	6665
2013807	1890696,5	10625714,33	5136575,101	3060922,722	2428216,509	204	374
1993763	295913384,8	1686156515	35332676,7	41466026,7	1609357812	135186	247841
1993798	197374370,8	1124263377	109480758,6	9448082,451	1005334536	84448	154822
1993715	10075720,28	56625547,98	10006724,05	0	46618823,94	3916	7179
1994010	17253564,96	100413399,6	15723215,99	18463090,02	66227093,55	5563	10199
1993255	2505091,322	14078613,23	2474265,728	8110475,883	3493871,619	293	538
1993256	6595163,74	37064820,22	6427766,002	9905750,233	20731303,98	1741	3193
1991062	348898,0103	2086410,101	0	232335,889	1854074,212	156	286
1991064	25669988,21	151902748,7	51967466,13	202713,3964	99732569,22	8378	15359
2020860	15675025,94	92012402,26	0	59962847,64	32049554,62	2692	4936
2021034	48221217,66	275825365	54772560,8	5091431,283	215961372,9	18141	33258
2022452	35684210,94	209466318,2	163908236	11705217,98	33852864,24	2844	5213
2026792	98897,80109	570640,3123	0	0	570640,3123	48	88
2026794	1317744,169	7603383,854	0	0	7603383,854	639	1171
2027064	2212937,793	12768651,07	0	802620,7487	11966030,32	1005	1843
2027404	608523,0461	3511177,976	0	2144681,911	1366496,065	115	210
2027403	256970,6385	1482720,584	0	1247974,934	234745,6505	20	36
2027405	420871,0124	2428425,741	0	1188549,611	1239876,131	104	191
2027698	146437,1359	844942,2743	0	177269,9316	667672,3427	56	103
2027881	312022,0701	1800367,344	0	1717474,639	82892,70503	7	13
2030237	44905671,02	263596288,9	98215337,97	15045367,24	150335583,7	12628	23152
2030470	17478,4274	100850,5261	0	10936,64372	89913,88236	8	14
2030463	2658052,769	15336964,48	0	0	15336964,48	1288	2362
2030471	164311,8056	948079,1182	0	20197,9646	927881,1536	78	143
2030473	36660,88572	211533,3106	0	0	211533,3106	18	33
2030476	81147,30054	468219,9241	0	7476,429327	460743,4948	39	71
2030525	4531735,419	26148113,37	0	284086,5174	25864026,85	2173	3983

2032564	13003,88523	75032,41777	0	0	75032,41777	6	12
2032565	29134,90666	168108,4114	0	0	168108,4114	14	26
2032567	476313,7151	2748330,136	0	289442,0898	2458888,047	207	379
2032596	811340,324	4681433,669	0	3639028,728	1042404,941	88	161
2032603	2801432,013	16164262,72	1392257,85	0	14772004,87	1241	2275
2035415	1174358,841	6858255,633	2483179,734	0	4375075,899	368	674
2036770	24272800,73	142481340,3	16725016,95	4297100,398	121459222,9	10203	18705
2041832	4500043,081	26460253,31	1140500,772	9496561,481	15823191,06	1329	2437
2043528	477996,4499	2734139,693	1745309,857	144663,8268	844166,0094	71	130
2051178	629913,2049	3697590,513	1988102,079	704883,3305	1004605,103	84	155
2051572	22183205,66	124669615,8	57541572,95	13195887,12	53932155,74	4530	8306
2093396	986005,7513	5748413,53	0	281756,1672	5466657,363	459	842
2097662	7058313,358	41291133,14	9179273,517	565571,8771	31546287,75	2650	4858
2097666	8137297,916	47603192,81	20496334,85	1240371,127	25866486,84	2173	3983
2097670	19152256,05	112040697,9	30824887,06	1812321,684	79403489,15	6670	12228
2097674	120821,6421	706806,6064	0	25540,61435	681265,992	57	105
2112730	1596468,695	9307412,491	0	3459495,442	5847917,049	491	901
2113231	25912880,92	151072095,8	3030035,513	3713567,195	144328493,1	12124	22227
2113212	56465773,6	330482689,4	5082182,023	30639989,79	294760517,6	24760	45393
2113230	902480,3296	5261460,321	396004,835	77139,16184	4788316,325	402	737
2113954	85400430,22	500001810,8	6482856,908	80587162,87	412931791	34686	63591
2134300	7678417,706	44765175,22	0	13900749,12	30864426,11	2593	4753
2134299	11811016,42	68858225,74	852810,0332	1908249,713	66097165,99	5552	10179
2122773	2080150,742	12210484,86	7612137,52	101680,7396	4496666,597	378	692
2127546	8423,558455	49446,28813	0	0	49446,28813	4	8
2128402	5971925,379	34816324,96	0	7986015,542	26830309,42	2254	4132
2137726	169590069,6	989199562,7	115386336,1	72778591,32	801034635,3	67287	123359
2137787	142678851,7	831817705,3	14009090	4913991,015	812894624,3	68283	125186
2140434	4731850,367	27302776,62	0	35400,41648	27267376,2	2290	4199
2140466	4374250,986	25501883,25	419673,6285	1012760,727	24069448,89	2022	3707
2141002	11087873,39	64642301,84	0	35394279,64	29248022,2	2457	4504
2146870	66600838,65	384286839	21757681,38	7996998,683	354532159	29781	54598
2150908	24854,6542	145896,8202	0	61953,50383	83943,31634	7	13
2150913	9999,770813	58698,65467	36326,78444	0	22371,87023	2	3

2151126	129342,9926	759243,3664	0	139940,7234	619302,643	52	95
2151128	2063,436047	12029,83215	0	4222,314773	7807,51738	1	1
2151129	800014,3684	4675961,361	0	111657,8311	4564303,53	383	703
2151215	77192,19087	453118,1604	0	99330,12594	353788,0345	30	54
2151218	1564548,892	9183901,995	0	1345886,881	7838015,114	658	1207
2151219	149995,4187	880473,1077	0	51309,60733	829163,5004	70	128
2151220	1570422,185	9218378,225	0	3447792,867	5770585,359	485	889
2151221	157419,8966	924054,7928	0	568089,7495	355965,0433	30	55
2151222	39973,40541	234643,8897	0	231563,5642	3080,325575	0	0
2151224	1264059,786	7420030,943	0	2573461,179	4846569,764	407	746
2151225	360053,6291	2113514,803	0	1592873,026	520641,7775	44	80
2151226	6727,971044	39493,19003	0	37724,16564	1769,024387	0	0
2151240	2284277,111	13408706,64	0	5869084,476	7539622,166	633	1161
2151242	268872706,9	1572870901	23590371,75	366404124,6	1182876404	99362	182163
2151674	1434847,21	8279068,404	0	2600163,797	5678904,607	477	875
2151675	6577621,942	37952878,61	0	5612645,368	32340233,24	2717	4980
2151676	8860523,229	51125219,03	0	541765,9668	50583453,06	4249	7790
2151686	1146023,51	6612555,651	0	2437742,422	4174813,229	351	643