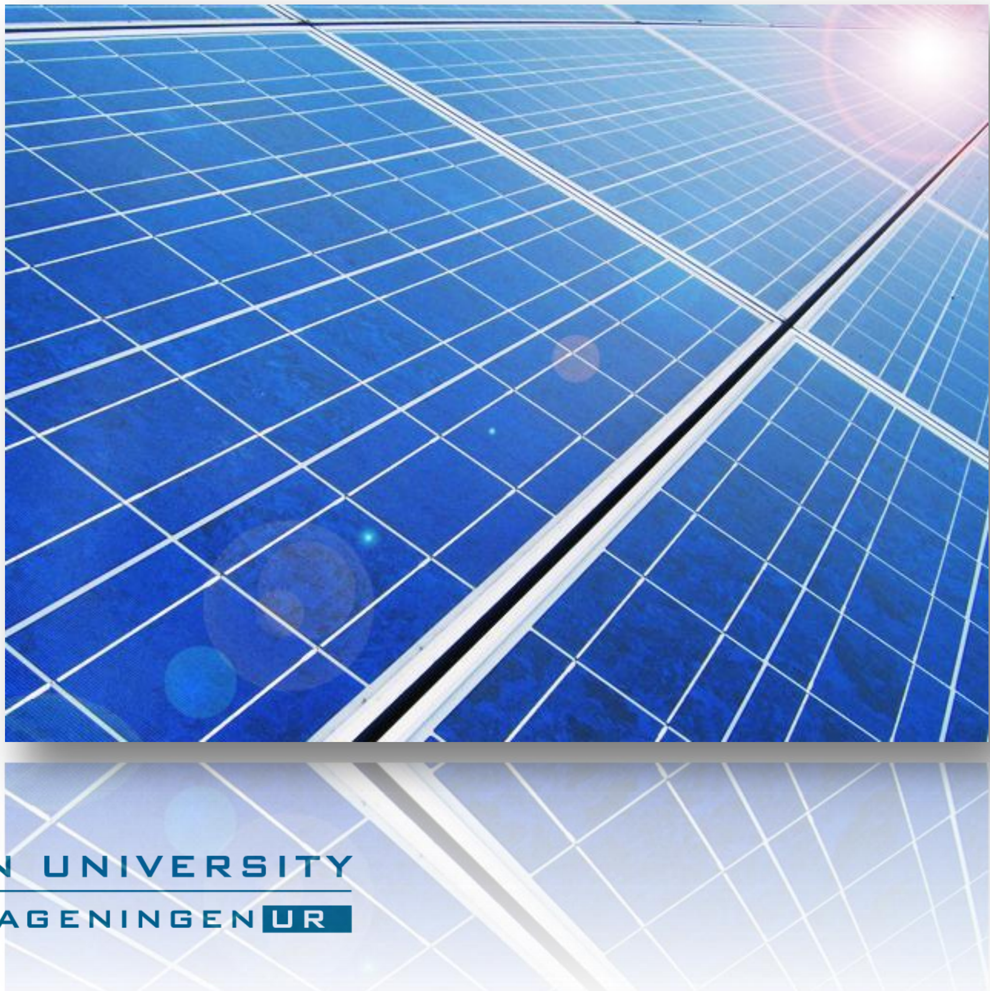
Centre for Geo-Information

Thesis Report GIRS-2011-24

Estimating the return on investment of solar panels in urban environments using 3D geodata

Paul Ruiter

October 2011



WAGENINGEN UNIVERSITY
WAGENINGEN UR

Estimating the return on investment of solar panels in urban environments using 3D geodata

Paul Ruiter

Registration number 86 11 17 716 030

Supervisors:

Ron van Lammeren

Arnold Bregt

A thesis submitted in partial fulfilment of the degree of Master of Science
at Wageningen University and Research Centre,
The Netherlands.

4 October 2011
Wageningen, The Netherlands

Thesis code number: GRS-80436
Thesis Report: GIRS-2011-24
Wageningen University and Research Centre

Laboratory of Geo-Information Science and Remote Sensing

FOREWORD

This report is the results of half a year of research for my MSc thesis in Geo-Information Science. The research has a strong emphasis on 3D GIS and more importantly how to apply these possibilities.
For me this research has not only been an exploration of the possibilities of 3D GIS, but it also forced me to dive in the world of climatology, mathematics and scripting. During this period I found out about the limitations of the available ArcGIS software, and luckily also found a solution in the Python scripting language. The combination of different data, formula's, scripts and more results in the document that lies before you today.

Thanks go out to the people that helped me in some way in this process. Special thanks goes out to the municipality of Groningen for supplying data and also Ron van Lammeren, for the periodical feedback and giving me the space to find my own way in this research.

Paul Ruiter, October 2011

ABSTRACT

The main research objective of this thesis research is to determine and apply a method for solar energy potential estimation based on 3D virtual city data. The methodology used in this research benefits from the height spatial accuracy that can be achieved by using polygons instead of a raster. Two benefits of using raster (faster processing and simple geometry) are achieved by using points instead of polygons for simulating insolation values on roof surfaces. A polygon based method would suffer too much performance loss from iterative splitting a polygon in multiple polygons. The downside of this approach is that surface areas cannot be retrieved based on the points.

The research area consists of a small area (approximately 300 by 250 meter) near the inner city of the Municipality of Groningen; this area is selected based on availability of the data.
The topic of this research is the potential of small scale urban solar energy generation; this topic is captured in the two main research questions:

1. What are potential locations for local solar energy production in an urban environment?
2. What is the predicted impact of local solar energy production on the existing infrastructure?

The first research question is answered by using a 3D city model in combination with meteological data to determine the insolation on the roof surfaces. Both direct, diffuse irradiance and the possible shadows cast from neighboring buildings are taken into account. Since the temporal scale is also of importance, for 14 days (evenly spread over a year) every hour the position of the sun and irradiance are simulated. Locations with the highest potential for solar energy production are characterized by the highest Return on Investment (RoI). To determine this, the yearly insolation values are converted to RoI times of solar panels.
The potential for solar panels on roofs is determined by calculating the RoI values, which are derived from the insolation, cost of electricity, cost of solar panels and yearly price mutations. The total area of potential suitable locations is 8412m2. 1825m2 (22%) roof surface will not achieve a positive RoI before the lifetime of a solar cell expires, assuming that the lifespan of a solar panel is 25 years. Even when the solar panel is positioned with an optimal orientation is most of the time in direct sunlight, the RoI time is still 19 years.

The second research question compares the potential solar energy production and energy demand in the study area. Due to lack of information with a higher temporal precision only an indication is given on the potential impact this can have on the existing energy infrastructure. The effect of the solar energy on the distribution grid depends on the coincidence between energy production and demand. Also the flexibility in energy output of the distribution grid is of importance.
Assuming 10% of the area with an RoI time =< 20 years (regardless of to which property a roof belongs) will be used for solar cells and all solar energy is used 52% of the yearly energy demand from private properties can be met, or 2% of the energy demand from commercial properties. Literature shows that in the USA there already is a significant waste of energy when 10%-20% of the yearly energy is provided by solar technology, which reduces the financial benefit of this technology. Assuming that the USA is comparable with the Netherlands this would mean that when more solar panels are installed at some point the flexibility in energy production in the distribution grid has to be improved.

CONTENTS

1 Introduction ... 6

1.1 Context and Background ... 6

1.1.1 Photovoltaic Solar Cells ... 6

1.1.2 Distributed Energy Resources and Smart Grids ... 7

1.1.3 3D Virtual Cities ... 8

1.2 Problem Definition ... 9

1.3 Report Outline ... 10

2 Methodology ... 11

2.1 Research Questions ... 11

2.2 Study Area ... 11

2.3 Conceptual Framework ... 13

2.3.1 Data Description and Data Preprocessing **(1)(2)** ... 13

2.3.2 Processing **(3)** ... 15

2.3.3 Processing (4) ... 16

3 Potential Energy Production ... 20

3.1 Optimal positioning for solar panels ... 20

3.1.1 Yearly pattern ... 20

3.1.2 Montly Variations ... 22

3.2 Usable Roof Surface ... 24

3.2.1 Point Representation of Roof Surface ... 24

3.2.2 Roof Surface after Simulation ... 24

3.3 Potential Energy Production ... 25

3.4 Financing Solar Energy ... 26

3.4.1 Cost of Modules ... 26

3.4.2 Cost of Electricity ... 26

3.5 Return on Investment Time ... 27

3.6 Conclusion chapter 3 ... 29

4 Balance between Demand and Production ... 30

4.1 Energy Demand ... 30

4.2 Demand versus Potential Production ... 31

4.3 Conclusion chapter 4 ... 32

5 Conclusion ... 33

6 Discussion ... 33

6.1 Input Data ... 33

6.2 Modelling ... 33

6.3 Solar Cell potential and Return on Investment ... 35

6.4 Energy Demand and Production ... 35

7 Works Cited ... 38

Appendix 1: Mathematics ... 40

Appendix 2: Python Scripts ... 42

1 INTRODUCTION

This chapter introduces the topics relevant for this research. The first paragraph puts the research in context and gives background information on some relevant aspects, such as photovoltaic solar cells, smart grids and 3D virtual cities. The second paragraph contains the problem definition and the final paragraph gives the report outline.

1.1 CONTEXT AND BACKGROUND

Climate change is and will be an important topic for decennia to come. Governmental organizations are actively generating knowledge and pursuing goals to reduce the emission of greenhouse gasses. This happens on every governmental level, from worldwide of municipal. However, even the best intentions cannot be fulfilled when the public does not agree with a new policy. A recent report commissioned by the European Parliament (TNS Opinion & Social, 2009) shows that the citizens of different countries have different opinions and views towards climate change. This report shows for example that the Dutch citizens are well informed on the causes of climate change and the actions that can be taken to reduce climate change. More than half of the Dutch respondents already claimed that they have taken action to help fight climate change, but the Netherlands are still only on the 12th place (of total of 32 countries) with the percentage of people that have personally helped to fight climate change. What people have done to fight climate change is also not clear. It could for example be the purchase of a new cleaner car, installing solar panels or replacing regular light bulbs with energy saving light bulbs.
The public can undertake different actions to reduce their carbon footprint. This ranges from consuming less or different products to investing in renewable energy. The latter is currently stimulated in different countries with tax incentives and subsidies. This can be valid option from a policy point of view, but the technical implications of this are often ignored.

This research focusses on the possibility of solar panels in an urban environment based on 3D models and to a lesser extent on the impact on the existing electricity infrastructure. The following paragraphs introduce different relevant topics.

1.1.1 PHOTOVOLTAIC SOLAR CELLS

"Photovoltaic conversion is the direct conversion of sunlight into electricity without any heat engine to interfere. Photovoltaic devices are rugged and simple in design requiring very little maintenance and their biggest advantage being their construction as stand-alone systems to give outputs from microwatts to megawatts." (Parida, Iniyan, & Goic, 2011). This quote sums the main advantages of the photovoltaic (PV) solar cells; they are low on maintenance requirements and have little or no moving parts that can cause any noise disturbance.
A solar panel consists of different cells connected to each other. Multiple solar panels or modules together form an array.
Due to its flexibility in size, layout and positioning it has a high potential to be used in urban environments as energy source, especially on roof surfaces. To be used the position of the sun relative to the solar cell has is important. It uses direct and diffuse sunlight to generate power, and the more sunlight it receives the more electricity is generated. This also means that the amount of electricity produced can fluctuate highly over time.
An extensive overview of PV technology can be found in (Parida, Iniyan, & Goic, 2011).

It is important to note that there are different terms often used with regards to solar energy, namely irradiance and insolation. Irradiance is the amount of energy from the sun per area over a given time; insolation is the amount of energy received by a surface area over a given time.

Source: http://www.solarnavigator.net/solar_panels.htm

Figure 1: Array of solar panels

1.1.2 DISTRIBUTED ENERGY RESOURCES AND SMART GRIDS

There is consensus in the scientific community that localized production of energy a step forward. Localized energy production is often referred in scientific literature as Distributed Energy Resources (DER). This refers to all electric power generation resources that are directly connected to medium or low voltage distribution systems. There is a wide variety of DER (Akorede, Hizam, & Pouresmaeil, 2009) of which photovoltaic systems or solar cells is one.

One of the main problems with DER is that the current distribution grid isn't set up to deal with energy production. Figure 2 represents the current average electricity infrastructure where energy mostly flows in one direction, a few large energy suppliers supplying the consumers. This situation however changes more and more to a situation there is a significant energy production at the level of the city network. The effect of this change is that the energy infrastructure has to become more flexible, handle energy flows in multiple directions.

There are several concepts on how to improve the distribution grid are made, such as smart grids (Hammons, 2008). A more extensive overview of technologies can be found in (Chicco & Mancarella, 2009).

The current electricity infrastructure can be seen as an ancient remnant of the past. It is still focussed on supplying energy from a few big sources to a large range of customers; while the focus nowadays shifts more too locally produced and distributed energy. This means that there have to be changes to the existing infrastructure to allow for such distributed energy resources. These changes are captured in a concept called smart grids.

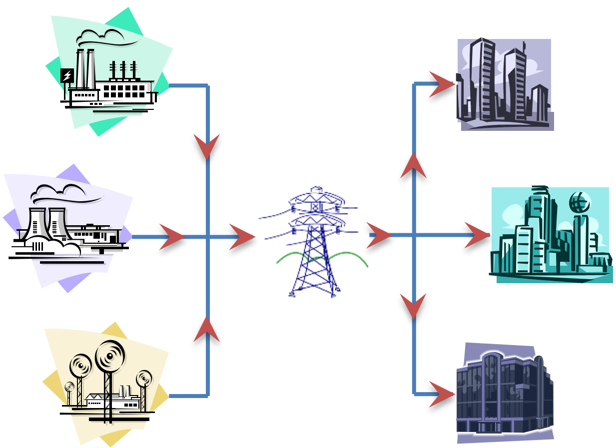


Figure 2: Current network with energy producers (left) and energy consumers (left)

The next generation of electricity infrastructure is often referred to as smart grids. "*The term 'Smart Grid' refers to a modernization of the electricity delivery system so it monitors, protects and automatically optimizes the operation of its interconnected elements – from the central and distributed generator through the high-voltage transmission network and the distribution system, to industrial users and building automation systems, to energy storage installations and to end-use consumers and their thermostats, electric vehicles, appliances and other household devices.*

The Smart Grid will be characterized by a two-way flow of electricity and information to create an automated, widely distributed energy delivery network. It incorporates into the grid the benefits of distributed computing and communications to deliver real-time information and enable the near-instantaneous balance of supply and demand at the device level". (Electric Power Research Institute, 2009).

Shortly summarized a smart grid is an electricity network that is 'aware' of all relevant parties within the network, both consumers and producers of electricity. This awareness of the smart grid is essential to make the concept of DER work, since you need bi-directional energy flows and you have to track where energy is produced and required to make sure a minimum amount of energy is wasted. This means that the scheme shown in Figure 2 will change to a scheme where every energy producer and consumer is interconnected with new infrastructures.

Smart grids also generate new possibilities such as direct energy pricing and smart use of appliances. For example, when there is a overproduction of energy the energy price could drop. If a washing machine is then connected to the smart grid it can detect the drop in the energy price and start its washing cycle. This way less energy is wasted and the user has a financial benefit.

1.1.3 3D VIRTUAL CITIES

There is always a discrepancy between the real world around us and the virtual reality that is used for modelling certain phenomena. The urban environment is a unique environment, mainly due to the complex structures and high variability in height dimensions. Without a third dimension, there is a great deal of information lost such as the overhang of a roof (see Figure 3)

Source: http://www.americansteelbuildings.com/accessories.htm

Figure 3: Overhanging structure

For modelling 3D urban environments there is an open industry standard issued by the Open Geospatial Consortium (OGC) called CityGML. CityGML is best explained by the following quote: *"CityGML is a common information model for the representation of 3D urban objects. It defines the classes and relations for the most relevant topographic objects in cities and regional models with respect to their geometrical, topological, semantical and appearance properties. Included are generalization hierarchies between thematic classes, aggregations, relations between objects, and spatial properties. This thematic information go beyond graphic exchange formats and allow to employ virtual 3D city models for sophisticated analysis tasks in different application domains like simulations, urban data mining, facility management, and thematic inquiries"* (Kolbe, 2005).

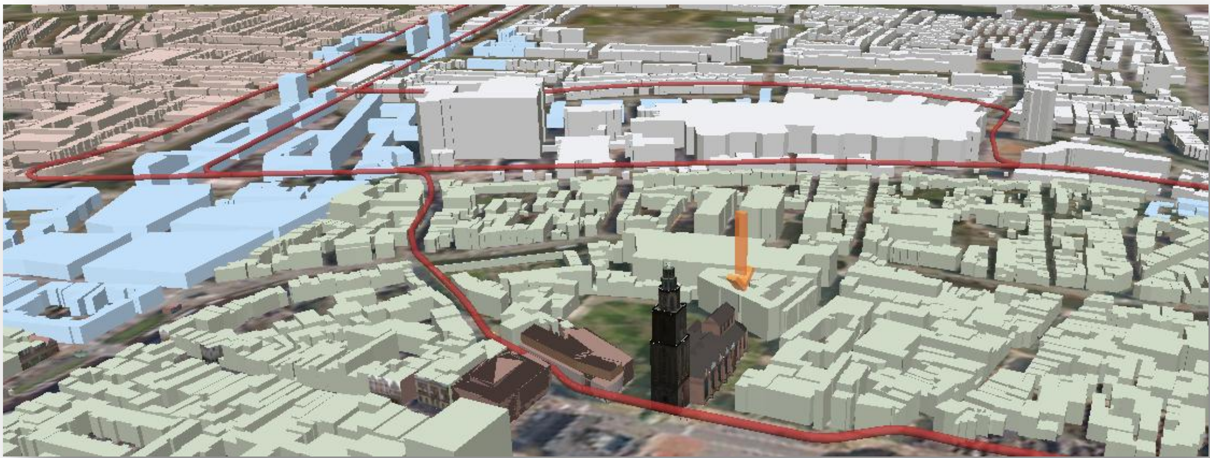


Figure 4: VirtuoCity in Groningen

VirtuoCity is a dutch experimental platform of the CityGML information model. Different municipalities use this as a medium to experiment to supply information to and communicate with the general public using a 3D environment. In most cases this involves new spatial plans. The municipality of Groningen for example uses VirtuoCity to communicate about a new tram line (see Figure 4). Users can 'fly' over the city or zoom in and walk through the streets. At one location is it also possible to see some additional information for that location and leave a comment, which can be seen as the first step to virtual public participation in urban planning.
Information stored using the CityGML information model has much more potential then visualization and communication purposes, it can also be used for analysis different phenomena, for example wind (Reiter, 2010), turbulence (Barbason & Reiter, 2010) or energy demand (Strzalk, Eicke, Coors, & Schumacher, 2010).

1.2 PROBLEM DEFINITION

The public is willing to buy energy saving lights, however investments in energy production often aren't made due to lack of knowledge. People often don't have an idea on what the potential of energy production using solar cells is. This lack of knowledge often is both technical and financial, the possible energy production and the possible financial benefits are often unknown.
To increase public participation and investments in sustainable energy production the public needs proper information. One of the goals of municipalities is to increase the use of renewable energies; however municipalities often lack the resources and knowledge to provide proper information. Some other parties do have the resources, knowledge and/or need to provide this information, which can help serving the public interest.

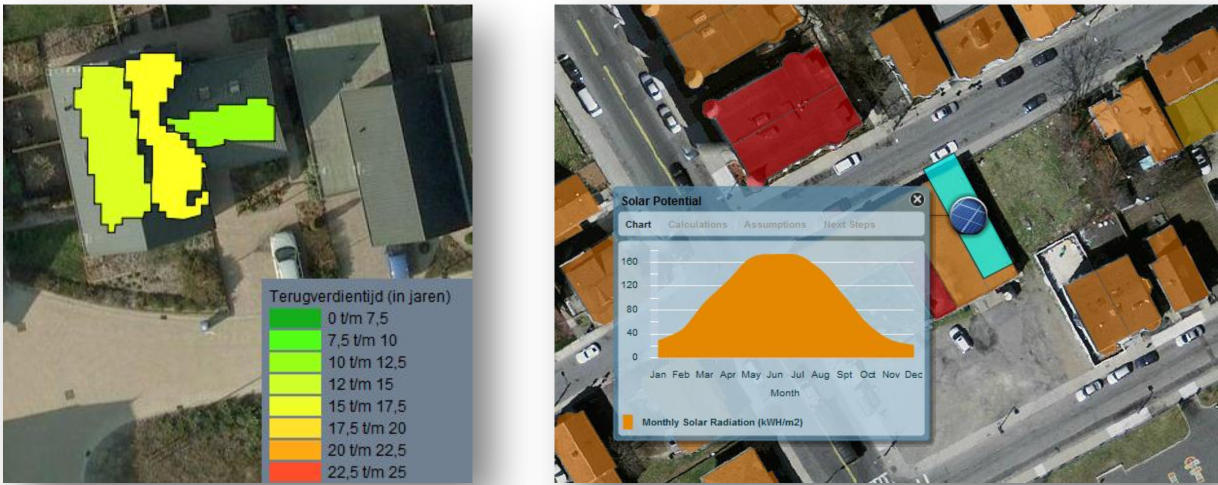


Figure 5: Example of solar potential visualization left Assen, right Boston

Anticipating that the current trend of decreasing prices for solar panel (Bradsher, 2010) (Solarbuzz, 2011) continues, it would be wise to improve knowledge on which locations are optimal. Creating an overview of potential good locations for solar energy that is available to the public would be a good step to improve knowledge for the public. Using this information the public can make a more easily an informed decision. There are already different examples available (Solar Boston) (Van der Vaart, 2009) (Municipality Assen) where the public can find out what the potential is for solar energy is for their rooftop.
The data shown in Figure 5 is accurate on the scale of a city. The methods are based on 2D rasters; in the case of Boston this raster is reverted back to polygons. The inherent problem of raster formats is that they cannot capture complex roof geometry in high detail. This can be seen in the Boston example, where a building consists of one of a couple of polygons. It can also be seen in the example of Assen, where the raster is slightly misaligned and also ignores small roof elements. Since the public often does not know much about solar energy and are not GIS professionals so these representations might not convince them easily.

The research objective starts with determining a methodology to improve solar panel return on investment estimation by using 3D geodata, after which this method is applied for accurate analysis of potential solar panel locations. Additionally the possible impact of localized solar energy production is determined.

This research investigates a new way on analyzing the potential for solar panels in an urban environment based on 3D models. Two main research questions will be answered in this report, namely:

1. What is the potential of local solar energy production in an urban environment?
2. What is the predicted impact of local solar energy production on the existing infrastructure?

Both of these research questions are further decomposed in chapter 2.1.

1.3 REPORT OUTLINE

The following chapters give answer to the research questions. In the next chapter firstly the research questions are further decomposed and the study area introduced. After this the methodology is explained (§2.3). This methodology mainly applies to the first research question, since this question requires the most extensive processing.

Chapter 3 builds step by step towards the answer of the first research question, starting with the theoretical optimal position calculated for solar panels (§ 3.1) and the usable roof surface(§ 3.2). After this the potential energy production on the roof surfaces is determined (§ 3.3). Finally the financial aspect is described (§3.4) and the insolation values on the roofs are translated to a return on investment time (§ 3.5).

Chapter 4 deals with the second research question. The energy demand in the study area is determined and compared with the potential energy production (§ 4.1). After this an indication of the effect of solar panels in this area is made (§ 4.2).

2 METHODOLOGY

This chapter explains the methods the steps that are taken and the methods used. This chapter also gives background information where required on choices made.
The methodology starts with stating the research questions, followed by the conceptual framework. This framework explains different choices made and individual modelling parts are further elaborated.

2.1 RESEARCH QUESTIONS

The goal of this research is to answer two main research questions. Each of these research questions are composed of different detailed research questions (DRQ). These research questions are:

1. What is the potential of local solar energy production in an urban environment?
   1.1. What are the roof characteristics (such as geometry and orientation)?
   1.2. What are the hourly metrological conditions on the location?
   1.3. What is the effect of cloud cover and shadows on the irradiance values?
   1.4. Which roofs have a potential for solar cells?
   1.5. Which roofs are affected by shadow cast by other buildings?
   1.6. What are the simulated insolation values on the roofs?
   1.7. What is the potential energy production per m2 for each surface?
   1.8. What could be the possible financial Return on Investment of placed solar panels?

2. What is the predicted impact of local solar energy production on the existing infrastructure?
   2.1. What is the energy demand and potential energy production in the area?
   2.2. Is there a temporal mismatch between energy production and demand?

2.2 STUDY AREA

The study area for this research is situated near the center of the city Groningen in the northern part of the Netherlands (see Figure 6). This is an area that consists mostly of older buildings with some exceptions. The study area consists of a mix of building types, ranging from a large 15 century church to residential buildings to a large modern office building.
This specific area is selected because of this area a 3D model is available that is made for the VirtuoCity project. The model has a high accuracy, which means that complex roof structures are modeled with centimeter accuracy (see Figure 9).

Figure 6: Location of study area Source: http://maps.google.nl Source: http://nl.wikipedia.org/wiki/Nederland

Figure 7: Position of the study area and its surrounding area (2D)

Source (background): http://maps.google.nl

Since there are no 3D models available of the surrounding area at the time of this study a small selection of the study area is used for the actual calculations (see Figure 7 and Figure 8). The other buildings serve as surrounding objects.

The total area has a dimension of approximately 300 by 250 meters. The total area footprint of the buildings in the whole area is 31125m^2. The footprint of the actual study area is 6935m^2. This means that ~22% the whole 3D model is used for determining the suitable roofs and irradiance values. The resulting ~78% is used as surrounding buildings that have the potential to influence the shadow conditions.

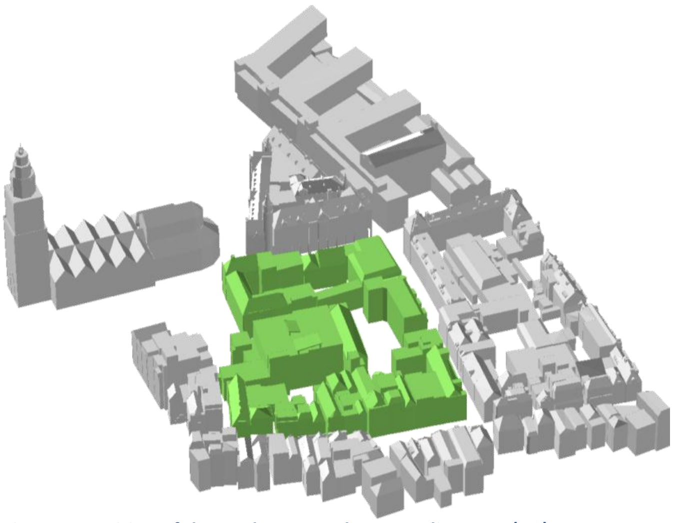


Figure 8: position of the study area and surrounding area (3D)

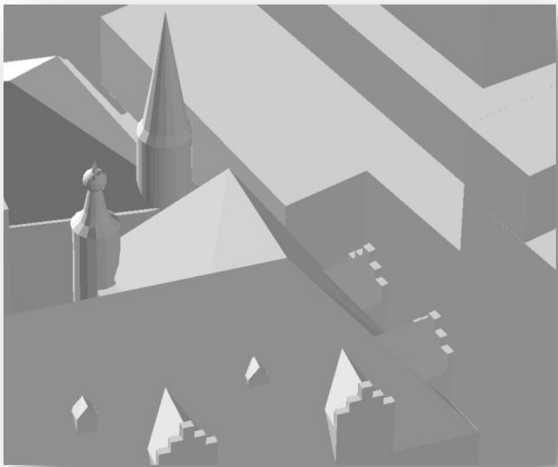


Figure 9: Detail 3D model

2.3 CONCEPTUAL FRAMEWORK

The work done in this research project can be divided in two different parts, namely data generation and data analysis. Data generation deals with the steps of preprocessing, processing and creating the desired output. Data analysis analyzes the output and does not require extensive modeling or complex methodologies and thus this will not be discussed in this chapter but where required in chapter 3 or 4.

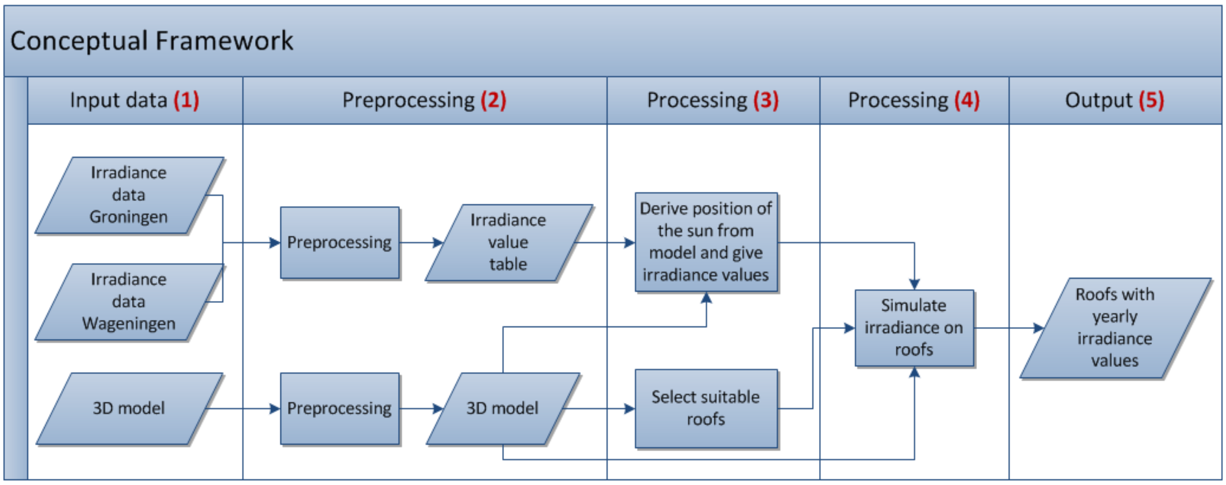


Figure 10: Conceptual framework

In Figure 10 the conceptual framework is presented, where the input data and preprocessing are case specific. The different phases are numbered to provide better overview. After the data is preprocessed there are two processing parts. This division is made because Processing **(3)** prepares data for Processing **(4)** and the second step is also more elaborate then the first.
In the next paragraphs the data **(1)**, preprocessing **(2)** and processing **(3)** is described.

2.3.1 DATA DESCRIPTION AND DATA PREPROCESSING (1)(2)

For this research three different data sources are used. Firstly, climatological data from the weather station near Groningen (KNMI) and a weather station in Wageningen (WUR) are used. Secondly, a 3D model from the study area in Groningen (Municipality Groningen) is used. All other data is used in this research is derived from these sources.

3D CITY MODEL

The 3D model supplied by the municipality of Groningen required some more preprocessing, mainly due to incompatibility of ArcGIS 10 with the file format the data was supplied in. The original model was made in Bentley Microstation, a CAD based system. ArcGIS 10 offers some support for CAD files, but the importing the file in question resulted in some random shapes. Using the conversion software FME from Safe Software to convert from .dgn to .shp files was more successful, however also using this software the complex roof characteristics were not translated properly.
experimenting the final solution for this problem was exporting the data in Microstation to a Sketchup file, saving this file to an older version in Sketchup (since ArcGIS can’t handle the new version files) and importing this to ArcGIS. Importing the Sketchup file results in a single Multipatch object, which is for this project is not suitable.
There is no tool available within ArcGIS to get the separate polygons that make up the multipatch. To get the separate polygons a small program is used called Arcv2CAD 6.0 (Guthrie CAD/GIS Software, 2011). This creates a CAD file from the shapefile, this CAD file can then be read back in arcGIS and stored as a shapefile.

The only downside of these conversion steps is that some polygons are corrupted in a way that, according to ArcGIS, they have an invalid geometry. These polygons cannot be taken into account during the analysis.

CLIMATOLOGICAL DATA

To create the climatological model two different datasets are used. One dataset is from the Royal Netherlands Meteorological Institute (KNMI) and the other is from Wageningen University and Research Center (WUR), Meteorology and Air Quality section.

Table 1: Yearly irradiance averages (weather station Eelde)

Year	kJ/cm2
2001	355
2002	345
2003	392
2004	365
2005	364
2006	364
2007	355
2008	365
2009	379
2010	373

The dataset from the KNMI has the following properties:

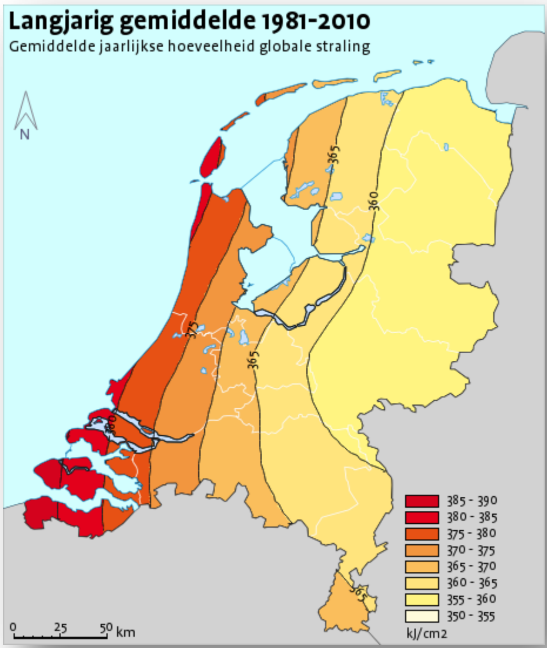
- Per hour total irradiance values (J/cm^2)[1]
- Measured at airport Groningen Eelde (~10km from the study area)

The dataset from WUR has the following properties:

- Per hour direct and diffuse irradiance values (Wh/m^2)[2]
- Measured in Wageningen (approx. 160 kilometres from the study area)

Two datasets are used since none of the datasets offer all the information required. The total irradiance from KNMI gives accurate information for the location; however it misses information on diffuse irradiation. To generate this information data from WUR is used. Since the relation between the two different units (J/cm^2 and Wh/m^2) is linear, the ratio direct and diffuse radiation is calculated and this ratio can be directly applied to the data from KNMI. On short term this may not be the most accurate way, but on the longer term the inaccuracies even out (Kroon, 2011). Also, on the short term this method maintains the fluctuations in diffuse radiation that occur.

The climatological data is gathered over many years. Since global irradiance is an important measure for the possible energy production of solar panels this is the main criteria to select a year. According to Figure 11 the long-term average of global irradiation for the municipality of Groningen is 355 360 kJ/cm^2.
When looking at the yearly global irradiation of the weather station near Groningen (see Table 1) it is clear that only 2001 and 2007 fall in this range. For this study the data from the year 2007 is used. This date is more recent which increases the probability that other data is also available for the same year.



Source: http://www.klimaatatlas.nl/klimaatatlas.php

Figure 11: Long-term average global radiation

[1] 1 Joule is an energy unit per second
[2] 1 Watt-hour equals 3600 Joules

2.3.2 PROCESSING **(3)**

After the data preprocessing the first steps are selecting suitable roofs, deriving the position of the sun and adding irradiance values. This processing step also answers the first two detailed research questions.

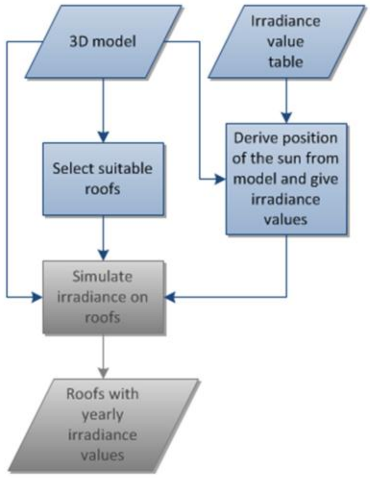


Figure 12: Concept processing (1)

SELECTING SUITABLE ROOFS

To select suitable roofs several criteria are set. These criteria are:

- area should be at least $2m^2$
- Angle of the roof should be between 0° and 75° (see also § 3.1)
- Height should be at least 6 meters.

Within ArcGIS 10 there is no tool available to determine the angle or direction of polygons. To calculate the angles a script (see appendix 2) is used that calculates the normal vector of each polygon. Based on the normalized normal vector (unit vector) the angle, rotation and minimum height of the polygons is determined and written back to the attribute table. Once this new data is added a selection is made based on the predetermined criteria. It should be noted that this selection also selects areas that are (partially) inside the building. This however should not influence the final outcome of the study, since these polygons will always lie in the shadow of others and can be removed from the final output.

DERIVING THE POSITION OF THE SUN AND ADDING IRRADIANCE VALUES

To derive the position of the sun a script from the ArcGIS resource center is used. This script determines the center of the input features and using this center point is calculates the position of sun for a given timeframe. This script uses a module called PyEphem (PyEmphem Home page, 2009). Since the study area is small the position of the area relative to the sun is accepted as equal for the whole study, so the position of the sun is represented by 1 position per time interval.

The simulated position of the sun is used as input in further steps. For one day every four weeks (so 13 days in total) the hourly position of the sun is simulated. This results in 156 sun positions. The temporal interval is chosen to balance processing time and results.

The irradiance values that are gained in the preprocessing consist of a table with the values for every hour of every day in the year 2007. To create irradiance values for the different positions of the sun the sum of irradiance values two weeks before and after the date is taken. This way the total sum of irradiance represents the yearly irradiance best and also preserves the different irradiance values in those four weeks.

Aggregating the data is done using MS Excel, after which the data is joined to the sunpoint shapefile based on the combination of date and time.

2.3.3 PROCESSING (4)

The second part of the processing is the simulation of the insolation values on roofs based on the different positions of the sun. This part of the processing deals with the insolation simulation and creating the output data as can be seen in Figure 13.
Simulating the insolation values is actually a process of several steps. First it has to be determined how the insolation is calculated and secondly the geometry used for the simulation. This processing step answers detailed research questions 3-6.

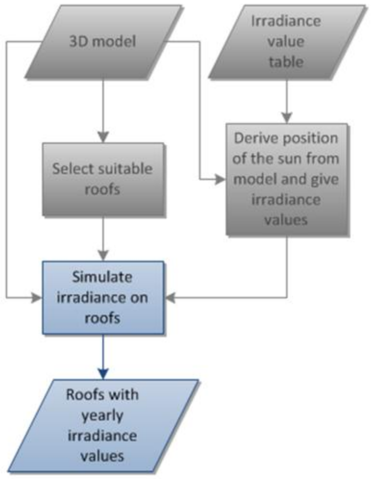


Figure 13: Concept processing (2)

INSOLATION ASPECTS

The insolation value on a roof depends on several factors:

- The angle of incidence between the sun beam and roof
- Direct/diffuse irradiance
- Shadows from neighboring objects

Each of these factors are taken into account for this research.
The angle of incidence is important to take into account because it is assumed that radiation travels in beams. If a beam is not perpendicular then irradiance is divided over a wider area (see Figure 14), reducing the insolation per area unit. The angle of incidence is calculated as a ratio (see appendix 1) with a value between -1 and 1, were:

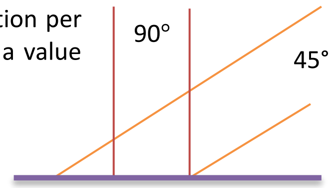


Figure 14: insolation area at different angles

- When the sun beam is perpendicular to the surface the value is 1
- When the sun beam is parallel the value is 0
- If the surface is facing away from the sun the value is < 0

To calculate the insolation of direct irradiance on a roof the ratio is multiplied with the direct irradiance value for that specific date.

Calculating the insolation of diffuse irradiation is not so straight forward, since there is not a single method that can calculate the exact values (except for measurements on location).
There are a wide variety of models available in literature that attempt to calculate the diffuse irradiation, both isotropic (diffuse radiation is uniform over the sky dome) and anisotropic (diffuse radiation is not uniform over the sky dome). These models range from limited input requirements to more elaborate models that require more input parameters. For this research only the position of the roofs, position of the sun and diffuse irradiance value is known. This limits the availability of methods. It is also important to note that isotropic models are simpler but the assumption of isotropic diffuse radiation is incorrect (Hamilton & Jackson, 1985).
Literature shows (Notton, Cristofari, & Poggi, 2006), (Noorian, Moradi, & Kamali, 2008) that the model of Perez (Perez, Ineichen, Seals, Michalsky, & Stewart, 1990) gives the most accurate results. This model however requires data (such as sky clearness and brightness) that is not available for this research.

Since there is little data available and diffuse radiation is more complex to predict in an urban environment due to for example reflection and limited visibility of the sky dome and different concentrations of aerosols. Because of this a simple model with low data requirements will suffice for this research, since no data is available to capture all the relevant factors for solar radiation calculations. This limits the number of models that can be used.

Based on literature (Noorian, Moradi, & Kamali, 2008) it becomes clear that all models investigated are biased, there is a significant difference in the root mean square and mean bias errors. Of the models that have a low data requirement and easy implementation the best model is a pseudo-isotropic model that partially corrects for the assumption that the diffuse radiation is isotropic (Koronakis, 1986). The formula to calculate the hourly diffuse radiation on a sloped surface is $I_{d,\beta} = \frac{1}{3} I_d (2 + \cos\beta)$
Where:

- $I_{d,\beta}$: diffuse radiation on tilted surface
- I_d: diffuse radiation on a flat surface
- β: angle of the surface

MODELING ASPECTS

To actually simulate insolation values in a geographic environment there are different approaches that can be taken. As indicated in the chapter 1 most research use a 2D (raster) or 2.5D approach, since this simplifies the geometric dimensions, allowing for easier and faster analysis. However the downside of this that there is an accuracy loss, and thus analysis in a 3D environment should provide better results. Working with 3D geometry does mean more geometric data and also more complex modeling. There are two approaches that can be used in to simulate shadows in a 3D environment in ArcGIS 10, both with their benefits and downsides.

The first approach consists of using the tools from the 3D analyst toolbox to create shadow volumes **(3,5)** based on the position of the sun **(1)** and the 3D model **(0)** itself. Once this is done you can use intersect 3D (3D analyst) **(7)** with the shadow volumes **(5)** and the roofs **(2)**, the output of this are roof segments that lie in the shadow. Using difference 3D (3D analyst) **(6)** you get the roofs that are not in the shadow. Afterwards you can merge **(4)** the two outputs and use it as roof input **(2)** for the next round of calculations (see Figure 15).Part of this method is also available at the ESRI resource center (ESRI, 2011)

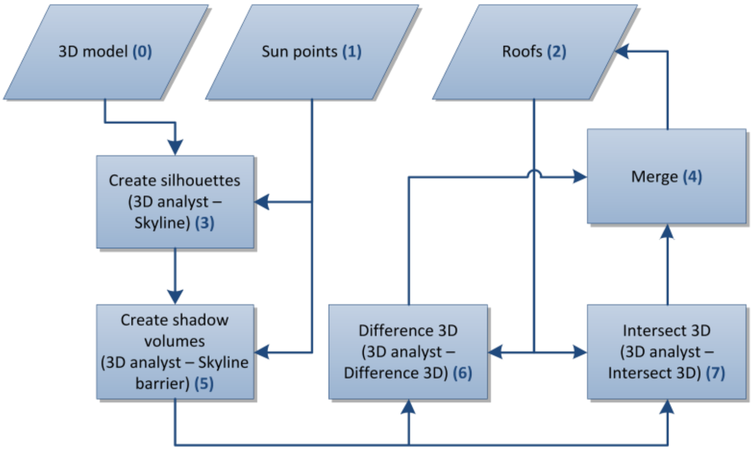


Figure 15: Polygon based approach

The benefit of this method is that it maintains the original shape of the roofs, since there are no dimensions lost. However there are two major downsides to this method. Firstly this method takes a lot of time to compute. Also due to the intersections every round of processing the polygon count of the roofs increase, which also increases the processing time with every cycle. Secondly shadows are created at the back (seen from the position of the sun) of the object it originates. Because of this the shadow cannot intersect with the polygon it originated from but it leaves a gap reducing the accuracy.

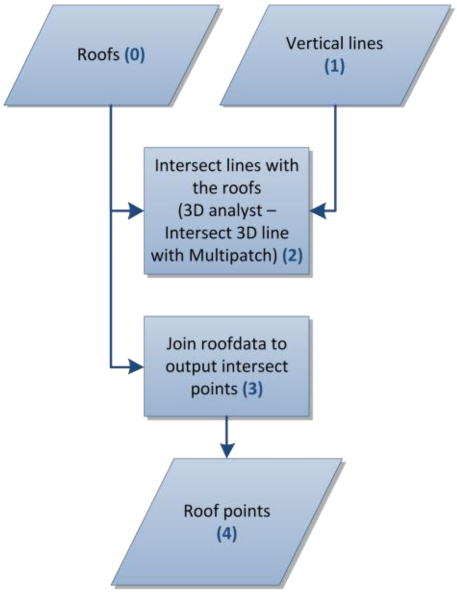


Figure 16: 'preprocessing' point based approach

The second method is to create a custom model to detect shadows based on points. This modeling approach benefits from the fact that points do not multiply when the model iterates and thus is also shorter. Also creating a custom script is can be faster than using predefined functions since it reduces the amount of overhead that usually comes with these functions. The downside of this approach is that is requires time to create the custom scripts and it loses some accuracy duo to the polygon to point conversion. This approach requires some extra 'preprocessing' (see Figure 16) to convert the roof surfaces to points. For this a custom script is created (see appendix 2). It stores a series of XY coordinates in memory based on an interval distance and then uses these coordinates to write a series of vertical lines **(1)** with a fixed height. After this these lines are intersected **(2)** with the roofs **(0)**, creating intersection points. These points get the attributes (angle and rotation) of their parent roof **(3)**, since these attributes are required in future steps. The results are points at the position of the roofs **(4)** with the attributes of the original polygons.

Once this 'preprocessing' is done the model start to iterate over each sun point (see Figure 17)**(1)** and does calculations in memory, no ArcGIS functions are used. Summarized every iteration does the following:

- Calculate the unit vector from the study area midpoint and the position of the sun **(4)**
- For each roof point **(2)** the angle of incidence is calculated using the sun azimuth, zenith and the roof angle and rotation **(6)**.
- Each roof point iterates over the roof polygons:
    - Each roof point iterates over the 3D model polygons **(0)**:
        - Per polygon the normal vector is calculated **(3)**
        - The intersection point between the line and plane is calculated **(5)**
        - Test to determine if the intersection point lies in the polygon **(7,8,9)**
- Once all roof points have passed all the roof points that do not have their ID stored in memory **(9)** get a value added to their total insolation value (irradiance from the sun * angle of incidence) and continue with the next sun point

Once all the iterations are done a final calculation is done to calculate the insolation of diffuse irradiance **(not in scheme)**. Since diffuse radiation is not dependent on the position of the sun but only on the orientation of the roof the yearly sum of diffuse irradiance is used. The sum of all insolation values is then taken and finally the data is written back to a shapefile containing the points with the calculated total insolation values **(not in scheme)**.

A graphic visualization of the output can be seen in Figure 19 (next page). As can be seen the lines towards the sun are parallel to each other, this is because the vector towards the sun is only calculated once from the midpoint of the study area **(4)**. A close-up on how the points are distributed can be seen in Figure 18. These figures are only illustrative of the intermediate results; only the final output of the model is required for this research.

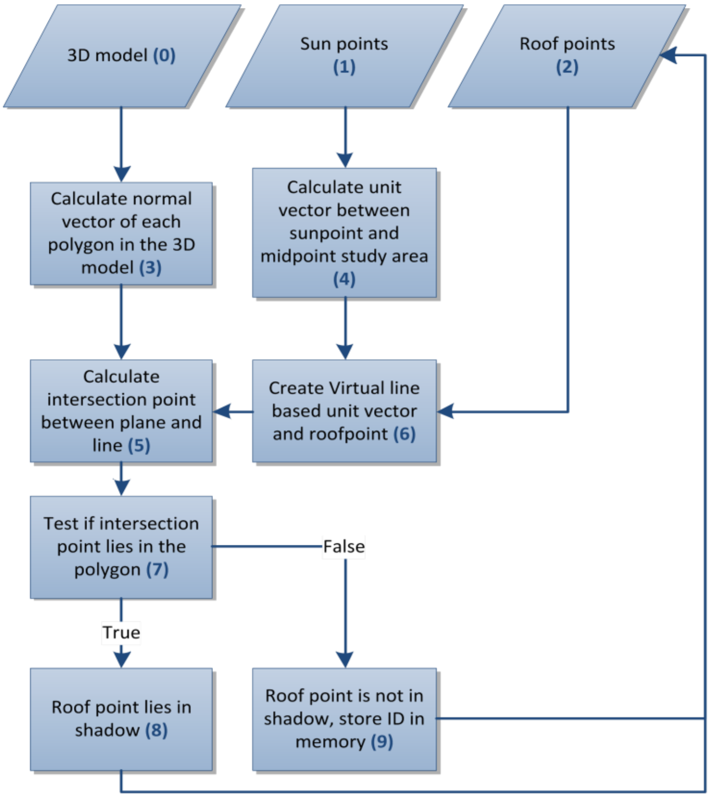


Figure 17: calculating insolation values point based approach

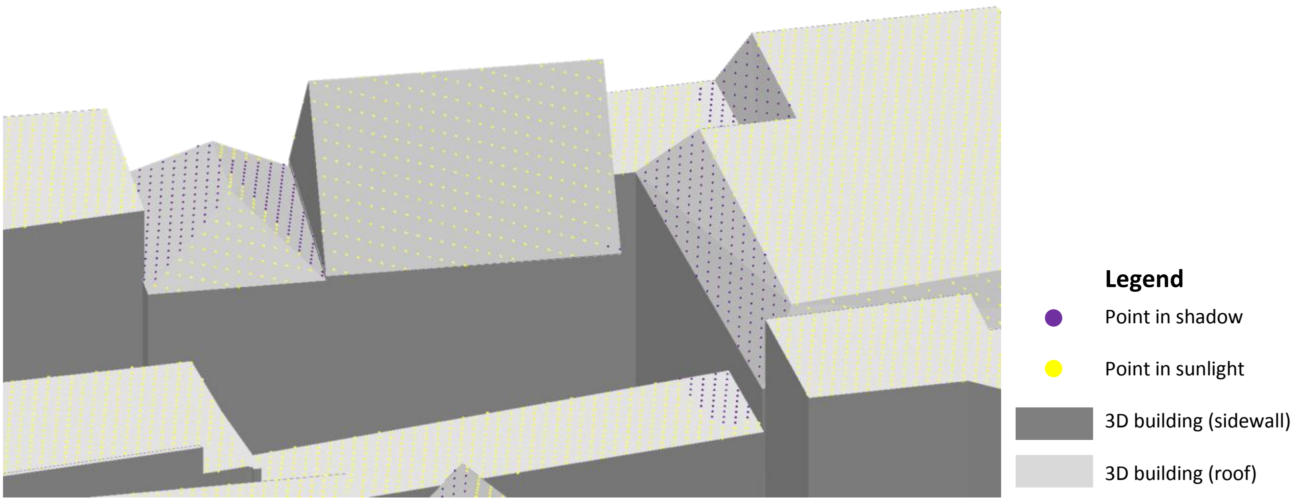


Figure 18: Close up model output (sun comes from the right)

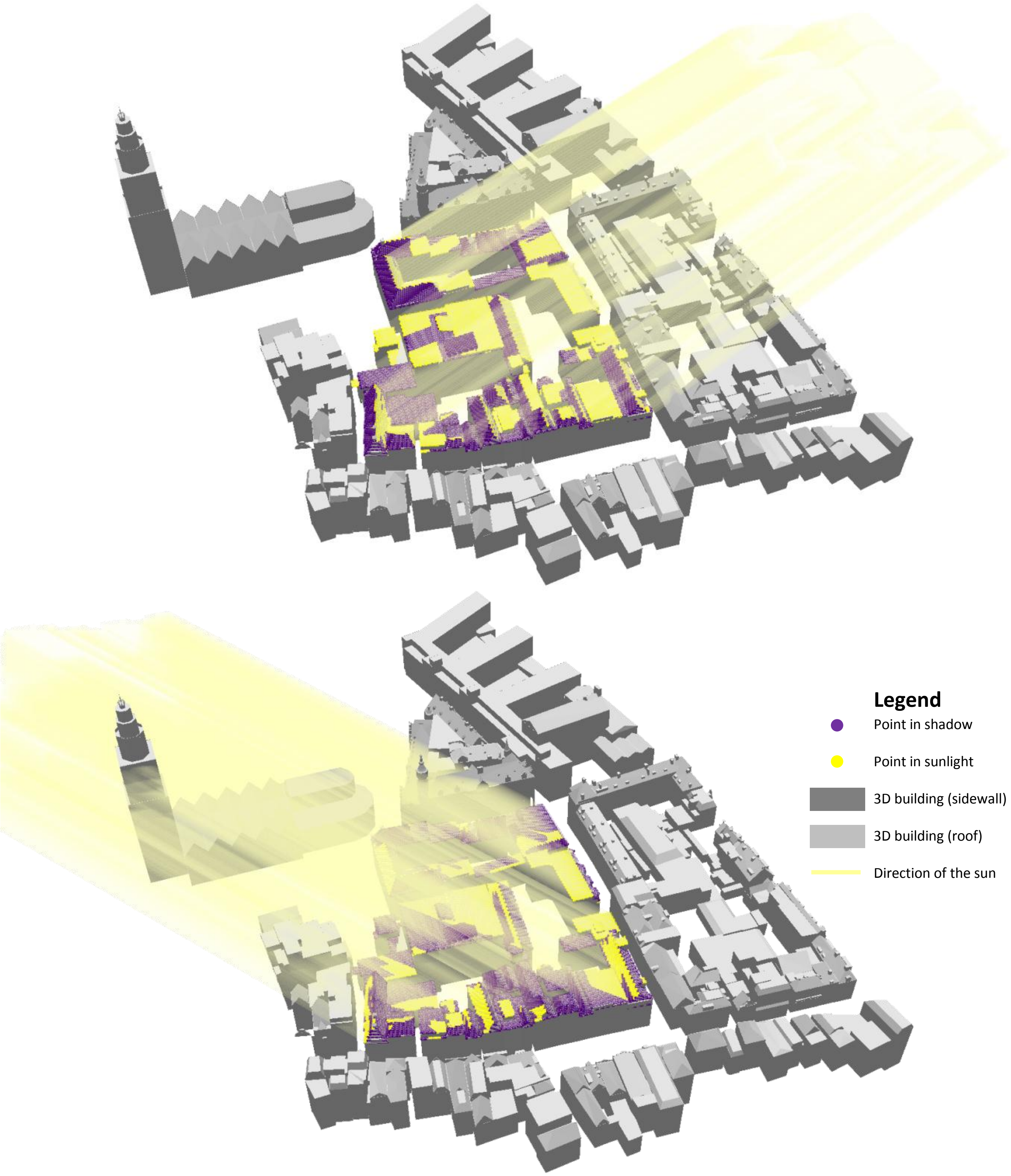


Figure 19: Shadow of 25 June 2007 at 5AM (top) and 8PM (bottom)

3 POTENTIAL ENERGY PRODUCTION

This chapter contains information about the potential energy production. First the optimal position for solar panels is determined (§ 3.1) so solar panels on flat roofs can be simulated at this optimal angle. Secondly the usable roof surface is determined (§ 3.2), after which the results of the simulation is used to calculate the potential energy production on the roof surfaces (§ 3.3). Finally the insolation values are translated to return on investment times, to determine if there is a realistic potential for solar panels based on insolation and finance(§3.4 & § 3.5). This chapter ends with an conclusion of the most relevant findings in this chapter.

3.1 OPTIMAL POSITIONING FOR SOLAR PANELS

The optimal position for solar cells is used to simulate solar cell positioning on flat roofs, since in most cases solar panels are placed at an angle on flat roofs. On the northern hemisphere the most obvious placement of solar panels on flat roofs is at an angle facing southwards, since the sun rises in the east, reaches his highest point south and sets in the west.

To determine the optimal position for solar panels there are several aspects of importance namely: the position of the solar panel (angle and rotation), the position of the sun in combination with the direct irradiance and finally the diffuse irradiance.

3.1.1 YEARLY PATTERN

Based on climatological data of 2007 insolation values are calculated for each possible variation in position of the solar cell. In Figure 20 the result of the calculations can be seen. The maximum yearly insolation value calculated is 785.8 kWh/m^2 and the minimum is 197.2 kWh/m^2. The highest values are approximately in the middle, but the values around an angle between 0° and 10° are also relatively high, making the shape of the distribution look as an inverse bell curve. This is mainly caused by the fact that 28,6% (Table 2) of the total irradiance is diffuse irradiance. Flat and near flat surfaces receive the highest amounts of diffuse irradiance. This is also apparent when looking at the formula for diffuse irradiance (see chapter 2.3.2).

Table 2: Division of irradiance types

Irradiance type	Amount of irradiance
Direct	253673,3 (71,4%)
Diffuse	101526,7 (28,6%)
Total irradiance	**355200**

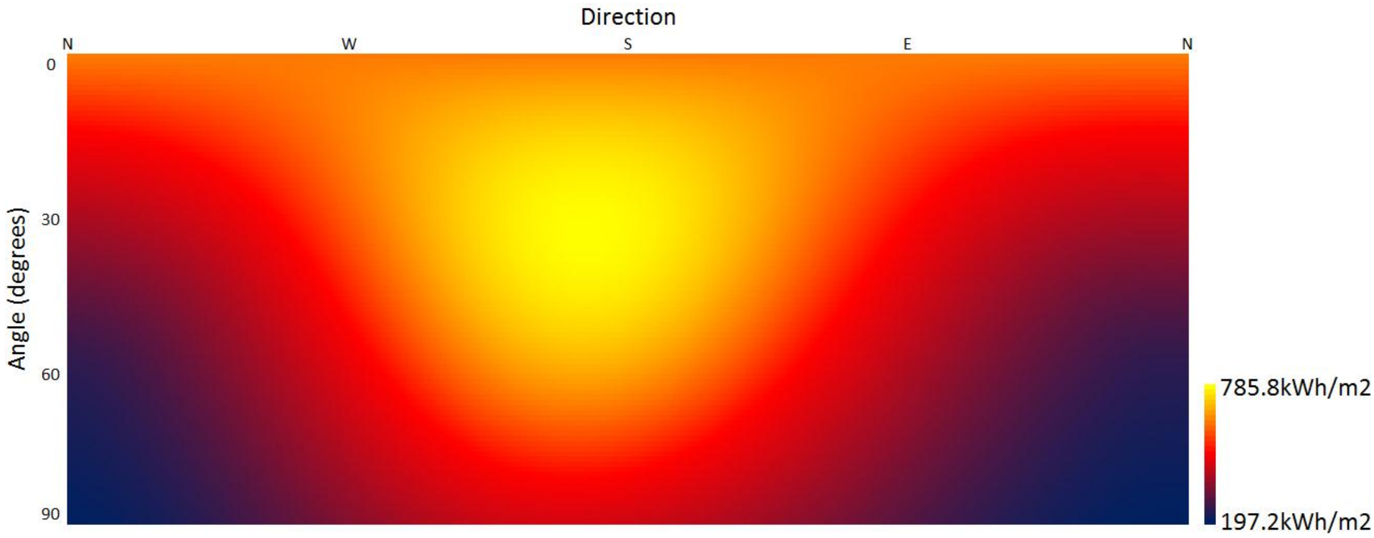


Figure 20: yearly insolation values for different positions of solar

The distribution of the results can be seen in Figure 20 and Figure 21. Different conclusions can be drawn from this data. Firstly surfaces with a low angle in general have higher insolation values because they are in direct sunlight most of the time and the contribution of diffuse irradiance is highest at low angles. The opposite is true for surfaces that have high angles. These surfaces often do not receive direct sunlight and less diffuse irradiance.

Secondly the 1% highest insolation values are not in the low angle range but their angle varies between 27° and 41°. Their rotation lies between 154°-180° (southwest – south). The 10 highest values angle between 33° and 35° and a rotation between 165°-169° (south-southwest).

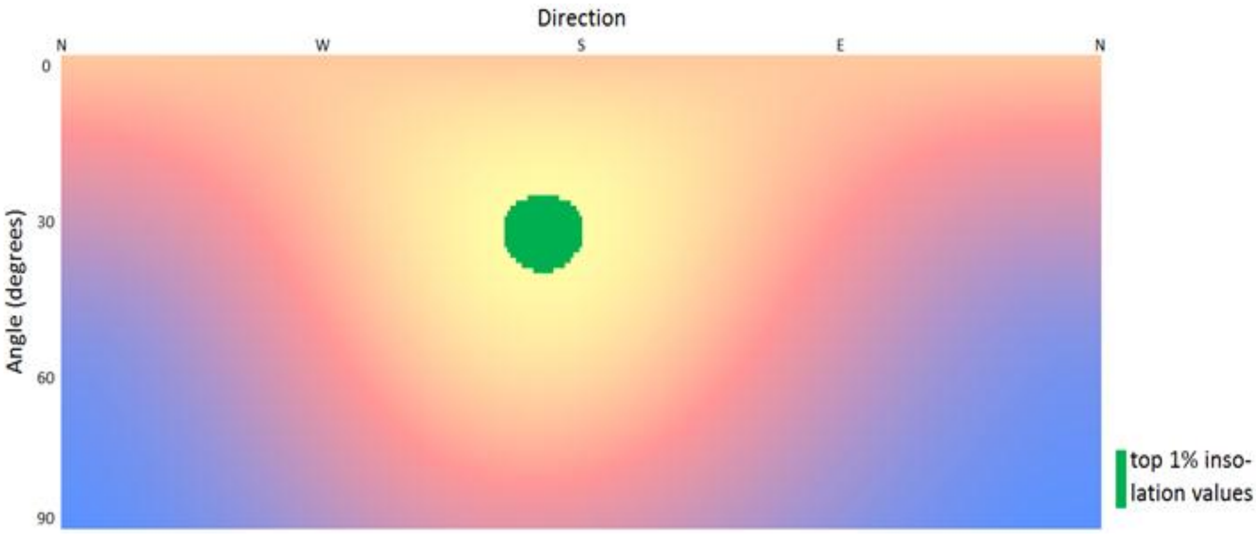


Figure 21:Position of the top 1% insolation values

Figure 20 exists of a total of more than 32.000 cells, each representing a variation in 1 degree in angle or rotation. When grouping this data in insolation ranges (see Figure 22) it is clear that the range of 650-700 kWh/m^2 is the most common range, this is mainly due to the positions with an angle around 5° and lower, these almost all fall in this range. Approximately 40% of all possible positioning of solar panels fall in the range of 650-800kWh/m^2.

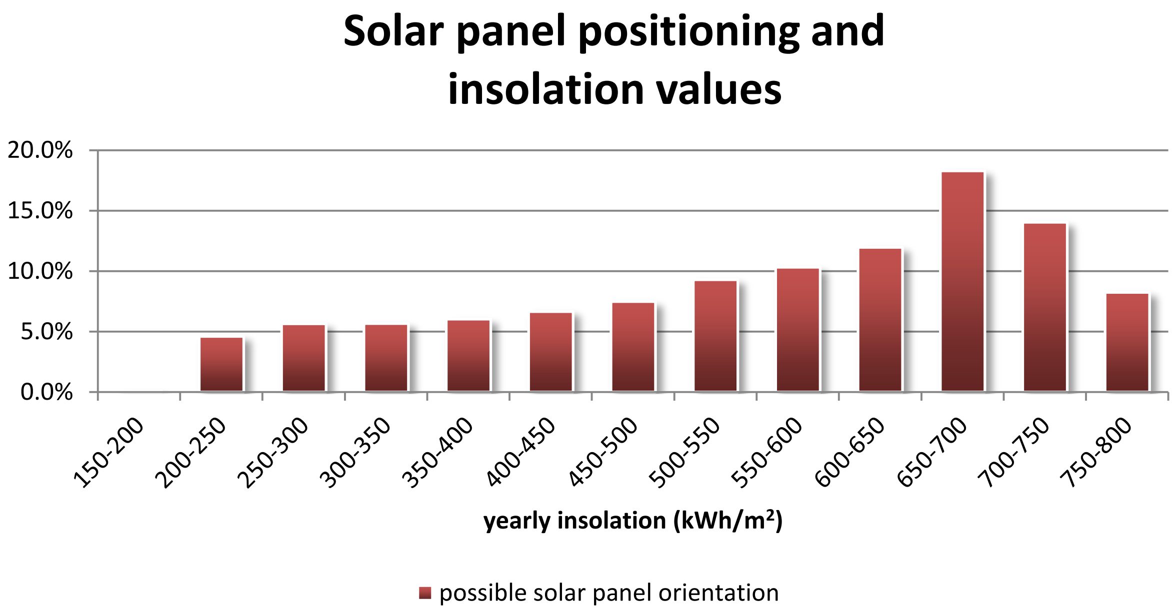


Figure 22: Positioning and insolation values

3.1.2 MONTHLY VARIATIONS

Since the position of the sun and the insolation values differ per month it is also important to get an insight in the monthly dynamics of insolation on sloped surfaces. In Figure 24 the distribution per month can be seen. As Figure 21 also here the green area represents the top 1% of insolation values.

It is clear that the optimal position differs per month. The greatest changes occur in the variation of the angle and not so much for the rotation. This effect can be expected when looking at the path of the sun of a year. In summer time the path is longer than in the winter times, but the highest position of the sun always is in approximately the same direction, but the angle does change significantly.

The range of maximum monthly insolation over a year is also striking, ranging from a mere 13,9 kWh/m^2 to 116,7 kWh/m^2, a difference of a factor 8. This difference can also be seen when looking at the monthly total insolation values (Figure 23). This indicates that there the influence of seasons is rather high and the majority of the yearly solar irradiation is actually gathered during the spring and summer. This is also why Figure 20 looks rather similar to the months April to August in Figure 24, since these months have a high yearly contribution.

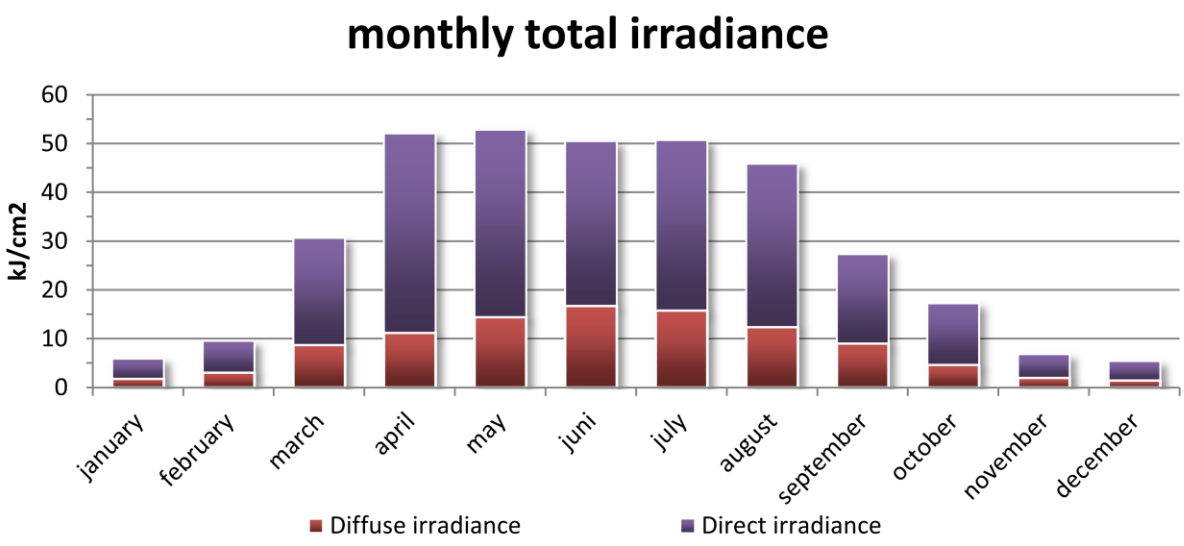


Figure 23: components of monthly total irradiance

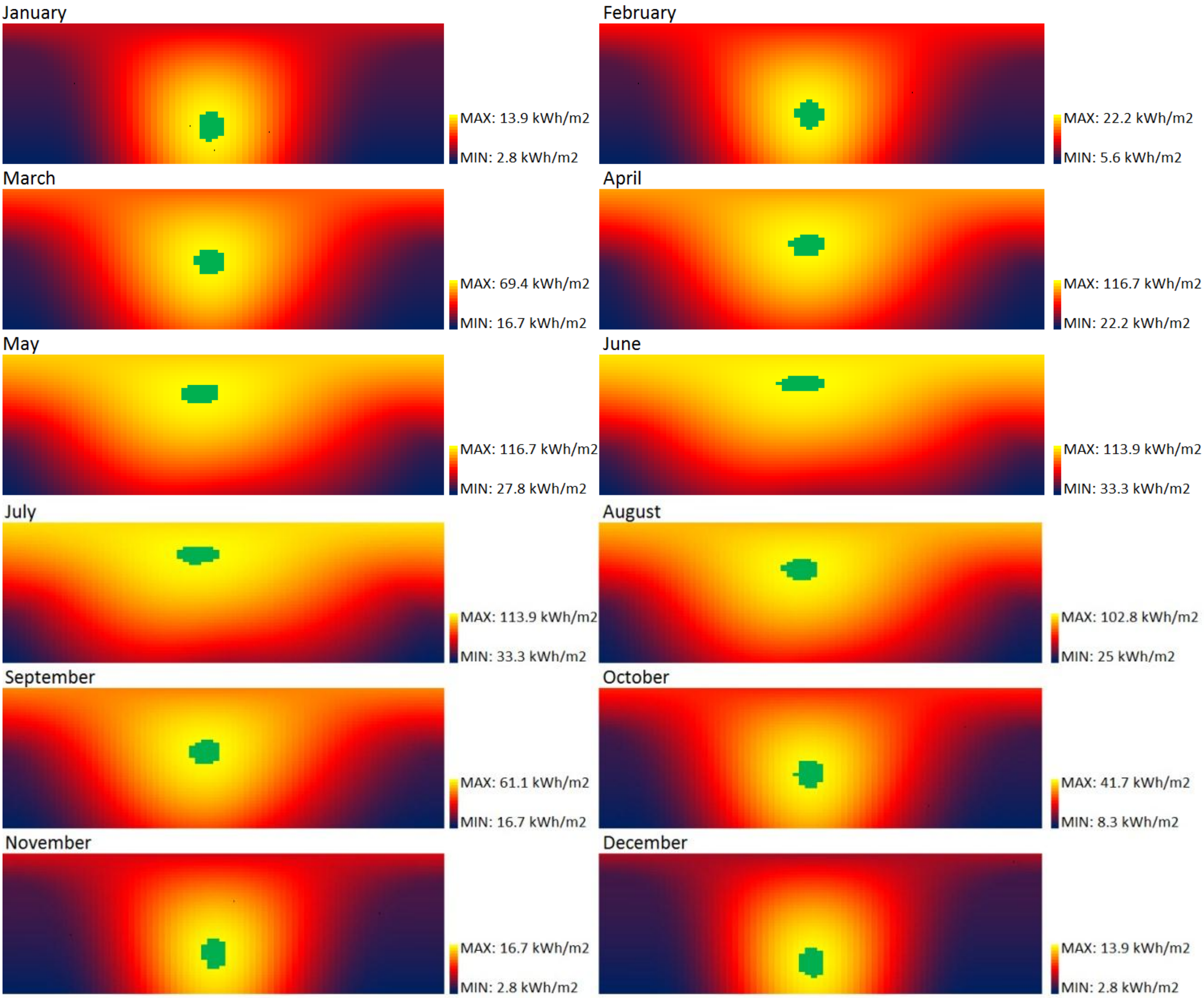


Figure 24: Monthly irradiance values on surfaces with different angles

3.2 USABLE ROOF SURFACE

Roofs have certain dimensions to be usable for the placement of solar cells.

- A roof has to have an area greater than $2m^2$
- The angle of the roof cannot exceed 75°
- A roof has to have a minimum height of 6 meters

Table 3: area suitable roofs

Flat(< 10°)	11490 m^2
Sloped(>= 10°)	2829 m^2
Total (<= 75°)	**14319 m^2**

The angle of the roof cannot exceed 75° because of two reasons. Firstly in most cases such roofs are technically unsuitable for solar cells and secondly based on the results from the previous paragraph such areas will not yield a great potential.
For flat roofs the most common placement of solar cells is not flat but at an angle facing south (180°). This special positioning on flat roofs is used for modeling the insolation. Based on the results of the previous paragraph the angle is 34° and the rotation is 167° (south-southwest) for potential solar panels on flat surfaces.

3.2.1 POINT REPRESENTATION OF ROOF SURFACE

For the simulation of the yearly insolation on the roof surfaces points are used. The polygons are converted to points by using vertical lines with an offset of 50cm. The intersection point of the lines and the polygons are the points that are used for the yearly insolation simulation.
In total there are 50.894 points. With the 50cm distance between each line one would expect that a point represents approximately $0{,}25m^2$ of the roof surface. When you divide total area with the number of points he outcome is actually ~$0{,}28m^2$, a difference of 12%. This is caused by two different effects, firstly the border of the polygons can cause issues and secondly there is an influence of the slope angle on the results.

The distance between points on sloped roofs is actually greater than 50cm, depending on the angle of the roof. The maximum possible angle is 75°. Based on the Pythagorean Theorem you can determine the length between the points; $\frac{0{,}50}{\cos 75} \approx 0{,}54$. So a point on a roof of 75° actually represents an area of $0{,}27m^2$ (0.54*0.50). This effect contributes to an larger area, but most roofs in the study area are actually (almost) flat and this effect is less significant at lower angles.

The border effect occurs at the edge of the polygons. The points here can represent a larger or smaller area as can also be seen in Figure 25. The blue area is this figure represents a flat polygon. The red area represents an area that is not accounted for by a point, the green area represents a point that represents a smaller area. This effect can lead to both an over- and underestimation of the area based on the points. In this research however this effect leads to an overestimation. Due to this effect there are fewer points than is expected based on the area of the polygon.

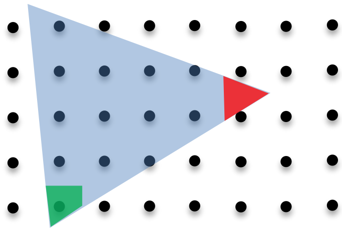


Figure 25: Border effect

3.2.2 ROOF SURFACE AFTER SIMULATION

For this analysis a total of 156 hours are simulated. One can safely assume that if the amount of hours a point is in the shadow of an object is equal or greater than 150 this point does not represent an exposed roof surface. Based on this assumption the number of points that remain as valid roof representations drop from 50894 to 29898, this means a 41,2% reduction in the number of points. Based on this new selection of points the total usable roof surface also drops from $14319m^2$ to 8412 m^2.

After visual inspection of the selection it appears most of the non-exposed roof points are identified. However due to the invalid geometry of some vertical polygons there are some points that, according to the model, have received direct sunlight especially when the sun was at a low angle. Due to the fact that there was another surface above these points the actual insolation values are too low to actually be considered a suitable area for solar cell placement.
There is also another theoretical possibility where a point could not be identified properly according to the model. When the point lies exactly on point of intersection of the potential roof polygon and another polygon (Figure 26), then this point would be treated as a point on the exterior of the structure.

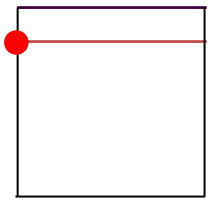


Figure 26: point at intersection

3.3 POTENTIAL ENERGY PRODUCTION

To calculate the energy generated by solar cells there are a few things to take into account. Firstly there are large differences between type and performances of solar cells. The performance is influenced by many factors, for example the temperature, internal resistance or spectral response. For this research using average values is sufficient, since only the potential has to be determined and not the actual performance of a specific solar cell. Secondly there are different methods to calculate the efficiency and energy output of solar cells (van Sark, 2007). The different methods do not yield great differences in results but do differ greatly in complexity. To calculate the output of a solar cell a static conversion efficiency factor is used. In reality however the efficiency can fluctuate based on the insolation values, temperature and other variables. The formula to calculate the yearly potential energy production per m^2 is because of this assumption rather straightforward

$$Y = \sum G\eta$$

- Y = annual energy(kWh/m^2)
- η = conversion efficiency
- G(h_i) = Hourly insolation (kWh/m^2)

Based on a small selection of solar cells (see Table 4) approximately average values can be determined to be used for this research. For this research conversion efficiency of 15% and a watt peak value of 150W per m2 is used.

Table 4: Solar cell specifications

Company	Product	Dimension	η (max at 1000w/m2)	watt peak (W)	Watt peak (W) per m^2
Suntech	STP250S - 20/Wd+ STP245S - 20/Wd+	1665mm × 991mm 1,65m^2	15,2% 14,8%	250 245	152 148
Sharp	NU-R240J5 NU-R245J5 NU-R250J5	1652mm × 994mm 1,64m^2	14,6% 14,9% 15,2%	240 245 250	146 149 152
JA Solar Holdings[3]	P6MF 3BB (01) P6MF 3BB (06) P6MF 3BB (11)	156mm x 156mm 0,02m^2	16% 17,6% 20%	3,83 4,24 4,48	157 174 184
Bosch[3]	M 3BB C3 1200	156mm x 156mm 0,02m^2	18,43%	4.44	182

Source: (Sharp, 2011) (Bosch, 2011) (Suntech, 2011) (JA solar, 2011)

Based on the sum of all the insolation values you can calculate what the theoretical maximum yearly energy production is. The yearly insolation average on all the roofs is ~632kWh/m^2. The minimum insolation is 231 kWh/m^2 and the maximum is 781 kWh/m^2. Based on a 15% total theoretical energy production in this area (8412m^2) could be almost 800.000 kWh (797.907,8 kWh). To put this value in perspective, the energy demand between July 2010 and April 2011 a large office building that houses the department Information and administration of the municipality Groningen is 945.702 kWh.

[3] These are single modules, when combined in a full panel the efficiency will be reduced

3.4 FINANCING SOLAR ENERGY

The financial aspect of solar energy is probably the single most important factor for public involvement. Firstly there is the investment which consists of purchase cost, installing the hardware and the loss of interest over the years. Secondly there is the profit you gain by reducing your energy bill.

3.4.1 COST OF MODULES

The cost of installing a solar panel module is mainly made up of purchase and installing costs. These can differ greatly however depending on for example the vendor or the quality of the product. Based on a retail price index (Solarbuzz, 2011) We assume that the price of a standard module is approximately €2,54 per Watt peak. There is no specific information on the cost of installing, but it is assumed that the cost of the module is approximately between 50% and 60% (Solarbuzz, 2011) of the total cost of a solar energy system. Based on this information one can assume that at this moment the total cost per Watt peak is approximately €4,60. This makes the cost of a module of $1m^2$ €690,-

3.4.2 COST OF ELECTRICITY

Based on data from Statistics Netherlands (CBS StatLine, 2011) the energy prices for consumers can be determined. The energy price differs slightly depending on tariff one has selected. Households that have a tariff for 3000kWh per year pay slightly less per kWh. Depending on the tariff a household would pay on average in the second quarter of 2011 between 0,252 and 0,287 euro per kWh. As can be seen in Figure 27 these figures did show a higher than usual fluctuation in the past three years. For this study an initial energy price of €0,25 is used.

The average yearly increase in energy prices can be determined by plotting a (exponential) trend line (R^2=0,96). Based on the formula for these lines the average increase in price per quarter is 1,5%, so the yearly price increase is $1{,}5^4 \approx 6\%$.

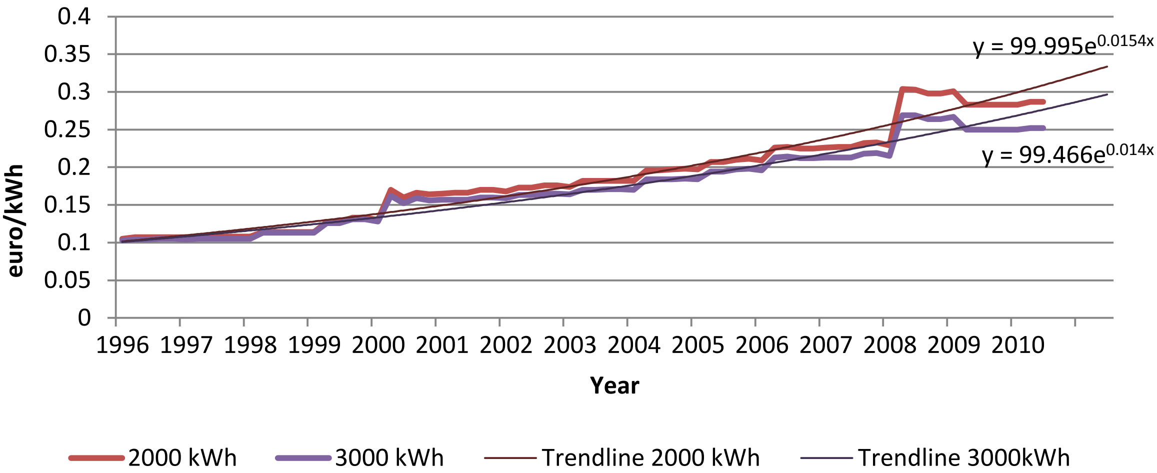


Figure 27: Yearly consumer energy prices

3.5 RETURN ON INVESTMENT TIME

In the previous paragraphs different variables are defined. These variables can be used together to make an estimation of how many years it will take for the Return on Investment (RoI). The following variables are used to estimate the RoI:

- Electricity price (€0,25 per kWh)
- Yearly energy price rate (6%)
- Cost solar module per m^2 (€690,-)
- Interest rate (2%)

The scenario used for the RoI calculations assumes that the investment is done with personal finance, leading to a loss of interest. If the money is loaned the interest will be higher, increasing the RoI time.

THEORY AND BACKGROUND

The interest rate is yearly applied to the cost of the module. The price of electricity is also increased yearly with the energy price rate. For calculation it is assumed that all the electricity that is generated by the solar panel substitutes the energy bought from the grid, reducing expenditures. When plotting the data (Figure 28) you get an idea of the time it takes to get a positive return on your investment. For each insolation value it is possible to calculate how long it takes for a positive RoI. So when you receive yearly 800kWh/year per m^2 on your roof you could potentially get a return on your investment after 19 years (see Table 5).

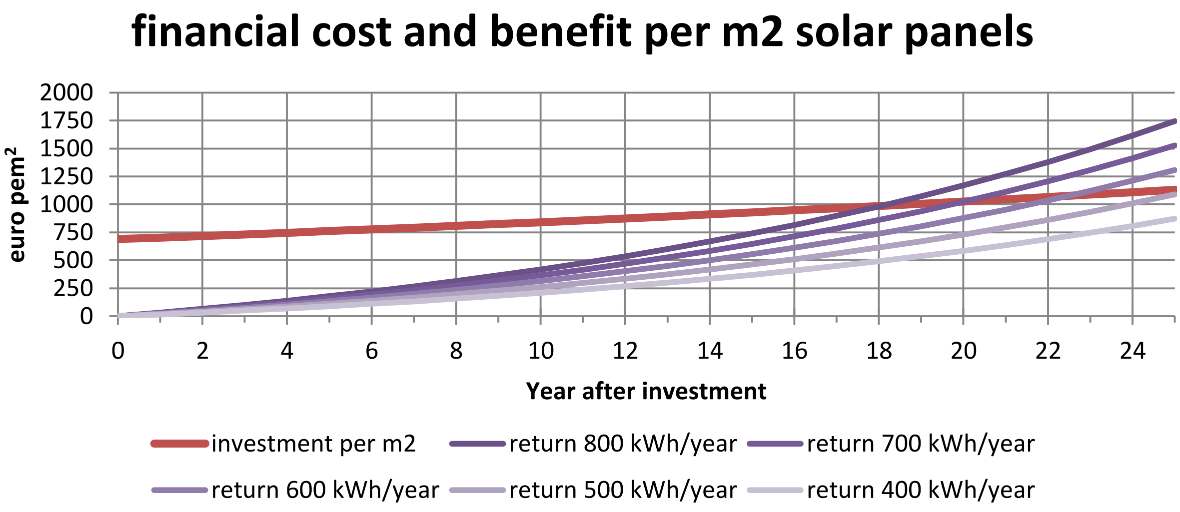


Figure 28: Investment versus return at different insolation levels

These figures however are highly dependent on the variables. The efficiency of the solar cell is not 15% but 17%, the RoI time comes closer to 16 years, the same happens when increasing the energy price rate with 1%.

The opposite also occur when changing the cost variables. If the cost of a module is increased from €690,- to €750,- the RoI time becomes 19 years. A 1% percent increase in the interest rate pushes the same RoI time towards 22 years. Future predictions are however always uncertain and the numbers represented here are the best estimate based on the data.

Table 5: RoI times

RoI time	Insolation (kWh/m^2) between
19	750-802
20	702-749
21	658-701
22	619-657
23	583-618
24	550-582
25	520-549
>25	<520

APPLICATION

When applying the RoI values at the study area it becomes apparent that the only areas that have a positive RoI before or at 20 years are the surfaces that have an optimal or near optimal angle and rotation towards the sun (see Figure 29).

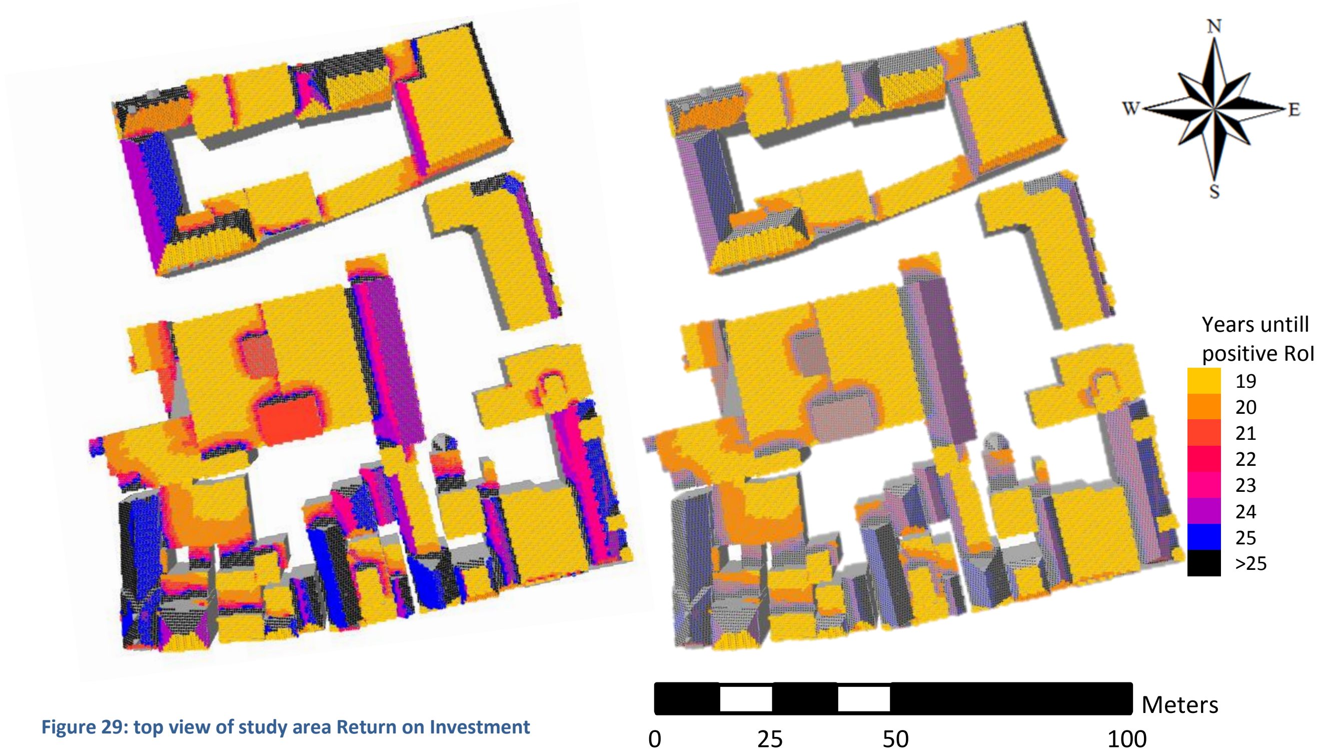


Figure 29: top view of study area Return on Investment

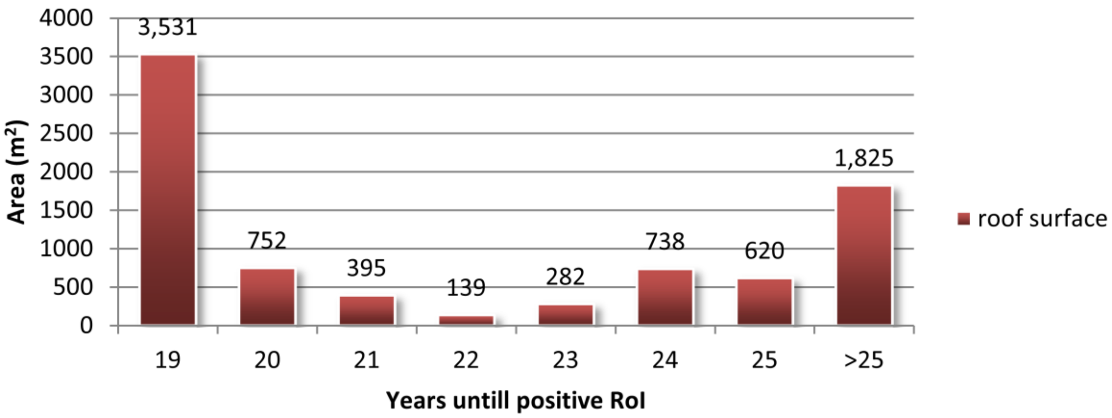


Figure 30: Area per Return on Investment time

The total area of potential suitable locations is 8412m^2. 1825m^2 (22%) roof surface will not achieve a positive RoI before the lifetime of a solar cell expires, in Figure 29 these areas can be seen.

Assuming that the public wants some financial benefit from their investment all areas with a RoI times longer than 20 years are not suitable for positioning solar cells with their current efficiency and price. In this study area that means that approximately 50% of the roof surface is suitable for positioning solar cells (see Figure 30). These surfaces are mainly the flat areas where an optimal position towards the sun can be achieved.
The total sum of insolation of this area is 3.281.929kWh, assuming that the whole area can be coated with solar cells the maximum energy potential is 492.289kWh

SHADOW AND ROI

A striking aspect of the areas that have a RoI time =< 25 year is how often they are in the shadows, which can be seen in Figure 31 (see also Table 5 for the RoI times). The general trend is as expected, higher yearly insolation values are less likely to be in the shadow of other objects. However until a yearly insolation of ~650kWh there are some areas that are in a shadow half of the time. It appears that besides the shadow the position towards the sun and the moments a surface does receive direct sunlight are of equal or greater importance.

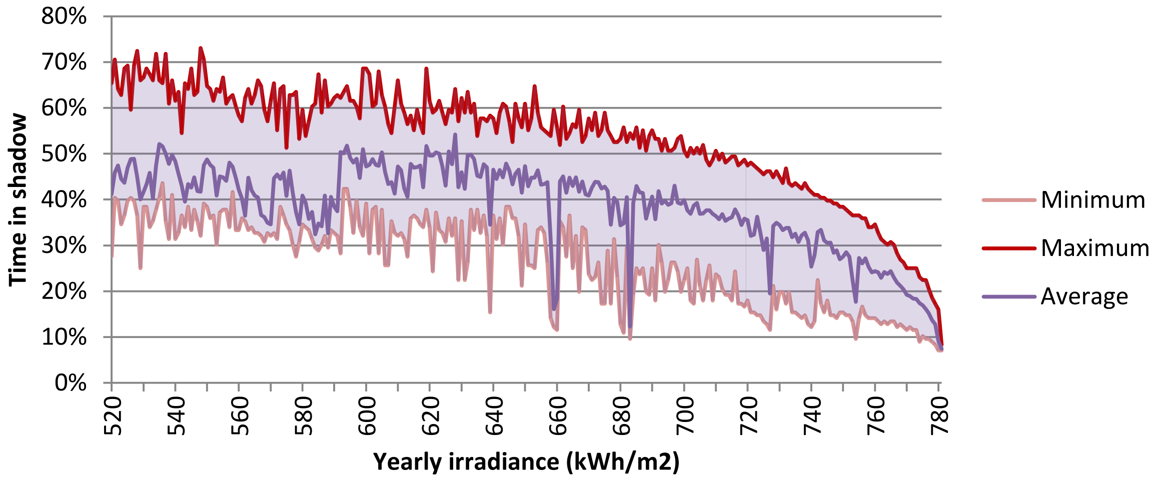


Figure 31: hours of shadow per insolation

3.6 CONCLUSION CHAPTER 3

Based on the data presented in this chapter there are several intermediate conclusions drawn.
Firstly there can be some conclusions drawn with regard to insolation, irradiance and solar cell positioning. It is calculated that the ideal position of solar cells, without taking the surroundings into account, is at an angle between 34° facing south-southwest (167°). Also it has become clear 40% of the possible positions a solar cell can have has relative high insolation values (>650 kWh/m^2) Additionally, there is a high difference in monthly irradiance values (ranging from 4,8 kJ/cm^2 to 42,5 kJ/cm^2) which leads to a high monthly variation in insolation values. The optimal position of a solar cell differs per month, but the change is mainly in the angle, the direction does not differ much over the month.

Secondly some conclusions can be drawn for the usable roof surface. Based on the criteria that a roof has to have an area greater than 2m^2, an angle between 0° and 75° and a minimum height of 6 meters an initial selection is made, with a total roof area of 14.319m^2. Some of this area is actually on the interior of a structure. After converting the polygons to points based on evenly spaced vertical lines (distance between lines is 50cm) it is clear that there are fewer points than can be expected based on the roof surface. After simulation of the insolation on the roof points a new selection is made based on the frequency of shadows to select only the points that lie on the exterior. Based on this new selection it is determined that 8412m2 of the initial 14.319m2 is actually an exterior roof surface.

Thirdly, the potential energy production is calculated. The sum of the hourly insolation per point (average is ~632kWh/m2) is used, together with an conversion efficiency (15%). This results in a total theoretical energy production of almost 800.000kWh if the whole usable area can be coated with solar cells.
Fourthly, the financial aspects are determined, resulting in a Return on Investment time. The following variables are used here: Electricity price (€0,25 per kWh), yearly energy price rate (6%), cost solar module per m2 (€690,-) and an Interest rate (2%). Based on these values it is determined that most flat surfaces (where the solar panels are set at an optimal position) can achieve an RoI time of 19 or 20 years. Most surfaces are able to get a positive RoI time within 25 years, however north and west facing sloped surfaces often do not achieve a positive RoI time within 25 years.

4 BALANCE BETWEEN DEMAND AND PRODUCTION

To determine the possible impact solar energy can have on the energy distribution grid a comparison is made between the energy demand and the potential energy production (§ 4.1). Additionally the interaction between energy demand and production is further illustrated (§4.2).

4.1 ENERGY DEMAND

The exact energy demand in the study area cannot be determined precisely unless one has direct access to the databases of the energy providers. To make an estimation of the energy demand a website (Liander & Enexis, 2011) is used that presents aggregated data for municipalities to use. This website allows the user to view and download yearly data of electricity and gas use for different aggregation levels, ranging from 6digit postal codes to a whole municipality. Energy use data on 5 and 6 digit postal codes however is often not available or shielded from view; the numbers of objects however are available. The data for the municipality of Groningen is not available for the year of 2007, so data of 2008 is used.

Based on a selection of the 6 digit postal codes (see Figure 32) it is determined that the study area consists of approximately 68 real estate objects (each object has a unique address), 34 business properties and 34 private properties. This is only an estimate, since the exact locations of the properties are not known and the postal codes do not fit exactly with the study area.

Since some of the postal code areas do not contain data on energy use these have to be derived from the aggregated data on neighborhood level. The energy use in the neighborhood can be split up in two, namely for energy use of businesses and private property. In 2008 there were 462 businesses with an average yearly electricity use of 55.566kWh and 843 private properties with an average yearly electricity use of 2.742kWh. These numbers are only an average based on the count of properties and do not take into account other factors such as the size of the property.

Based on this data the expected energy use can be calculated for the study area by multiplying the average energy use with the count of the properties. The estimated energy demand for private properties is 93.228kWh and for business properties 1.889.244kWh.

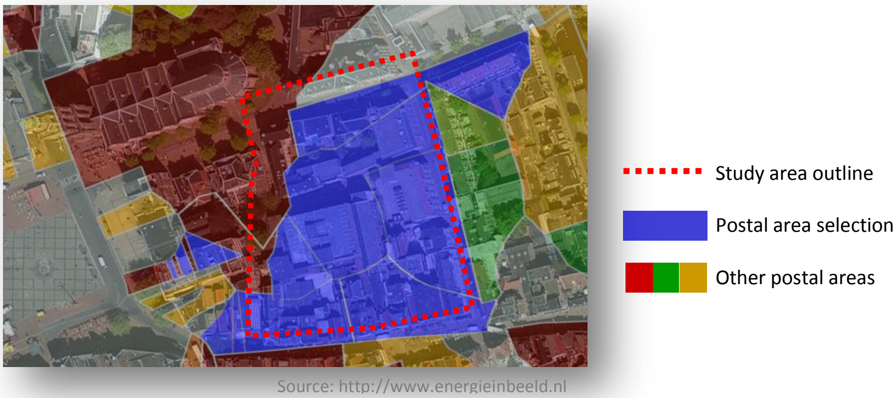


Figure 32: 6 digit postal code area selection

4.2 ENERGY DEMAND VERSUS POTENTIAL PRODUCTION

Based on the previous chapter and paragraph the energy demand and production potential is determined (see Table 6)[4].

It is clear that the highest energy consumers in this area is linked to the business properties and that the private properties have a relative low energy demand.

Table 6: Energy demand and production potential

Total energy demand private properties	93.228 kWh
Total energy demand business properties	1.880.244 kWh
Total energy production potential (Chapter 3.3)	797.908 kWh
Total energy production potential with RoI time =<20	492.289 kWh

If you assume that 10% of the area with an RoI time =< 20 years (regardless of to which property a roof belongs) will be used for solar cells and all solar energy can be used that still means a reduction in energy demand of 52% from private properties. But when taking in account the total estimated energy demand in the study area only a reduction of 2,3% can be made with the same solar panels.
These values are not representative for the whole city; however inner city areas have a higher number of commercial properties which affect these numbers. This means that in residential areas solar energy can have a high impact on the energy demand.

FLUCTUATIONS IN TIME AND GRID LOAD

Since there are only yearly estimates on the energy demand available it is not possible to estimate the effect of fluctuations in energy demand and production on the electricity distribution grid. Literature provides some

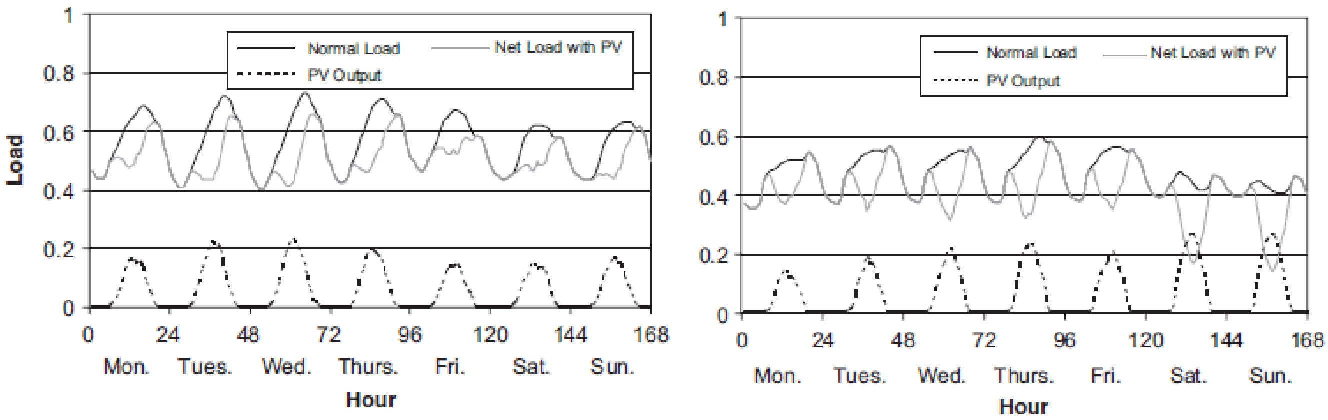


Figure 33: Coincidence between production and demand (left June 2000, right March 2000)

Source: (Denholm & Margolis, 2007 (a))

insight in the hourly coincidence between energy demand and solar energy production (Denholm & Margolis, 2007 (a)) (Denholm & Margolis, 2007 (b)). These studies[5] indicate that there is a coincidence in the summer times, but in the other months this coincidence is less or non-existent. In Figure 33 the effects of 20% energy generation from solar energy are shown, where:

- *normal load* is the energy demand without solar energy
- *PV output* represents the solar energy production
- *Net load with PV* is energy demand of conventional energy sources taking into account solar energy

When there is no coincidence the fluctuations in demand from other energy sources fluctuates more. Whether the system can cope with this depends on the flexibility of other energy producers. Nuclear energy plants for example have a low flexibility because the change in their energy output is slow. On the other hand, hydroelectric plants can change their energy output rather fast. Theoretically a system with 100% flexibility can fit the energy production exactly to the energy demand; however such systems do not exist yet. All

[4] This value is an overestimation due to the special conditions used on flat surfaces, also see chapter 5.2.
[5] This research is done for an area in the USA. It can be assumed that the patterns here in the Netherlands are similar, but this paragraph is only illustrative for the impact solar energy can have on the distribution grid.

systems have a limited flexibility, meaning that the synergy between energy demand and production is not perfect.
Based on this information it is clear that the potential of solar energy is limited when an energy system has limited flexibility. The more flexible a system becomes the more solar and other variable energy sources (such as wind) can be applied without too much loss of energy, which is illustrated in Figure 34. This figure shows the minimum energy output from conventional energy sources (*Minimum Load*) and the area where there is excess solar energy (*Surplus PV*). You can see two scenarios; the right one represents a situation where 9% of the annual energy demand is fulfilled by solar energy, the left represents a scenario where 18% of the annual energy demand is fulfilled by solar energy. An energy system that is not 100% flexible will always provide a minimum amount of energy, all excess solar energy is then wasted and either has to be stored or will go to waste.

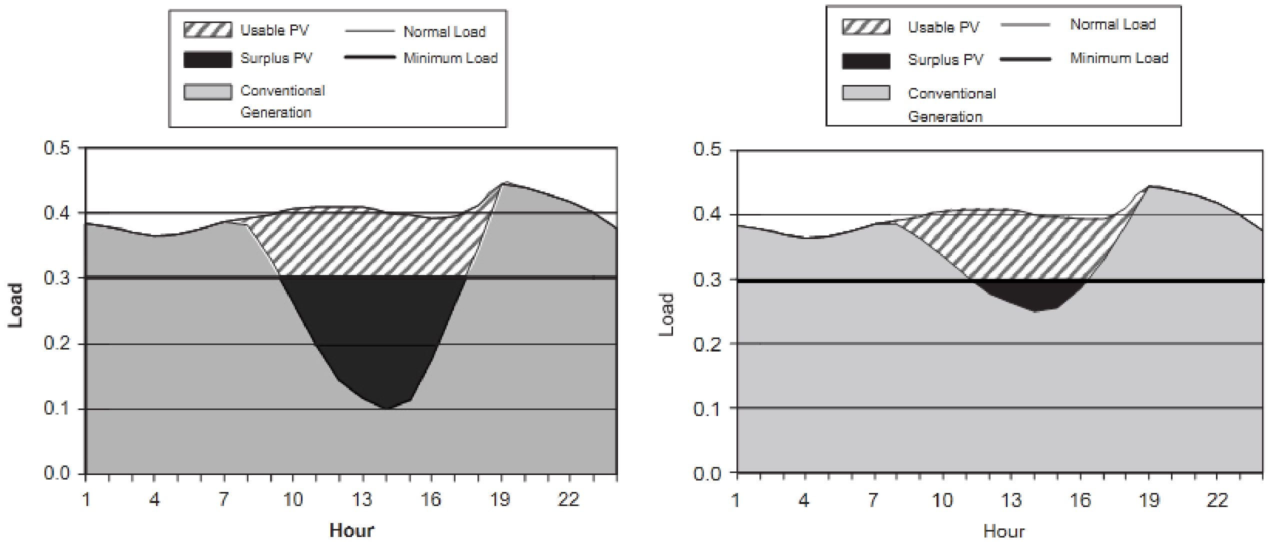


Figure 34: Usable and excess solar energy (left 18%, right 9%)

Source: (Denholm & Margolis, 2007 (a))

4.3 CONCLUSION CHAPTER 4

Based on the previous chapter it is already determined that the total potential energy production is 482.289 kWh. The energy demand is derived from data with different spatial resolution and the estimated energy demand for private properties is 93.228kWh and for commercial properties this is 1.889.244kWh. When 10% of the energy production potential with a RoI time <=20 years is realized and only applied to private properties this means a possible 52% reduction in energy demand.

The possible energy reduction should be seen in context however of the larger picture. When there is no storage of solar energy the produced solar energy should be directly applied, there has to be a coincidence between solar energy production and energy demand.

Due to the lack of data it is not possible to make an accurate estimation. Literature however does show that there is a coincidence during the summer and less or none coincidence in the winter. Based on the literature the possible 52% reduction in energy demand will require a high level of distribution grid flexibility to lack of coincidence and system flexibility. It can safely be assumed that the produced solar energy in this scenario will go partly to waste. Further study is required however to conclude anything conclusive.

5 DISCUSSION

In this chapter different aspects of this research are critically reviewed. First the input data is reviewed (§ 5.1), followed by the modelling (§ 5.2), The results discussed in chapter 3 (§ 5.3) and chapter 4 (§ 5.4) are also reviewed on factors such as (in)accuracy or errors.

5.1 INPUT DATA

This research uses three different source datum, namely the 3D city model, climatological data from the KNMI and climatological data from the WUR.

The 3D city model has a centimetre accuracy according to the original creators (municipality Groningen). Visual inspection of the model shows no obvious deviations in the geometry of the objects. the exact position of different elements cannot be verified since there is no reference model. The footprint of the buildings fits well with other sources such as ArcGlobe layers, so the spatial accuracy of the footprint is good.

Due to the multiple conversions the original model underwent there are some corrupted polygons. These are not read by ArcGIS for analysis and thus are not taken into account. On some locations this is an issue, however the influence of this is small.

To get local diffuse and direct irradiation values two datasets are used, one local dataset gathered near Groningen (KNMI) containing absolute irradiance values and one dataset collected in Wageningen (WUR) containing both direct and diffuse irradiance. The ratio of direct and diffuse irradiance is directly applied to the data in Groningen, not taking into account factors such as cloud cover. On the short time this leads to numbers that do not necessarily represent the actual climatological situation on the location, over the long time it is expected that the errors will even out. This however is not confirmed and further analysis could be performed on this subject. This is also true for the difference between the climate of a city that of a rural area nearby. It is known that the temperature and wind condition differ, which in term affect the amount of aerosols in the air. Aerosols also influence the ratio between direct and diffuse irradiance. High buildings in the area also influence the sky view and thus the ratio between direct and diffuse irradiance.

5.2 MODELLING

The modelling part consists of two parts, firstly creating new data and adding attribute data, secondly the simulation of insolation.

For the first part the position of the sun is determined based on the centre of the study area. The position of the sun relative to a position on earth is not fixed, however the change in position is minimal for the size of the study area so one sun position suffices.

The simulation of the insolation from direct irradiance (which accounts for ~71% of the total irradiance) is accurate, assuming the direct sunlight can be represented by a beam. The simulation of insolation from diffuse irradiance (which accounts for ~29% of the total irradiance) it not 100% accurate as can also be seen in chapter 2.3.2. Another factor is that the diffuse models assume that the sky view is clear but in urban environments however this rarely is the case. Further research should be performed to determine what the influence of a limited sky view is on the diffuse irradiance.

For this research irradiance values are simulated per four weeks due to time constraints, so one simulated day actually represents four weeks. The position of the sun changes significantly in these four weeks, so for future work a finer temporal resolution is recommended.

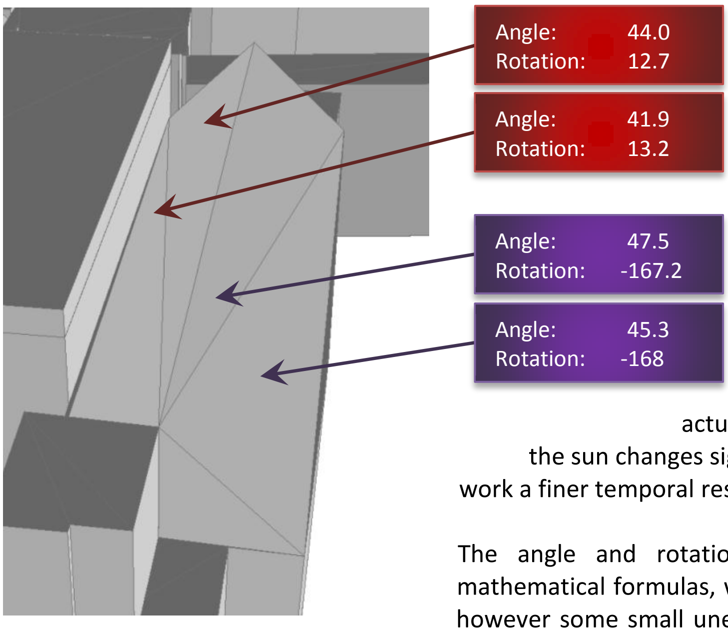


Figure 35: Error in roof calculations

The angle and rotation of roof surfaces is calculated using mathematical formulas, which are assumed to be correct. There are however some small unexpected errors. It is expected that each of

the two triangles that make up a roof share the same angle and rotation. Random visual inspection of the data shows that this is not always the case (see Figure 35); the reason of this is unknown. To get an idea on the impact of these deviations on the final result one firstly has to look at the effect of 1° variation in angle and/or rotation is. This is easily done by calculating the average of the absolute difference between cells. So for each cell from Figure 20 (page 20) the average difference with the eight neighbouring cells is calculated. This data can then be plotted again in similar fashion, as can be seen in Figure 36. Here can be seen that the difference per degree in variation is limited and highly variable depending on the positioning of the surface.
Additionally, the insolation values are also aggregated to a Return on Investment (RoI) time. Each aggregated group has a spread of ~50kWh/m^2 (see Table 5, page 27), so even when there is a deviation of several degrees will have a limited impact on the results.

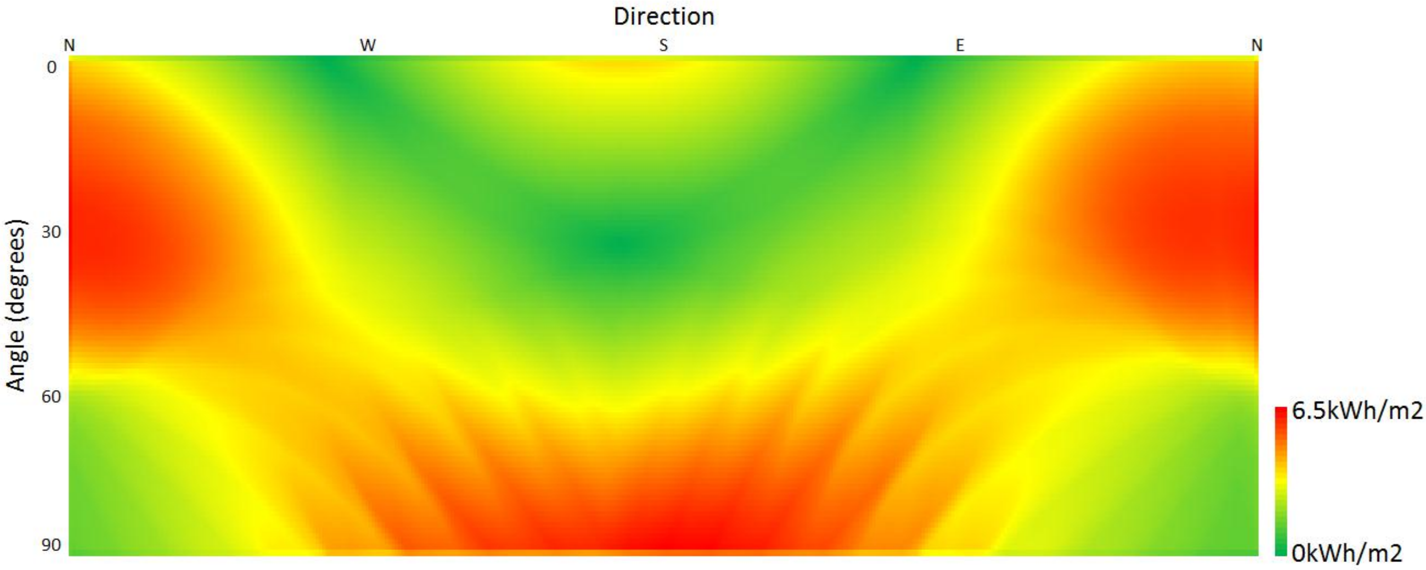


Figure 36: Average absolute difference per 1° variation in angle and/or rotation per year

For analysis the roofs are converted to points. This makes sure that the number of geometries does not increase with every iteration, there are however some consequences of this action which are also discussed in chapter 3.2. To create the points that represent the roof the intersection point of evenly spaced vertical lines and roof polygons is taken. This means that the shape of the polygon is not taken into account. The distance between points on sloped surfaces is larger than the distance of points on flat surfaces. Some problems can also occur around the edges of polygons, similar to those when converting polygons to raster. The first issue can be countered by using a different algorithm that reads the shape and determines where the points should be placed, but the last issue is a problem that can only partially be countered.

The second part of the modelling is the simulation of the shadows and sunshine on the roof points. The model as a whole functions as desired, there are however some issues regarding the accuracy.
First off, the sun is represented by a point. This makes calculations easy, but the sun is actually a circle. This can cause errors when the sun is partially visible, for example when the sun rises just partially above a structure. If a point on a roof can only view the sun partially, the model indicates that that point is either direct sunlight or not, there is no middle ground.
Secondly, as noted before, some polygons have invalid geometry that causes them to be ignored by the model.
Thirdly, insolation values on a flat roof are simulated as if the surface is in the optimal position towards the sun. This is the most common positioning of solar panels on flat roofs, it does however reduce the actual available roof surface, since the area directly behind a tiled panel is not suitable for solar panels. This effect is not taken into account when calculating the available area or total insolation values.
Furthermore, since the irradiance values are aggregated to a four week period there is a slight inaccuracy in the simulated values, since every day the position of the sun and the irradiance values differ.
Finally, only for specific points the potential is calculated, a solar panel overlaps multiple points. The effects of for example different insolation values or partial shadows on a solar panel are not taken into account.

5.3 SOLAR CELL POTENTIAL AND RETURN ON INVESTMENT

For determining the solar cell potential first the yearly insolation value is used together with a conversion efficiency factor to get a potential energy output. After this the return on investment time is calculated to determine if it is possible to get a financial benefit. As already noted in chapter 3.3 the assumption that the conversion efficiency is a fixed number is not correct. To get a generic overview of the potential however it is sensible to use a static factor here.
For the RoI calculations the output of the model is used. Any inaccuracies in this model will propagate to the RoI calculations, the exact effects of this are not clear. The RoI calculation is a method of aggregating the data to groups, and this means that small inaccuracies either are removed or lead to a shift in RoI times which is assumed will not yield a difference of more than a year.
The figures used for the RoI calculations are the most reasonable approximations for this moment. The RoI is a calculation that assumes that over a longer period of time a trend exists. If in the future big financial changes occur, such as the current economic crisis, these calculations do not hold.

Some figures are expected to change in the near future. Firstly the conversion efficiency shows a trend of increasing continuously, and it is expected that this will continue for now. Secondly the cost of a solar panel module shows a trend of decreasing prices, this is also a trend that will continue. If this trend indeed does continue the RoI time will decrease. The extent of this decrease is not clear, but chapter 3.5 gives impression on the impact of different values on the RoI time.
The initial investment is assumed to be done without subsidies and with a person's own savings, losing an interest rate of 2% yearly. If there are subsidies, or if the money is lend the numbers will differ greatly, of this the impact is also not clear. It should be noted in some cases the applications showing information on roof mounted solar allow their users to insert their numbers on finances (Municipality Assen).

5.4 ENERGY DEMAND AND PRODUCTION

The comparison between energy demand and production is a difficult one since the study area is not a conventional residential area. The high difference in energy use between commercial and public properties in combination with the fact that the numbers applied are neighbourhood averages (~1300 properties) of a heterogeneous neighbourhood makes the numbers only rough approximations. Also the actual numbers of properties in the study area is not certain. This uncertainty does not renounce the fact that there can be a significant impact on the energy systems when a portion (10%) of the surface is actually used for solar panels. Also literature clearly indicates that at high level of solar energy penetration much of this energy could go to waste if the electricity system as a whole cannot relocate this energy.

6 CONCLUSION AND RECOMMEDATIONS

This chapter contains the conclusion and recommendations. The conclusion paragraph presents the most important conclusions that can be drawn from this study. The recommendations look forward to possible future research to continue and improve this study.

6.1 CONCLUSION

The main research objective of this thesis research is to determine and apply a method for solar energy production potential estimation based on 3D virtual city data and return on investment calculations. The method used is this research benefits from the high spatial accuracy that can be achieved by using 3D polygons instead of a raster. Two benefits of using raster (faster processing and simple geometry) are achieved by using points instead of polygons, while maintaining the high spatial accuracy from the 3D polygons for simulating insolation values on roof surfaces. A polygon based method would suffer too much from splitting a polygon in multiple polygons, making the process slower with each iteration. The downside of this approach is that exact surface areas cannot be retrieved based on the points.
Concluding from this you can state that using 3D geodata does lead to better spatial accuracy over raster datasets, especially analysis in complex urban environments can benefit greatly from 3D geodata. The main downside is that processing time increases when data becomes more complex, but this issue becomes less relevant each year.

The goal of this research is to answer two main research questions. These research questions are:

1. What are potential locations for local solar energy production in an urban environment?
2. What is the predicted impact of local solar energy production on the existing infrastructure?

To answer these two research questions a total of ten detailed research questions are formulated (§ 2.1). Each of these detailed research questions (DRQ) is answered in chapter 3 and 4. Then numbering will refer back to the detailed questions as formulated in § 2.1.

The insolation is dependent on the orientation of the solar panel, the position and irradiance of the sun in combination with the surrounding buildings. The hourly metrological conditions on the location (DRQ 1.2) are derived from two data sources. Data measured near the study area (distance of 10km) only contains Total irradiance values, while for this study the two components that make up this irradiance (direct and diffuse) are required so that different climatological conditions are taken into account (DRQ 1.3). To get this data the ratio between direct and diffuse irradiance measured in Wageningen by WUR is applied to the data of the KNMI. The spatial difference between the two locations does lead to less accurate results in comparison with in situ measurements, since the metrological situation can differ easily between the two locations. Expert opinion is that the errors this causes even out over time, this is however not validated by measurements.

The ideal orientation of a solar cell, without taking into account the surroundings, is calculated to be at an angle between 34° facing south-southwest (167°). It is also determined that surfaces with an angle > 75° have low insolation values (DRQ 1.4). This ideal orientation is applied to flat roofs, since on these roofs solar panels are often placed at an angle.
The roof areas are selected based on the criteria that the area should be at least $2m^a$, have an angle less than 75° and have a height of at least 6 meters. To make this selection first the angle, rotation and height are added as attributes to the polygons by using custom scripts that calculate these based on the normal vector of the polygons (DRQ 1.1). Based on this selection $14.319m^2$ of roof area was selected, of which $8412m^2$ was determined to be actually exposed roof surface.

The hourly conditions are simulated for 14 days in a year to determine the yearly insolation values on the roofs (DRQ 1.6). The simulation takes the surrounding buildings, roof orientation, diffuse and direct irradiance into account. Insolation values range from $231kWh/m^2$ to $781kWh/m^2$, on average the insolation value is $632kWh/m^2$ (DRQ 1.6).
The simulation is also set up to count the number of times a surface is not in direct sunlight (DRQ 1.5). The general trend is as expected, higher yearly insolation values are less likely to be in the shadow of other objects. However until a yearly insolation of ~650kWh there are some areas that are in a shadow half of the time.

Based on literature the average energy conversion rate for solar panels is determined to be 15%, so 15% of the insolation is converted to electricity (DRQ 1.7). Based on this electricity production the Return on Investment (RoI) time is calculated, which indicated from which year a possible investment in solar panels starts to give a positive return (DRQ 1.8). For this calculation the cost of a solar panel is determined to be €690,- per m^2, electricity price is €0,25 per kWh, with a yearly increase of 6%, and an interest rate of 2%.

The fastest RoI time is calculated to be 19 years (see Table 5), the surfaces in question are mostly the flat surfaces where the optimal position for solar panels is used (see also Figure 29 and Figure 30). 22% of the roof surface will not achieve a positive RoI time within 25 years. Not all flat surfaces however have similar RoI times due to the surrounding structures.

The energy demand in the study area is derived from yearly data on the level of 6 digit postal codes and neighborhoods. The energy demand for private properties is 93.228kWh and for business properties this is 1.889.244kWh. With 10% of the roof surfaces with a RoI time <= 20 years used for solar panels it is possible to generate energy that can compensate more than 50% of the energy demand (48.229kWh) of the private properties if you do not take into account the hourly variations in energy demand and production (DRQ 2.1).

The only data available with regard to energy demand is yearly data. This makes it impossible to investigate the possible mismatch between energy production and demand. Literature however does provide a strong indication. Literature indicates that there is a coincidence between energy production and demand during summer times, during the winter this is not the case however. Based on the literature it can be safely assumed that in a scenario where 10% of the roof surfaces is used for solar energy, some energy will go to waste when there are no proper energy storage measures taken (DRQ 2.2).

6.2 RECOMMEDATIONS

This study presents a way to use 3D geodata in combination with other data to determine the return on investment of solar panels. This study however is not conclusive and there is room for improvement as can also be seen in Chapter 5. For future research in this or a similar domain there are several topics that require further attention.

Firstly the climatological models and its output area not validated for this study, this requires field study to determine the accuracy. For direct irradiance the method used is often applied in other research. For diffuse irradiance however the method can be improved greatly, starting with taking into account the spatial surroundings of the area. There is a possibility here for a GIS solution, where the skyview for a location is analyzed and used to determine the diffuse irradiance for a location.

Secondly there is a problem with validating 3D models. At this moment there is no sensible way to accurately validate the geometry of 3D models. Cadastral data can be used to determine the accuracy of the building footprint, but even when using original building plans there likely is a deviation in the third dimension.

Thirdly this study uses only a small 3D model, further study can be done on how to apply this methodology to larger area, for example whole cities to determine the benefit of 3D geodata over raster data on this scale. In this context another question is how to create whole 3D cities without manually creating the buildings. Comparing the results of analysis using 3D geodata with analysis using rasters is also a topic that demands attention.
Also a further study into the dynamics between (solar) energy production and energy demand have to be further investigated to be able to really determine the effects of localized energy production.

Finally similar methods that are presented here can also be used in different topics, for example a short term prediction model of solar energy production for smart grids based on previous performance of the solar panel and metrological short term predictions for that location.

7 WORKS CITED

PyEmphem Home page. (2009, June). Retrieved March 5, 2011, from PyEmphem Home page: http://rhodesmill.org/pyephem/

3D Virtual City: Shadows over Time | ArcGIS Resource Center. (2011, July 22). Retrieved February 2011, from ArcGIS Resource Center: http://resources.arcgis.com/gallery/file/3dgis/details?entryID=A6CE4866-1422-2418-3411-3EE356CA2F56

CBS StatLine. (2011, August). Retrieved August 08, 2011, from CBS: http://statline.cbs.nl/StatWeb/selection/default.aspx?VW=T&DM=SLNL&PA=71556ned&D1=0&D2=(l-11)-l&HDR=T&STB=G1

Akorede, M. F., Hizam, H., & Pouresmaeil, E. (2009). Distributed energy resources and benefits to the environment. *Renewable and Sustainable Energy Reviews*, 724-734.

Barbason, M., & Reiter, S. (2010). About the choice of a turbulence model in building physics simulations. *Proceedings of the 7th conference on Indoor Air Quality, Ventilation and Energy Conservation in building.* Syracuse: Syracuse University.

Bosch. (2011, June). Bosch Solar Module c-Si M 60. Erfurt, Thuringia, Germany.

Bourke, P. (1987). *Determining if a point lies on the interior of a polygon*. Opgeroepen op 2011 May, van Geometry, Surfaces, Curves, Polyhedra: http://paulbourke.net/geometry/insidepoly/

Bradsher, K. (2010, September 8). *On Clean Energy, China Skirts Rules*. Opgeroepen op November 18, 2010, van New York Times: http://www.nytimes.com/2010/09/09/business/global/09trade.html?pagewanted=2

Chicco, G., & Mancarella, P. (2009). Distributed multi-generation: A comprehensive view. *Renewable and Sustainable Energy Reviews*, 535-551.

Denholm, P., & Margolis, R. (2007). Evaluating the limits of solar photovoltaics (PV) in electric power systems utilizing energy storage and other enabling technologies. *Energy Policy*, 4424-4433.

Denholm, P., & Margolis, R. (2007). Evaluating the limits of solar photovoltaics (PV) in traditional electric power systems. *Energy Policy*, 2852-2861.

Electric Power Research Institute. (2009). *Report to NIST on the Smart Grid Interoperability Standards Roadmap.*

ESRI. (2011, August 2). *3D Virtual City: Shadows over time*. Retrieved March 1, 2011, from 3D Virtual City: Shadows over time: http://resources.arcgis.com/gallery/file/3dgis/details?entryID=A6CE4866-1422-2418-3411-3EE356CA2F56

Guthrie CAD/GIS Software. (2011). *Guthrie CAD :: GIS*. Retrieved februari 1, 2011, from ArcView/ESRI Shapefile to DXF/DWG Converter: http://www.guthcad.com/arcv2cad.htm

Hamilton, H., & Jackson, A. (1985). A shield for obtaining diffuse sky radiation from portions of the sky. *Solar energy 34*, 121-123.

Hammons, T. (2008). Integrating renewable energy sources into European grids. *Electrical Power and Energy Systems*, 462-475.

JA solar. (2011). JAC P6MF 3BB. Shanghai, China.

Kolbe, T. (2005, May 28). *Homepage of CityGML: What is CityGML*. Opgeroepen op August 22, 2011, van Homepage of CityGML: Current News: http://www.citygml.org/index.php?id=1523

Koronakis, P. (1986). On the choice of the angle of tilt for south facing solar collectors in the Athens Basin area. *Solar Energy 36*, 217-225.

Kroon, d. L. (2011, June 20). (P. Ruiter, Interviewer)

Liander & Enexis. (2011). *Energie in beeld*. Retrieved June 2011, from Energie in beeld: http://www.energieinbeeld.nl

Municipality Assen. (n.d.). *Energieloket Assen*. Retrieved July 15, 2011, from Energieloket Assen: http://energieloket.assen.nl/energieloket/jsp/

Noorian, A., Moradi, I., & Kamali, G. (2008). Evaluation of 12 models to estimate hourly diffuse irradiation on inclined surfaces. *Renewable Energy 33*, 1406-1412.

Notton, G., Cristofari, C., & Poggi, P. (2006). Performance evaluation of various hourly shlope irradiation models using Mediterranean experimental data of Ajoccio. *Energy Conversion and Management 47*, 147-173.

Parida, B., Iniyan, S., & Goic, R. (2011). A revies of solar photovoltaic technologies. *Renewable and Sustainable Energy Reviews*, 1625-1636.

Perez, R., Ineichen, P., Seals, R., Michalsky, J., & Stewart, R. (1990). Modeling daylight availability and irradiance components from direct and global irradiance. *Solar Energy 44*, 271-289.

pyshp - Python Shapefile Library. (n.d.). Retrieved May 2011, from pyshp - Python Shapefile Library: http://code.google.com/p/pyshp/

Reiter, S. (2010). Assessing wind comfort in urban planning. *Environment & Planning B : Planning & Design*, 857-873.

Sharp. (2011, June). Sharp Flyer Serrie NU J5. Hamburg, Hamburg, Germany.

Solar Boston. (n.d.). *Solar Boston*. Retrieved November 18, 2010, from City of Boston: http://gis.cityofboston.gov/solarboston/

Solarbuzz. (2011). *Module Pricing | Solarbuzz.* Retrieved August 8, 2011, from Solarbuzz | Solar Market Research and Analysis: http://www.solarbuzz.com/facts-and-figures/retail-price-environment/module-prices

Strzalk, A., Eicke, U., Coors, V., & Schumacher, J. (2010). Modeling energy demand for heating at city scale. *Fourth National Conference of IBPSA-USA*, (pp. 358-364). New York.

Sunday, D. (sd). *Intersection of Lines and Planes*. Opgeroepen op May 2011, van Geometry Algorithm Home: http://softsurfer.com/Archive/algorithm_0104/algorithm_0104B.htm#Line-Plane Intersection

Suntech. (2011). EN-STD-Wd+-NO1.01-Rev 2011.

TNS Opinion & Social. (2009). *Special Eurobaromet 313: Europeans' attitudes towards climate change.*

Van der Vaart, N. (2009). *Powermapping.*

van Sark, W. (2007). Teaching the relation between solar cell efficiency and annual energy yield. *European Journal of Physics*, 415-527.

Wikipedia. (n.d.). *Cross product - Wikipedia*. Retrieved May 2011, from http://en.wikipedia.org/wiki/Cross_product

Wikipedia. (n.d.). *Dot product - Wikipedia*. Retrieved May 2011, from Wikipedia, the free encyclopedia: http://en.wikipedia.org/wiki/Dot_product

Wikipedia. (n.d.). *Surface normal - Wikipedia*. Retrieved May 2011, from Wikipedia, the free encyclopedia: http://en.wikipedia.org/wiki/Surface_normal

Wikipedia. (n.d.). *Unit vector - Wikipedia*. Retrieved May 2011, from Wikipedia, the free encyclopedia: http://en.wikipedia.org/wiki/Unit_vector

APPENDIX 1: MATHEMATICS

Different mathematic functions are used in this research. This appendix summarized them.

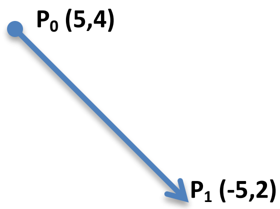
ANGLE OF INCIDENCE

For this research the angle of incidence is calculated as a value between -1 and 1, where >0 the surface is facing the sun and <0 is facing away from the sun. In calculations this ratio is directly applied to the insolation values from the sun.
The formula used for this: $\theta = \sin(S) * \sin(Z) * \cos(\alpha_s - \alpha_t) + \cos(S) * \cos(Z)$ (Noorian, Moradi, & Kamali, 2008), where:

- S: tilt of the surface from the horizontal in radians
- α_T: rotation of the surface from the north-south axis in radians
- θ: solar incidence ratio
- Z: solar zenith angle in radians
- α_T: solar azimuth angle in radians

DOT PRODUCT

The dot product takes two sequences of equal length and returns a single value (Wikipedia). For example numbers representing xy coordinates (5,4) and (10,9). The dot product is then: 5*10+4*9 = 85



NORMAL VECTOR (LINE)

The normal vector of a line is calculated by subtracting a point from the origin. If the origin (P_0) = (5,4) and a point (P_1) on the line is (-5,2) then the normal vector is: P_1-P_0 = (-10,2)

NORMAL VECTOR (POLYGON)

Calculating the normal vector of a convex polygon, also called a surface normal, is calculated with cross product (Wikipedia)of the vector (Wikipedia). In this research all polygons are convex. For each polygon the first three point coordinates are taken to calculate the surface normal.
Example:
P0, P1, P2 are three corner points of a polygon with xyz coordinates. The cross product is then:
$x = (P_{1.y} - P_{0 \cdot y}) * (P_{2.z} - P_{0.z}) - (P_{1.z} - P_{0.z}) * (P_{2.y} - P_{0.y})$
$y = (P_{1.z} - P_{0 \cdot z}) * (P_{2.x} - P_{0.x}) - (P_{1.x} - P_{0.x}) * (P_{2.z} - P_{0 \cdot z})$
$z = (P_{1.x} - P_{0 \cdot x}) * (P_{2.y} - P_{0.y}) - (P_{1.y} - P_{0.y}) * (P_{2 \cdot x} - P_{0 \cdot x})$

UNIT VECTOR

The unit vector (Wikipedia), also called the normalized normal vector, is derived from the normal vector. The ratio between x y (and z) stays the same, only the length of the vector is 1. So if we take the vector (-10,2) (see normal vector (line)), the unit vector is:

$$x = \frac{-10}{\sqrt{-10^2+2^2}} = -0.98 \quad y = \frac{2}{\sqrt{-10^2+2^2}} = 0.20$$

DERIVING ANGLES FROM UNIT VECTOR

unit vector = UV

To derive the direction (xy) in degrees from a unit vector the following formula is used:

$$direction = atan2(UV_y, UV_x) * \frac{180}{\pi}$$

To derive the angle (z) in degrees from a unit vector the following formula is used:

$$angle = atan2\left(\left(\sqrt{UV_x{}^2 * UV_y{}^2}\right), UV_z\right) * \frac{180}{\pi}$$

LINE-PLANE INTERSECTION

The first step of determining if a line intersects with a polygon is to determine the point of intersect with the plane derived from the normal vector of the polygon (Sunday, no date).

Three vectors are used in the calculation:
n: normal vector of the polygon
u: normal vector of the line
w: normal vector between P_0 and V_0

with the vectors two more variables are calculated:
N = dot product between n and u
D = dot product between n and w
If D = 0 then the line and plane are parallel and no intersection point can be calculated. If this is not the case S_I can be calculated: $S_I = \frac{N}{D}$
To calculate the xyz coordinates at $P(S_I)$: $P(S_I) = P_o + (S_I * u)$.

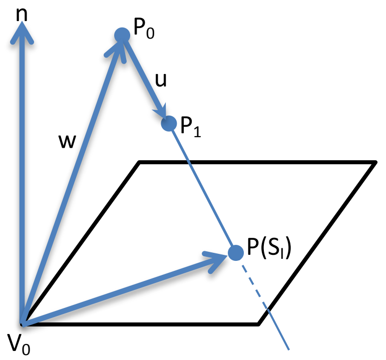


Figure 37: Line-plane intersection

DETERMINE POINT IN POLYGON

Different methods exist to determine if a point lies in a polygon (Bourke, 1987). For this research a solution is chosen based on the easy implantation of the solution.
This method takes the sum of the angles between the point and every pair of edge points. The sum of this will only be 2π if the point lies in the polygon.

Take for example the angle between P_0,P_1,$P(S_I)$
Two variables are first calculated:

$$M_1 = \sqrt{P_{0.x}{}^2 + P_{o.y}{}^2 + P_{o.z}{}^2}$$

$$M_2 = \sqrt{P_{1.x}{}^2 + P_{1.y}{}^2 + P_{1.z}{}^2}$$

$$angle = \cos^{-1}\left(\frac{P_{0.x}*P_{1.x}+P_{0.y}*P_{1.y}+P_{0.z}*P_{1.z}}{M_0*M_1}\right)$$

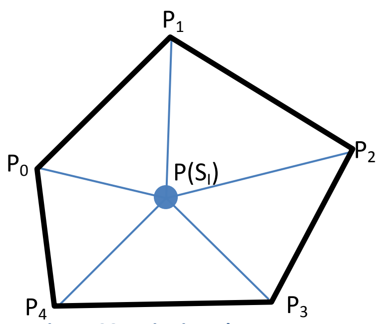


Figure 38: Point in polygon

APPENDIX 2: PYTHON SCRIPTS

Different python scripts are created for this research. This appendix contains all scripts created for this research. The additional python modules and 3rd party scripts used for this research are:
PyEphem (PyEmphem Home page, 2009), pyshp (pyshp - Python Shapefile Library), Create skymap (3D Virtual City: Shadows over Time | ArcGIS Resource Center, 2011)

CALCULATE INSOLATION FOR DIFFERENT POSITIONS (MONTH)

```
import math
from math import radians
from math import cos
from math import sin
from os import path


infile= #csv file with daily irradiance values
outloc= #output location

def runner(infile,outloc):
    sunlist=create_sunlist(infile)
    monthlist=create_monthlist(sunlist)
    print 'lists created'
    for i in range(len(monthlist)):
        print 'start',i+1
        outfile=path.join(outloc,'suntable'+str(i+1)+'.csv')
        calc_write_irr(monthlist[i],outfile)
    return

def create_sunlist(infile):
    sunfile=open(infile,'r')
    sunlist=[]
    for line in sunfile:
        line=line.replace(',','.')
        line=line.strip()
        sunlist+=[line.split(';')]
    sunfile.close()
    del sunlist[0]
    for hour in sunlist:
        hour[0]=int(hour[0])
    return sunlist

def create_monthlist(sunlist):
    monthlist=[]
    month=[]
    M=1
    for hour in sunlist:
        if hour[0] == M:
            month+=[hour]
        else:
            monthlist+=[month]
            month=[]
            M+=1
    monthlist+=[month]
    return monthlist
```

```
#[0]Month(2)
#[1]direct
#[2]diff
#[3]AZIMUTH
#[4]ANGLE (zenith = 90-angle)
def calc_write_irr(monthlist,outfile):
    result=[]
    for tilt in range(0,91,2):
     #   print tilt
        for rotation in range(0,361,5):
            sum_irradiance=0
            for hour in monthlist:
                if hour[2] != '-9999':
                    irr=float(hour[1])
                    diff_irr=float(hour[2])
                    azimuth=float(hour[3])
                    zenith=90-float(hour[4])

                    sum_irradiance+=(1.0/3)*diff_irr*(2+cos(radians(tilt))) #diffuse radiation

                    ratio=sin(radians(tilt))*sin(radians(zenith))*\
                          cos(radians(azimuth-rotation))+\
                          cos(radians(tilt))*cos(radians(zenith))
                    if ratio > 0:
                        sum_irradiance+=irr*ratio
                    else: pass
            result+=[[tilt,rotation,sum_irradiance]]
    _write_res(result,outfile)
    return

def _write_res(result,outfile):
    TF=open(outfile,'w')
    start=0
    TF.write(' ')
    for i in range(0,361,5):
        TF.write(';')
        TF.write(str(i))
    TF.write('\n')
    for i in result:
        if int(i[0])==start:
            TF.write(';')
            irr=str(i[2]).replace('.',',')
            TF.write(irr)
        else:
            start+=2
            TF.write('\n')
            TF.write(str(start))
            TF.write(';')
            irr=str(i[2]).replace('.',',')
            TF.write(irr)
    TF.close()
    print outfile
    return
```

CALCULATE INSOLATION FOR DIFFERENT POSITIONS (YEAR)

```
import math

from math import radians
from math import cos
from math import sin
from os import path


infile= #csv file with daily irradiance values
outloc= #output location

def runner(infile,outfile):
    sunlist=create_sunlist(infile)
    calc_write_irr(sunlist,outfile)
    return

def create_sunlist(infile):
    sunfile=open(infile,'r')
    sunlist=[]
    for line in sunfile:
        line=line.replace(',','.')
        line=line.strip()
        sunlist+=[line.split(';')]
    sunfile.close()
    del sunlist[0]
    for hour in sunlist:
        hour[0]=int(hour[0])
    return sunlist

#[0]Month(2)
#[1]direct
#[2]diff
#[3]AZIMUTH
#[4]ANGLE (zenith = 90-angle)
def calc_write_irr(monthlist,outfile):
    result=[]
    for tilt in range(0,91,1):
    #   print tilt
        for rotation in range(0,361,1):
            sum_irradiance=0
            for hour in monthlist:
                if hour[2] != '-9999':
                    irr=float(hour[1])
                    diff_irr=float(hour[2])
                    azimuth=float(hour[3])
                    zenith=90-float(hour[4])

                    sum_irradiance+=(1.0/3)*diff_irr*(2+cos(radians(tilt))) #diffuse radiation

                    ratio=sin(radians(tilt))*sin(radians(zenith))*\
                          cos(radians(azimuth-rotation))+\
                          cos(radians(tilt))*cos(radians(zenith))
                    if ratio > 0:
                        sum_irradiance+=irr*ratio
                    else: pass
```

```
            result+=[[tilt,rotation,sum_irradiance]]
    _write_res(result,outfile)
    return

def _write_res(result,outfile):
    TF=open(outfile,'w')
    start=0
    TF.write(' ')
    for i in range(0,361,1):
        TF.write(';')
        TF.write(str(i))
    TF.write('\n')
    for i in result:
        if int(i[0])==start:
            TF.write(';')
            irr=str(i[2]).replace('.',',')
            TF.write(irr)
        else:
            start+=1
            TF.write('\n')
            TF.write(str(start))
            TF.write(';')
            irr=str(i[2]).replace('.',',')
            TF.write(irr)
    TF.close()
    print outfile
    return
```

ADD AREA, ANGLE, ROTATION

```
#http://help.arcgis.com/en/arcgisdesktop/10.0/help/index.html#//002z0000001q000000.htm
import arcpy
import math
from math import cos,sin,radians,atan2,sqrt,degrees
import os

arcpy.env.overwriteOutput=True

infile=#input polygon file
outfile=#output location
suncsv=#.csv file with irradiance values

def runner(infile,outfile,suncsv):
   tempout=os.path.join(os.path.dirname(outfile),'tempcalcfile.shp')
   read_poly(infile,tempout)
   select_roofs(tempout,outfile)
   calc_diff_irr(outfile,suncsv)
   return

def read_poly(infile,outfile):
   arcpy.CalculateAreas_stats(infile, outfile)
#first, add fields
   Field_Name = "ANGLE"
   Field_Type = "FLOAT"
   arcpy.AddField_management(outfile, Field_Name, Field_Type, "" , "", "", "", "NON_NULLABLE",
"NON_REQUIRED", "")
   Field_Name = 'ROTATION'
   arcpy.AddField_management(outfile, Field_Name, Field_Type, "" , "", "", "", "NON_NULLABLE",
"NON_REQUIRED", "")
   Field_Name = 'MIN_Z'
   arcpy.AddField_management(outfile, Field_Name, Field_Type, "" , "", "", "", "NON_NULLABLE",
"NON_REQUIRED", "")
   Field_Name = 'MAX_Z'
   arcpy.AddField_management(outfile, Field_Name, Field_Type, "" , "", "", "", "NON_NULLABLE",
"NON_REQUIRED", "")

#secondly, iterate through rows
   rows=arcpy.UpdateCursor(outfile)
   for row in rows:
      pointdata=[]
      for part in row.shape:
         pnt = part.next()
         while pnt:
            pointdata+=[(pnt.X, pnt.Y, pnt.Z)]
            pnt = part.next()
            if not pnt:
               pnt = part.next()
               if pnt:
                  interiorRing = True
#if points are not NULL, calc normal vector and update table
      if len(pointdata) != 0:
         A,R,lZ,hZ=_get_ang_rot(pointdata)
         row.setValue('ANGLE',A)
         row.setValue('ROTATION',R)
         row.setValue('MIN_Z',lZ)
```

```
        row.setValue('MAX_Z',hZ)
        rows.updateRow(row)

    Coordinate_System                                                                       =
"PROJCS['RD_New',GEOGCS['GCS_Amersfoort',DATUM['D_Amersfoort',SPHEROID['Bessel_1841',6377397.155,
299.1528128]],PRIMEM['Greenwich',0.0],UNIT['Degree',0.0174532925199433]],PROJECTION['Double_Stereog
raphic'],PARAMETER['False_Easting',155000.0],PARAMETER['False_Northing',463000.0],PARAMETER['Central_
Meridian',5.38763888888889],PARAMETER['Scale_Factor',0.9999079],PARAMETER['Latitude_Of_Origin',52.15
616055555555],UNIT['Meter',1.0]]"
    arcpy.DefineProjection_management(outfile, Coordinate_System)
    del row,rows,outfile
    return

def _get_ang_rot(data):
    x1=data[0][0];    x2=data[1][0];    x3=data[2][0]
    y1=data[0][1];    y2=data[1][1];    y3=data[2][1]
    z1=data[0][2];    z2=data[1][2];    z3=data[2][2]

    Nx=(y2-y1)*(z3-z1)-(z2-z1)*(y3-y1)
    Ny=(z2-z1)*(x3-x1)-(x2-x1)*(z3-z1)
    Nz=(x2-x1)*(y3-y1)-(y2-y1)*(x3-x1)
    tmp=sqrt(Nx*Nx+Ny*Ny+Nz*Nz)

    normX=Nx/tmp;    normY=Ny/tmp;    normZ=Nz/tmp
    o=degrees(atan2(sqrt(normX*normX+normY*normY),normZ))
    q=degrees(atan2(normY,normX))#rotation x/y
    if o > 90:
        o-=90
    A=int(o*10)/10.0
    R=int(q*10)/10.0
    lZ=int((min(z1,z2,z3))*10)/10.0
    hZ=int((max(z1,z2,z3))*10)/10.0
    return A,R,lZ,hZ

def select_roofs(infile,outfile):
    #angle is still inverted (flat==90)
    expression="\"ANGLE\" >= 15 AND \"MIN_Z\" >= 6 AND \"F_AREA\" >= 2"
    arcpy.Select_analysis(infile,outfile,expression)
    return

def calc_diff_irr(infile,suncsv):
    #read sun values
    sunfile=open(suncsv,'r')
    sunlist=[]
    for line in sunfile:
        line=line.replace(',','.')
        line=line.strip()
        sunlist+=[line.split(';')]
    sunfile.close()
    #create field
    arcpy.AddField_management(infile,    'IRRADIANCE',    'FLOAT',    ""    ,    "",    "",    "",    "NON_NULLABLE",
"NON_REQUIRED", "")
    arcpy.AddField_management(infile, 'SIL_FID', 'DOUBLE', "" , "", "", "", "NON_NULLABLE", "NON_REQUIRED",
"")
    del sunlist[0]
    #per roof, calc yearly diffuse irradiance
    # diffuse formula:(1.0/3)*float(day[2])*(2+math.cos(math.radians(float(day[4]))))
```

```
rows=arcpy.UpdateCursor(infile)
for row in rows:
    roof_angle=row.getValue('angle')
    roof_angle=90-roof_angle
    if roof_angle>0:
        roof_rot=row.getValue('rotation')
        roof_rot=360-(roof_rot+180)+90
        if roof_rot>360:
            roof_rot -=360
        row.setValue('rotation',roof_rot)
    irr=0
    for hour in sunlist:
       # print hour
        if hour[4] != '-9999':
            irr+=(1.0/3)*float(hour[2])*(2+cos(radians(float(roof_angle))))
        else:pass
    row.setValue('angle',roof_angle)
    row.setValue('IRRADIANCE',irr)
    row.setValue('SIL_FID',(row.getValue('FID')))
    rows.updateRow(row)
del row
del rows
del infile
return
```

CREATE ROOF POINTS

```
import arcpy,shapefile,os
arcpy.CheckOutExtension("3D")
arcpy.env.overwriteOutput=True

#bounding box of the area and the interval
minx=233954;maxx=234088
miny=581950;maxy=582083
interval=0.5
outfile = # output pointfile. The
roof3 D= # input multipatch file on which the points have to be created

def create_points(minx,miny,maxx,maxy,interval,outfile,roof3D):
    listx=[]
    listy=[]
    x=minx
    y=miny
    points=[]
    while x < maxx:
        listx+=[x]
        x+=interval
    while y < maxy:
        listy+=[y]
        y+=interval
    listx+=[x]
    listy+=[y]
    for x in listx:
        for y in listy:
            points+=[(x,y)]

    outline=os.path.join(os.path.dirname(outfile),'temp_line.shp')
    _write_lineZ(points,outline)
    arcpy.Intersect3DLineWithMultiPatch_3d(outline, roof3D, "IDS_ONLY", outfile, "")
    return

def _write_lineZ(points,outfile):
    w = shapefile.Writer(shapeType=13)
    w.field('ID','N')
    count=0
    for point in points:
        point0=(point[0],point[1],0)
        point1=(point[0],point[1],50)
        part=[(point0,point1)]
        w.line(parts=part,shapeType=15)
        w.record(count)
        count+=1
    w.save(outfile)

    Coordinate_System =
"PROJCS['RD_New',GEOGCS['GCS_Amersfoort',DATUM['D_Amersfoort',SPHEROID['Bessel_1841',6377397.155,
299.1528128]],PRIMEM['Greenwich',0.0],UNIT['Degree',0.0174532925199433]],PROJECTION['Double_Stereog
raphic'],PARAMETER['False_Easting',155000.0],PARAMETER['False_Northing',463000.0],PARAMETER['Central_
Meridian',5.38763888888889],PARAMETER['Scale_Factor',0.9999079],PARAMETER['Latitude_Of_Origin',52.15
616055555555],UNIT['Meter',1.0]]"
    arcpy.DefineProjection_management(outfile, Coordinate_System)
    return
```

SIMULATE SHADOW

```
import arcpy, time, os,shapefile
from math import sqrt,sin,cos,radians,acos

arcpy.CheckOutExtension("3D")
arcpy.env.overwriteOutput=True


sunpoints= #input positions of the sun(point shapefile)
roofpoints= #input points on the roof (point shapefile)
workmap= #location for the output to be written to
roofs= #input roof areas (multipatch shapefile)
s_area= #input whole area representing the buildings (polygon shapefile)

midpoint=[234020,582020,0] #xyz approx mid study area


def runner(roofpoints,roofs,sunpoints,s_area,midpoint):
    #create workmap if not exists
    if not os.path.exists(workmap):
        os.makedirs(workmap)
    roofdata_poly = get_roofdata(roofs) #returns a dictionary; key=fid, data=angle/rotation
    RP_data = get_roofpoints(roofpoints,roofdata_poly) #combines point with roofdata: 0=FID,1=coor,2=A,R,3=counters
    sunlist = get_sundata(sunpoints,midpoint)  #OBSpoint,AZI,ZEN,IRR
    area_geom = get_poly_geom(s_area)#gets the polygons of the whole area
    print len(RP_data)
    print len(area_geom)
    for sun in sunlist:
        if sun[1][2] > 0:
            print time.asctime(),'\t','start',sun[2][0]
            #set global variables for later use
            global sun_zen;     global sun_azi;     global sun_irr
            sun_zen=sun[1][1];     sun_azi=sun[1][0];     sun_irr=sun[1][2]
            global sun_Vx;     global sun_Vy;     global sun_Vz
            sun_Vx=sun[0][0];     sun_Vy=sun[0][1];     sun_Vz=sun[0][2]

            RP_data=calc_incidence(RP_data)
            inshadow=intersect(RP_data,area_geom)
            outfile=os.path.join(workmap,sun[2][0]+'.shp')

                #for research purposes each iteration result is written to a file
            write_points(RP_data,outfile,inshadow)
            print time.asctime(),'\t',outfile,'written'
        else:
            continue
    outfile=os.path.join(workmap,'final.shp')
    write_points_final(RP_data,outfile)
    return

def write_points(RP_data,outfile,inshadow):
    w = shapefile.Writer(shapeType=11)
    w.field('ID','N')
    w.field('INSUN','N')
    w.field('IRR','N')
    w.field('RATIO','C')
```

```
    for i in RP_data.keys():
        if i in inshadow:
            RP_data[i][2][2]+=1
            P=RP_data[i]
            w.point(P[0][0],P[0][1],P[0][2])
            w.record(i,0,int(P[2][0]),P[2][1])
        else:
            RP_data[i][2][0]+=sun_irr*RP_data[i][2][1]
            P=RP_data[i]
            w.point(P[0][0],P[0][1],P[0][2])
            w.record(i,1,int(P[2][0]),P[2][1])
    w.save(outfile)

def write_points_final(RP_data,outfile):
    w = shapefile.Writer(shapeType=11)
    w.field('ID','N')
    w.field('COUNTSHAD','N')
    w.field('kJ_cm2','N')
    w.field('kWh_cm2','N')
    w.field('IRR','N')
    for i in RP_data.keys():
        if RP_data[i][2][2] < 150:
            tilt=RP_data[i][1][0]
            diff=101526
            RP_data[i][2][0]+=(1.0/3)*diff*(2+cos(radians(tilt)))
            P=RP_data[i]
            kJ=int(round((P[2][0]/1000),0))
            kWh=int(round((P[2][0]*pow(2.7778,-7)),0))
            w.point(P[0][0],P[0][1],P[0][2])
            w.record(i,P[2][2],kJ,kWh,int(P[2][0]))
    w.save(outfile)

def get_roofdata(roofs):
    data={}
    rows = arcpy.SearchCursor(roofs)
    for row in rows:
        FID = row.getValue('FID')
        A = row.getValue('ANGLE')
        R = row.getValue('ROTATION')
        if A < 5:
            A = 34
            R = 167
        data[FID]=[A,R]
    del row,rows
    return data

def get_roofpoints(roofpoints,roofdata):
    IRR=0.0;    COUNTSHAD=0.0;    RATIO=0.0
    RP_DATA={}; data=[]

    desc = arcpy.Describe(roofpoints)
    shapefieldname = desc.ShapeFieldName
    rows = arcpy.SearchCursor(roofpoints)
    for row in rows:

        feat = row.getValue(shapefieldname)
        pnt = feat.getPart()
```

```
        point_coor=[pnt.X,pnt.Y,pnt.Z]
        FID = row.getValue('FID')
        MP_ID = row.getValue('MPATCH_OID')
        AR=roofdata[MP_ID]
        data=[point_coor,AR,[IRR,RATIO,COUNTSHAD]]
        RP_DATA[FID]=data
    del row,rows
    return RP_DATA

def calc_incidence(points):
    for point in points.keys():
        data=points[point]
        A=data[1][0]
        R=data[1][1]
        ratio=sin(radians(A))*sin(radians(sun_zen))*\
              cos(radians(sun_azi-R))+\
              cos(radians(A))*cos(radians(sun_zen))
        data[2][1] = ratio
    return points

def get_poly_geom(poly):
    roofs_geom=[]
    desc = arcpy.Describe(poly)
    shapefieldname = desc.ShapeFieldName
    rows = arcpy.SearchCursor(poly)
    for row in rows:
        roof=[]
        feat = row.getValue(shapefieldname)
        partnum = 0
        partcount = feat.partCount
        while partnum < partcount:
            part = feat.getPart(partnum)
            pnt = part.next()
            pntcount = 0
            while pnt:
                if pntcount==0:
                    startpoint=[[pnt.X,pnt.Y,pnt.Z]]
                    Xmin=pnt.X; Ymin=pnt.Y; Zmin=pnt.Z
                    Xmax=pnt.X; Ymax=pnt.Y; Zmax=pnt.Z
                else:
                    Xmin=min(Xmin,pnt.X); Ymin=min(pnt.Y,Ymin); Zmin=min(pnt.Z,Zmin)
                    Xmax=max(Xmax,pnt.X); Ymax=max(pnt.Y,Ymax); Zmax=max(pnt.Z,Zmax)
                    currentpoint=[[pnt.X,pnt.Y,pnt.Z]]
                    if currentpoint==startpoint:
                        partnum=partcount-1
                    else: pass
                roof+=[[pnt.X,pnt.Y,pnt.Z]]
                pnt = part.next()
                pntcount+=1
            partnum += 1
        if len(roof) >= 3:
            BBOX=[Xmin,Xmax,Ymin,Ymax,Zmin,Zmax]
            roofs_geom+=[[BBOX,roof]]
    del row,rows
    return roofs_geom
```

```
def get_point_geom(points):
   point_geom=[]
   desc = arcpy.Describe(points)
   shapefieldname = desc.ShapeFieldName
   rows = arcpy.SearchCursor(points)
   for row in rows:
      feat = row.getValue(shapefieldname)
      pnt = feat.getPart()
      point=[pnt.X,pnt.Y,pnt.Z]
      point_geom+=[point]
   return point_geom

def get_sundata(sunpoints,midpoint):
   sun_data=[]
   desc = arcpy.Describe(sunpoints)
   shapefieldname = desc.ShapeFieldName
   rows=arcpy.SearchCursor(sunpoints)
   for row in rows:
   #get geom
      feat = row.getValue(shapefieldname)
      pnt = feat.getPart()
      point=[pnt.X,pnt.Y,pnt.Z]
   #calc normalized vector
      V=line_norm(midpoint,point)
      Vdist=sqrt(V[0]*V[0]+V[1]*V[1]+V[2]*V[2])
      V[0]=V[0]/Vdist;    V[1]=V[1]/Vdist;    V[2]=V[2]/Vdist;

   #calc vector
   #get data
      FID=int(row.getValue('FID'))
      AZI=row.getValue('AZIMUTH')
      ZEN=90-row.getValue('ANGLE')
      DIR_IRR=row.getValue('DIRR_IRR')
      DATE=row.getValue('time_sec')

      NEWDATE=transform_date(DATE)
      sun_data+=[[V,[AZI,ZEN,DIR_IRR],[NEWDATE]]]
   del row,rows,sunpoints
   return sun_data

def transform_date(date):
   newdate=''
   if date[1] == '/':
      date='0'+date
   if date[4] == '/':
      date=date[:3]+'0'+date[3:]
   date=date.replace(' ','')
   date=date.replace('/','')
   date=date.replace(':00:00','')

   newdate=date[0:4]+'_'+date[8:len(date)]
   return newdate

def intersect(RP_data,roof_geom):    #RP_data:0=FID,1=coor,2=A,R,3=counters
   X=True;Y=True;Z=True
   if sun_Vx <0: X=False
   if sun_Vy <0: Y=False
```

```
    if sun_Vz <0: Z=False
    inshadow=[]
    int_point=[]
    for i in RP_data:
        point=RP_data[i]
        if point[2][1] < 0:   #angle of incidence
            inshadow+=[i]
            continue
        else:
            for roof in roof_geom:
                BBOX=roof[0]    # 0:Xmin 1:Xmax 2:Ymin 3:Ymax 4:Zmin 5:Zmax
                geom=roof[1]
                #test if the roof in the right direction based on the vector and bounding box
                if (X and (point[0][0] <= BBOX[1])) or (not X and (point[0][0] >= BBOX[0])):pass
                else:continue
                if (Y and (point[0][1] <= BBOX[3])) or (not Y and (point[0][1] >= BBOX[2])):pass
                else:continue
                if (Z and (point[0][2] <= BBOX[5])) or (not Z and (point[0][2] >= BBOX[4])):pass
                else:continue

                is_int,int_P=intersect_plane(point[0],geom)
                if is_int:
                    #test if int point found falls inside the bounding box
                    if (int_P[0] >= BBOX[0]) and (int_P[0] <= BBOX[1]) and\
                        (int_P[1] >= BBOX[2]) and (int_P[1] <= BBOX[3]) and\
                        (int_P[2] >= BBOX[4]) and (int_P[2] <= BBOX[5]):
                        # if point falls in bounding box, calculate if point is inside polygon
                        if determine_inside(int_P,geom):
                            inshadow+=[i]
                            break
    return set(inshadow)

def intersect_plane(point,poly):#0=FID,1=coor,2=A,R,3=counters
    is_int=False
    if sun_Vx>0:X=1
    elif sun_Vx<0:X=-1
    else: X=0
    if sun_Vy>0:Y=1
    elif sun_Vy<0:Y=-1
    else: Y=0
    if sun_Vz>0:Z=1
    elif sun_Vz<0:Z=-1
    else: Z=0
    I=0
    MIN=9999
    if sun_Vx != 0: MIN=abs(sun_Vx)
    if abs(sun_Vy) < MIN and sun_Vy != 0: MIN=abs(sun_Vy)
    if abs(sun_Vz) < MIN and sun_Vz != 0: MIN=abs(sun_Vz)
    L=200/MIN

    P0=(point[0],point[1],point[2])
    P1=(P0[0]+(sun_Vx*L),P0[1]+(sun_Vy*L),P0[2]+(sun_Vz*L))
    u=line_norm(P1,P0)
    w=line_norm(P0,poly[0])
    n=poly_norm(poly)
    D=dot(n,u)
    N=dot(n,w)
```

```
    if abs(D) < 0.000001: # in this case the line is paralel to the surface
        return is_int,I
    else:
        sI = N/D
        I=[P0[0]+(sI*u[0]),P0[1]+(sI*u[1]),P0[2]+(sI*u[2])]
        difX=I[0]-P0[0];        difY=I[1]-P0[1];        difZ=I[2]-P0[2]
        if ((X == 1) and (difX >0)) or ((X == 0) and (difX == 0)) or ((X == -1) and (difX <0)) and\
          ((Y == 1) and (difY >0)) or ((Y == 0) and (difZ == 0)) or ((Y == -1) and (difY <0)) and\
          ((Z == 1) and (difZ >0)) or ((Z == 0) and (difY == 0)) or ((Z == -1) and (difZ <0)):
            is_int=True
    return is_int,I

def determine_inside(P,poly):
    PI='6.283185'
    inside = False
    anglesum=0
# test if the last coordinate equals first, for calculation purposes
    if poly[0] != poly[-1]:
        poly.append(poly[0])
    for i in range(len(poly)-1):
        p1x = poly[i][0] - P[0]
        p1y = poly[i][1] - P[1]
        p1z = poly[i][2] - P[2]
        p2x = poly[i+1][0] - P[0]
        p2y = poly[i+1][1] - P[1]
        p2z = poly[i+1][2] - P[2]

        p1=(p1x,p1y,p1z)
        p2=(p2x,p2y,p2z)
        m1=sqrt(p1[0]*p1[0] + p1[1]*p1[1] + p1[2]*p1[2])
        m2=sqrt(p2[0]*p2[0] + p2[1]*p2[1] + p2[2]*p2[2])
        if m1*m2 == 0: # in this case the point lies exactly on a polygon point
            inside=True
            return inside
        else:
            try:
                anglesum+= acos((p1x*p2x + p1y*p2y + p1z*p2z) / (m1*m2))
            except ValueError:
                return inside
    if str(anglesum)[:8] == PI:
        inside = True
    else: pass
    return inside

#=====================================================================

def dot(V1,V2):#U,V must be similar length
    result=float(0)
    for i in range(len(V1)):
        result+=(float(V1[i])*float(V2[i]))
    return result

def line_norm(P0,P1):
    Nx=P1[0]-P0[0]
    Ny=P1[1]-P0[1]
    Nz=P1[2]-P0[2]
 #   dist=math.sqrt((Nx*Nx)+(Ny*Ny)+(Nz*Nz))
```

```
    return [Nx,Ny,Nz]

def poly_norm(PG):
    P0=PG[0]
    P1=PG[1]
    P2=PG[2]
    Nx=(P1[1]-P0[1])*(P2[2]-P0[2])-(P1[2]-P0[2])*(P2[1]-P0[1])
    Ny=(P1[2]-P0[2])*(P2[0]-P0[0])-(P1[0]-P0[0])*(P2[2]-P0[2])
    Nz=(P1[0]-P0[0])*(P2[1]-P0[1])-(P1[1]-P0[1])*(P2[0]-P0[0])
    return [Nx,Ny,Nz]
```