NTNU – Trondheim
Norwegian University of
Science and Technology

Future Solar Photovoltaic and Electro Mobility Scenarios for Garmisch-Partenkirchen

Cecilie Sundsbø Arnemo

Abstract

Ever since the German Renewable Energy Act became effective in 2000, it has been a steady increase of renewable energy based electricity production in Germany. The largest share of this growth originates from wind and solar energy, where the solar energy to a large extent is utilized through small-scale distributed generators connected to the low voltage networks. This has altered the hierarchic structure of conventional electrical power systems, where power is generated at large sites, transmitted over long distances through high voltage grids, step-wise transformed down to lower voltages and reaching end-users at the lowest voltage level. Instead, the generators are connected directly to the distribution grid and surplus power from these can end up being transmitted upstream in the grid.

Besides the forthcoming additional installations of distributed generators, an increase in the number of electric vehicles in Germany is expected. The charging of all these could lead to significantly larger loading for the power systems.

If these two foreseen developments in the German electricity sector happen, it could introduce some challenging effects for the electricity network. Thus, it can be wise to analyze if, how or where the impact might occur.

The aim of this master's thesis is to statistically evaluate the impact of future solar photovoltaic and electro mobility scenarios on the grid of Garmisch-Partenkirchen. The town is located in the very south of Germany and has an electricity supply covered 45% by photovoltaics. In addition, it is a model community for electro mobility.

A futuristic Monte Carlo simulation model was developed. Based on a database containing information regarding all buildings in Garmisch- Partenkirchen, the simulation model installs the expected increase in photovoltaic capacity and electric vehicles for the year of 2030 randomly among the buildings in the distribution network. Subsequently, the resulting network situations were examined by a steady state power flow simulation program constructed in MatLab and the distribution system simulator OpenDSS.

The conclusions reached are that the projected photovoltaic capacity for 2030 in Garmisch-Partenkirchen most probably can be integrated without major violations of technical requirements. However, it is observed from the simulation results that network situations with several generators aggregated in close vicinity to each other, cause voltages that violate the 3%-limit set by the German VDE directive. In addition, lines are overloaded with respect to their thermal limits.

Further, the results for electro mobility show that the extra loading from charging vehicles causes large under voltages. The largest voltage drop is observed to be 27.5% below nominal voltage. In addition, congestion of line segments and transformer overloading are observed. Some simulations result in a total of 2.5% overloaded lines and 36% overloaded transformers. Thus, it can be concluded that the existing network in Garmisch-Partenkirchen could encounter difficulties handling large amount of charging vehicles, especially if fast-charging is utilized.

Sammendrag

Helt siden Tysklands *Erneuerbare-Energien-Gesetz* ("Loven om Fornybar Energi") trådte i kraft i 2000, har det vært en jevn økning av kraftproduksjon fra fornybar energi i Tyskland. Den største andelen av veksten stammer fra vind -og solenergi, der solenergi i stor grad utnyttes gjennom småskala genereratorer tilknyttet lavspent fordelingsnett. Dette har endret den hierarkiske strukturen i konvensjonelle kraftsystemer, hvor elektrisk kraft produseres på store anlegg, overføres over lange avstander gjennom høyspentnett, blir trinnvis transformert ned til lavere spenninger og ender hos sluttbrukere på det laveste spenningsnivået. I stedet er kraftproduksjonen koblet direkte til distribusjonsnettet og overskuddskraft fra disse kan ende opp med å bli overført oppstrøms i nettet.

Foruten flere installasjoner av distribuerte generatorer, er det forventet en økning i antall elbiler i Tyskland. Lading av disse kan føre til betydelig økt belastning for kraftsystemet.

Hvis disse to antatte endringene i tysk kraftsektor slår til, vil det kunne føre med seg noen negative effekter for elektrisitetsnettet. Av denne grunn kan det være lurt å analysere hvorvidt, hvordan eller hvor disse effektene vil oppstå.

Målet for masteroppgaven er å statistisk evaluere fremtidige scenarier av solenergi og elektromobilitet og deres påvirkning på distribusjonsnettet i Garmisch-Partenkirchen.

For å lykkes med dette, er en Monte Carlo simuleringsmodell utviklet. Modellens inngangsverdier er, blant annet, en database bestående av alle bygninger i Garmisch-Partenkirchen og informasjon om disse. Modellen plasserer den forventede økningen i hhv. solenergikapasitet og antall elektriske biler for året 2030 tilfeldig blant bygningene i distribusjonsnettet. Dette gjøres via repeterte simuleringer til modellen har konstruert 1000 individuelle fordelinger av solcelleanlegg og elbiler. De forskjellige fordelingenes effekt på nettet blir deretter studert ved hjelp av stasjonær effektflytanalyse utført av et simuleringsprogram konstruert i Matlab og distribusjonsnettsimulatoren OpenDSS. Gjennom effektflytanalysene er det observert hvor den antatte økningen mest sannsynligvis vil ha størst påvirkning.

Konklusjonen er at forventet solenergikapasitet for 2030 i Garmisch- Partenkirchen trolig kan integreres i ekstisterende nett uten, eller kun i beskjeden grad, å komme i konflikt med eltekniske krav. Imidlertid er det observert at geografiske fordelinger der flere solcelleanlegg er plassert i nærheten av hverandre, kan forårsake overspenninger og termisk overbelastning av kabler.

Analysene rundt elektromobilitet viser at den ekstra belastningen fra elbillading forårsaker kraftige spenningsfall i lavspentnettet. Enkelte simuleringer viste spenningsfall på 27.5%. I tillegg viser resultatene tilfeller av opp mot 2.5% termisk overbelastede linjer og opp til 36% overbelastede transformatorer. Det kan dermed se ut til at det eksisterende lavspentnettet i Garmisch-Partenkirchen vil få problemer med å håndtere den ekstra lasten fra lading av elbiler, spesielt hvis hurtiglading benyttes.

Preface

My five year master's degree in electrical power engineering at The Norwegian University of Science and Technology (NTNU) has come to an end. My second last year was spent abroad and this was the beginning of an adventure that led the way to the thesis in front of you. It was my great interest in renewable energy that made me go for exchange to the green pioneer above them all: Germany. I listened with affection to the lectures on the ongoing Energiewende at The Technical University in Munich. While being abroad, I found a highly interesting internship at General Electric Global Research (GE Global Research). There, I got to work with some of the challenges that has come along with the shift in the German energy sector. I studied mitigation of voltage issues caused by distributed generation on the grid of Garmisch-Partenkirchen. I ended up writing my NTNU specialization project on the subject. All this led to the organization of my master's thesis being performed in cooperation with GE Global Research. I am very happy that this arrangement was possible, and that I got the opportunity to learn more about the German energy sector and write my master's thesis on this interesting subject.

I would like to thank my supervisors at GE Global Research, Eva-Maria Bärthlein and Marianne Hartung, for continuous guidance during the whole master's thesis process. The biweekly teleconferences that you arranged, in addition to close contact via email, made the distance between Trondheim and Munich unproblematic. I especially appreciate that my questions, small or large, always were welcomed and met with motivating answers. For me it has been a great pleasure and highly rewarding to work with you. I am also grateful that you put in place my trip to Munich and organized a stay both at the Research Center for Energy Economics (FfE) and at GE Global Research this winter.

I thank my supervisor at NTNU, Trond Toftevaag, for your support in all parts of the master's thesis process. You always had time for my questions and I appreciate that you showed such a great interest in the project and always wanted to participate in the teleconferences with GE in Munich. Thanks also for your performance that amused me and the rest of my fellow students at the graduation ceremony!

I would also like to thank the Research Center for Energy Economics (FfE) in Munich for sharing helpful information and data, in addition to hosting an internship the last week of February this winter. It was very interesting to take part in the network digitalization process and meet other students and researchers who worked on the same project as me. Discussions and sharing of knowledge regarding ongoing research in Garmisch-Partenkirchen were very engaging.

Gratitude also to my family and friends for all your encouragement. The last tributes go to Lars, who patiently has been by my side, always supporting me to go through with my dreams, and Isa, who always makes me look at the bright side of life.

Cecilie Sundsbø Arnemo. Trondheim: June 12th 2014

Contents

List of Figures

List of Tables

Part I

Introduction

Motivation

In autumn 2013, the United Nations Intergovernmental Panel on Climate Change (IPCC) published its fifth assessment report "Climate Change 2013: The Physical Science Basis". It provides the most comprehensive assessment of scientific knowledge on climate change. Among other things, it shows that the atmospheric concentrations of the greenhouse gases carbon dioxide (CO_2), methane (CH_4), and nitrous oxide (N_2O) have increased by 40%, 150% and 20% respectively, since 1750 due to human activity. Further it concludes with 95-100 % certainty that human influence has been the dominant cause of the observed warming since the mid-20th century [1].

Scientific reports like this help arise the general environmental awareness and hopefully an increasing number of countries across the globe will shift towards greener economies and more renewable based energy consumption. A green pioneer in this context is Germany, which for the time being is going through a major transition in their energy sector. The German Renewable Energy Act contains principles that aim to help Germany reach an electricity supply covered 50% by renewable energies by 2030. The country is focusing its efforts on photovoltaics and on-/offshore wind energy [2].

One of the principles that has increased the renewable energy based electricity production in Germany is that all electricity produced from renewable energy sources are guaranteed a fixed feed-in tarif for 20 years [4]. This has led to an increase in small-scale electricity producers, such as households with rooftop photovoltaics. These are situated in a network, which was traditionally not designed for the connection of generating units. Originally, a house in the distribution grid is considered a load which consumes electricity. If this house installs rooftop photovoltaics and produce power that exceeds the consumption in the house, electricity is fed into the grid and the power flow changes direction. This might influence the network in an undesirable way.

In addition to the ongoing installations of distributed generators in Germany, the federal government has stated that in cooperation with the German industry they intend to make Germany the leading international market for electro mobility [3]. The national goal is 1 million electric vehicles in Germany by 2020 and 6 million electric vehicles by 2030 [26], all powered from sustainable energy sources [3]. Such an amount of electric vehicles will cause higher loading in the electricity network and the consequences arising from this will be studied in detail here.

The city Garmisch-Partenkirchen experienced a rapid increase in rooftop photovoltaic installations after the introduction of the Renewable Energy Act. In addition, a further increase in distributed generation and electro mobility is expected in the city. This makes Garmisch-Partenkirchen suitable for investigating whether the projected increase in small-scale photovoltaics and/or electric vehicles will lead to any technical issues for the distribution network operators.

Scope of Work

The overall goal of the master's thesis is to develop methods for statistical evaluation of the impact of future solar photovoltaic (PV) and electro mobility scenarios on the grid of Garmisch-Partenkirchen. The overall goal has been divided into the following work tasks:

- Photovoltaics

  - Based on the findings in [23], assess the effect of different future PV scenarios on the grid of Garmisch-Partenkirchen
  - Develop and perform Monte Carlo simulations distributing the expected additional PV installations locally and in size in the grid of Garmisch-Partenkirchen
  - Statistically examine the results and the effect on the grid, i.e.: with regard to voltage issues, transformer or line overloading etc.

- Electro Mobility

  - Based on several assumptions, define future scenarios for expected electro mobility development in Garmisch-Partenkirchen
  - Develop and perform Monte Carlo simulations distributing the expected additional electric vehicles locally and in size in the grid
  - Statistically examine the results and the effect on the grid, i.e.: with regard to voltage issues, transformer or line overloading, etc.

Focus of the work will be put on the development of the methods for the Monte Carlo simulations and the statistical evaluation of the results. The work will be done using MatLab and OpenDSS.

Outline

Part II presents the theoretical foundations relevant for the research in this thesis. First, the German Renewable Energy Act and some of its effects are elaborated, followed by a chapter explaining voltage characteristics. Last, relevant statistical terms and methods applied later in the thesis are introduced.

Part III covers all analysis performed in this thesis. Chapter 4 gives an overview of the city Garmisch-Partenkirchen. Chapter 5 presents the network model, the simulation program and the load modelling. In chapter 6, all analysis and simulations on photovoltaics are collected. Here, the photovoltaic forecast of Garmisch-Partenkirchen is introduced, as well as the development of the Monte Carlo simulation model. In addition, the results of all photovoltaic simulations are presented in this chapter. Electro mobility analysis and studies are collected in chapter 7. Here, future electro mobility scenarios are defined, before they are modelled and power flow simulations are performed. The chapter ends with a presentation of the results regarding electro mobility. Last, chapter 8 discusses the results, the challenges and the solutions regarding the findings.

Part II

Theory

Chapter 1

The German Renewable Energy Act and Its Effects

Currently, there is a major shift in Germany's energy sector. This is mainly due to The German Renewable Energy Act, called *EEG* (short for Erneuerbare-Energien-Gesetz) or *Energiwende* that came into force April 1st 2000. It legislates that electricity produced from renewable energy sources is guaranteed a fixed feed-in tariff for 20 years[4]. In addition, the distribution network operators are obliged to feed in the power produced by renewables before they feed in electricity from conventional sources. These principles have resulted in a widespread installation of renewables and the total electricity consumption in Germany is now 22% covered by electricity from renewable energy sources [5]. The share among the different renewables in these 22% is displayed in figure 1.1. As seen, a large part originates from intermittent power sources like wind and solar energy.

Source	MWh/year
Wind	59 039 301
Bio	36 419 637
Photovoltaics	32 376 442
Hydro	6 281 034
Sewage Gas	2 224 282
Geothermal	123 697
Total Renewables	**136 464 395**
Total All Sources	**608 050 600**
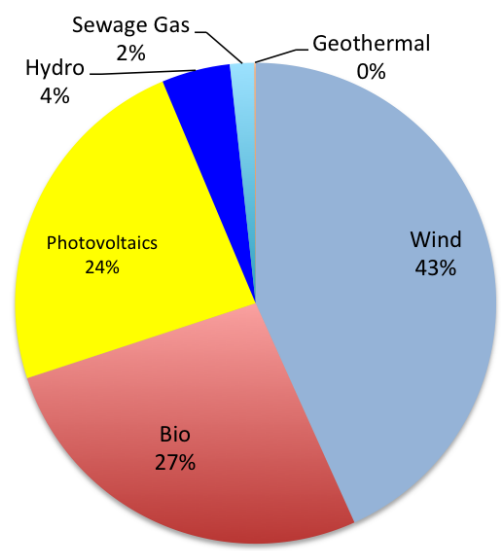
Figure 1.1: Installed Renewables in Germany. Numbers from [5]
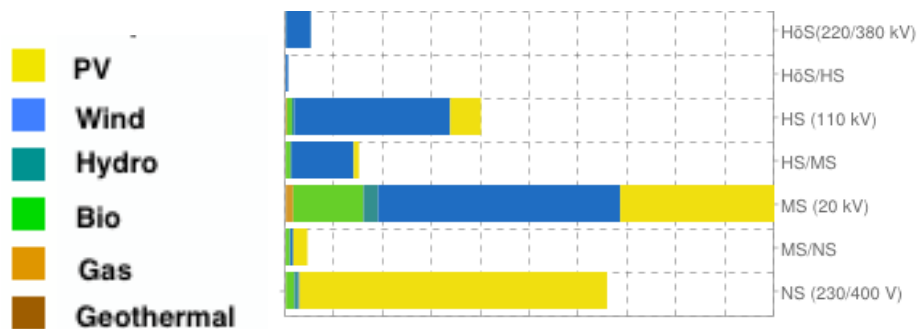
Figure 1.2: Electricity production from renewables divided by voltage level [5].
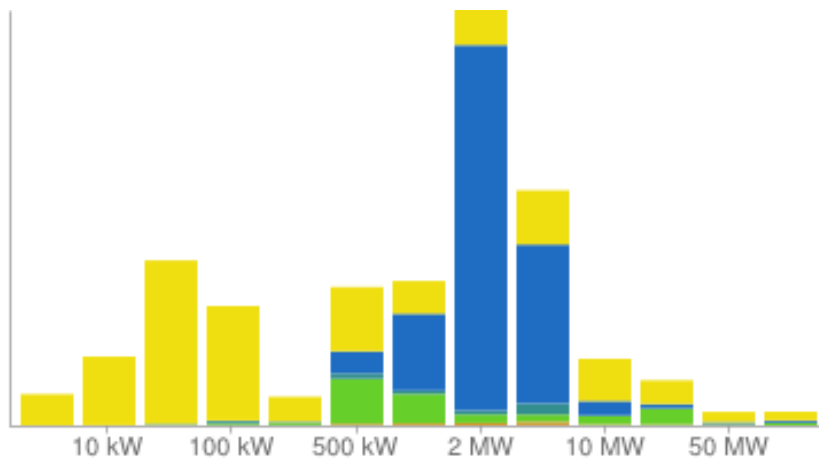
Figure 1.3: Electricity production from renewables divided by size [5].

Figure 1.2 depicts the distribution of the renewables among the different voltage levels. As seen, the major part is connected to the low voltage (230/400V) and the medium voltage (20 kV) level. Looking exclusively at the photovoltaics, the largest part is connected to the low voltage grid. Figure 1.3 depicts the distribution of the renewables after their size. As seen, a large share of the installed capacity is small-scale generators. These two figures illustrate an important effect of The German Renewable Act: the easy access to the electricity market for small and medium producers. Private households with rooftop photovoltaics, like in picture 1.4, are examples of such small-scale producers.

When travelling around in Germany, roofs of impressively many houses and barns can be seen covered by photovoltaic panels. Typically, at day time when the sun is shining and there is minimal consumption in the house, the power production can be larger than the consumption and thus surplus power is fed into the grid and the direction of power flow is reversed, even up to higher voltage levels. In the evening, on the other hand, when the load is typically at its highest, and there is zero power production from PV, the house consumes power. Thus, buildings with rooftop PV serve both as a generator and a load.

In addition to the German Renewable Energy Act, the federal government

Figure 1.4: Small scale electricity producer in Munich. Photo by author.

has developed the *National Platform for Electric Mobility in Germany* which will help Germany be the leading international market for electro mobility [3] with 1 million electric vehicles in Germany by 2020 and 6 million electric vehicles by 2030 [26]. With this increase in electro mobility in Germany, the evening peak load is expected to increase, as people tend to charge their electric vehicles in the afternoon and evening when returning from work.

Chapter 2

Voltage Characteristics

2.1 Framework for Permissible Voltage Variations

As for the rest of Europe, the general power quality in the distribution system in Germany is regulated by the standard EN 50160 *Voltage Characteristics of Public Distribution Systems*. Regarding permissible voltage variations, the following requirements from EN 50160 are especially relevant:

2.1.1 EN 50160 - Slow Voltage Variations

Definition 1 *Slow voltage variations are changes in voltage stationary root-mean-square (r.m.s.) value measured over a given time interval. [6]*

The requirements for slow voltage variations under normal operating conditions are:

- During one week, 95% of the 10 minutes mean r.m.s. values of the supply voltage shall be within the range of $\pm 10\%$

- During one week, 100% of the 10 minutes mean r.m.s. values of the supply voltage shall be within the range of $+10\%/-15\%$

2.1.2 EN 50160 - Rapid Voltage Changes

Definition 2 *A rapid voltage change (δu) is a change of the voltage r.m.s. value within $\pm 10\%$ of the agreed voltage level, which is faster than 0.5% of the agreed voltage level per second. [6]*

The requirements for δu is:

- Normal operating conditions: $\delta u \leq 5\%$

- In special cases a few times per day: $\delta u \leq 10\%$

2.1.3   Additional Framework for Germany

Low Voltage Grid Framework

For Germany, there is additional framework applicable when a generating unit is to be connected to the low voltage distribution grid. The directive *Generators connected to the low-voltage distribution network - Technical requirement for the connection to and parallel operation with low-voltage distribution networks* was published in 2011 by VDE (Verband der Elektrotechnik, Elektronik, Informationstechnik e.V.). It requires, among others, that voltage variation caused by a generating unit connected at the low voltage level may not exceed 3% [7].

Medium Voltage Grid Framework

The essential aspects which have to be taken into consideration when connecting a generating plant to the medium voltage grid is given in the technical guideline *Generating Plants Connected to the Medium-Voltage Network* published in 2007 by BDEW (Bundesverband der Energie- und Wasserwirtschaft e.V.). It states that under normal operating conditions of the network, the magnitude of the voltage changes caused by all generating plants with a point of connection to a medium-voltage network, must at no junction point within this network exceed a value of 2% as compared to the voltage without generating plants [8].

Illustration of Application of Regulations

Figure 2.1 illustrates an example how a network operator distributes the permissible ±10% voltage variation limit amongst the voltage levels in Germany. As seen in the graph, the +3% limit meets the +10% limit at the same point, but the +3% limit is absolute even though the deviation from nominal voltage is less than +10%. The limits of ±10% always apply.

2.2   Conventional Power Systems

In the following section, voltage characteristics for a conventional power system is elaborated, with the focus on slow voltage variations. Consider the simple circuit of figure 2.2. It represents a substation/transformer with voltage Us, a load with voltage Uc and the power line between them. The line is represented with its series impedance $Z = R + jX$. The resistance R represents the ohmic losses in the transmission. The reactance X represents the magnetic field that is caused by the currents in the line. In reality there is also a capacitance present, which is caused by the electric field between the lines and between the lines and the ground. However, for the sake of the coming explanations, it will be sufficient to use a simplified line diagram with only the series impedance. For clarification it should be mentioned that for the power flow simulations that are performed later in the thesis, the capacitance is not disregarded.

As shown in figure 2.2, the load consists of a parallel combination of a resistor and an inductor. This results in an *inductive load*, which often is the case of typical real life power system loads [10][13]. The reason for this is that a load in this context often is a house with all its electrical equipment. A large share of
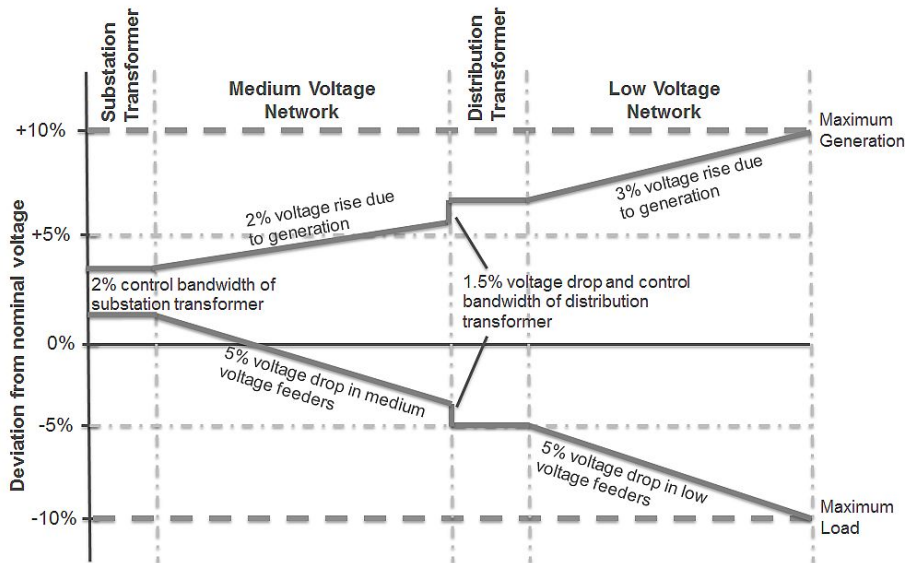
Figure 2.1: Illustration of Permissible Voltage Variations in Germany [9]

this equipment is slightly inductive, e.g. vacuum cleaners, refrigerators, freezers, washing and drying machines or dishwashers to mention some. Inductive loads operate at lagging power factor and require positive reactive power for their operation [13][14]. Instead of the term *require*, the expressions *absorb* or *consume* are often used. As the loads are inductive and demand reactive power, generators are usually overexcited, operating at lagging power factor and supply positive reactive power to the system [13]. Table 2.1 contains an overview of the subject. In this table the consumer reference system is applied.

Table 2.1: Q

System Unit	power factor	Characterization	Q- support	Sign of Q
Load	Lagging	Inductive	Consumes Q	+ Q
Generator	Lagging	Capacitive	Supplies Q	+ Q
Load	Leading	Capacitive	Supplies Q	- Q
Generator	Leading	Inductive	Consumes Q	- Q

19

Figure 2.2: Line diagram of inductive load

To explain the voltage characteristics for a power line, it can be illustrative to use phasor diagrams. Here, they serve as a per phase representation and they are simplified for illustrative purposes. The reference voltage is the phasor which is held constant. It can either be the load voltage, or the substation voltage. The reference phasor is often placed on the positive real axis, but this is no rule. Two phasor diagrams are presented in figures 2.3 and 2.4. The two diagrams illustrate the same. They only differ in which of the voltage phasors that is chosen as reference and placed on the positive real axis and so the other phasors end up in different quadrants.
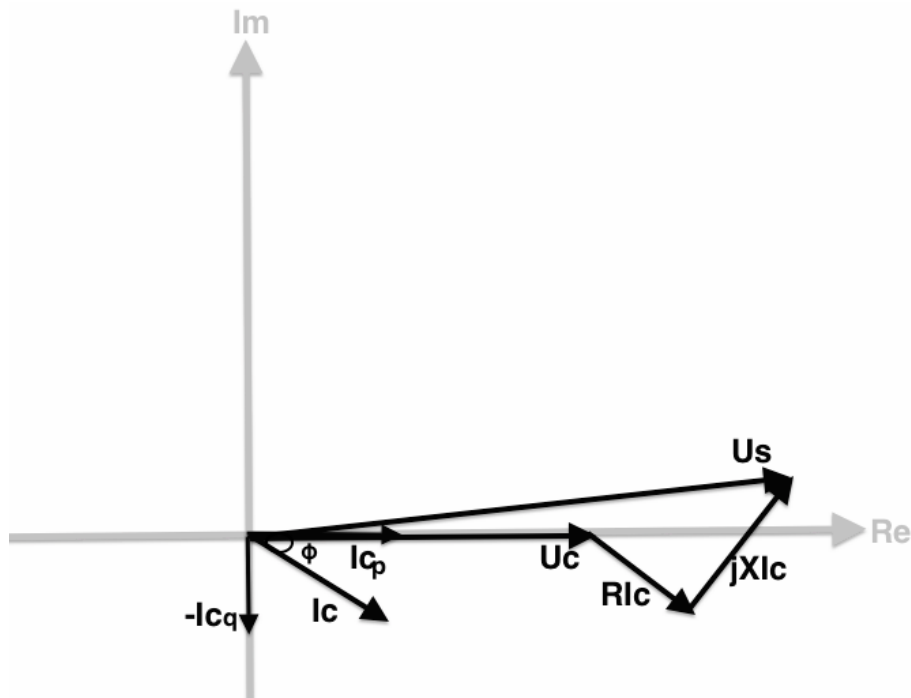
Figure 2.3: Phasor Diagram Voltage Drop 1st quadrant

An explanation of the phasors will now follow. The current at the load connection point is named Ic. It is known from basic circuit theory that the inductor branch current Icq lags behind the resistor branch current Icp by 90°. The three currents Ic, Icp and Icq are drawn in the phasor diagrams, starting from origin. Icq is in reality much smaller than Icp, so Ic lies in reality closer to the real axis.

Further, there is a voltage drop across the line impedance. The voltage drop across R_l is $R_l * Ic$ and in phase with the line current. The voltage drop across jX_l is $jX_l * Ic$ and leads the line current by 90°. These are also drawn in the phasor diagrams. For phasor diagram 2.3, the voltage at the substation Us can now be illustrated by drawing a phasor from origin to the tip of phasor $jX_l * Ic$. For phasor diagram 2.4, the voltage at the customer Uc can be illustrated by drawing a phasor from origin to the tale of phasor $R_l * Ic$.

As Ic lies closer to the real axis in reality, the voltage angle between phasor
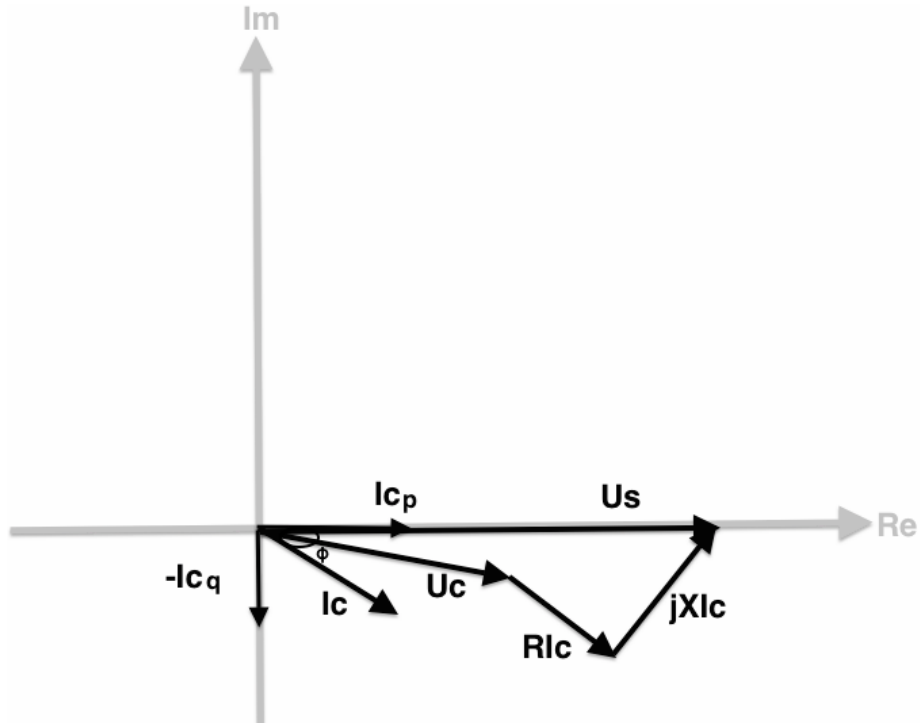
Figure 2.4: Phasor Diagram Voltage Drop 4th quadrant

Uc and Us is quite small. It can be seen that it is a good approximation to take the projection of Us on the line of Uc (or the opposite way if diagram 2.4 is applied) and use the difference between the projection and Uc as the voltage drop [10]. This can be mathematical expressed as:

$$\Delta U \approx RIc * \cos\phi + XIc * \sin\phi, \quad \phi \geq 0 \tag{2.1}$$

Where ϕ is the load angle. The apparent power at the customer connection point is [12]:

$$S = P + jQ = Uc(Ic * \cos\phi + Ic * \sin\phi) \tag{2.2}$$

Which can be reformulated to:

$$\frac{P}{Uc} = Ic * \cos\phi \tag{2.3}$$

$$\frac{Q}{Uc} = Ic * \sin\phi \tag{2.4}$$

Combining these with equation 2.1 gives the final equation, where the voltage drop is given by the power and the voltage at the customer connection point.

$$\Delta U \approx \frac{PR + QX}{Uc} \tag{2.5}$$

From this equation, it can be seen that the voltage drop increases with the power at the customer connection point and with the impedance of the line. In

high voltage transmission networks, the reactance is much higher than the resistance ($\frac{X}{R} > 10$) [11]. Because of this, even though the active power P usually is higher than the reactive power Q, the voltage variations at higher voltage levels are assumed to be highly dependent on the QX product of equation 2.5 [10]. In low voltage distributions grids on the other hand, the $\frac{X}{R}$-ratio is much smaller, around 1 or less [10][11]. Because of this, the PR product is of higher importance, meaning the active power at the customer connection point will affect the voltage drop to a large extent.

When characterizing lines and cables, the impedance is often given per kilometre. This means, that the total impedance of a line depends on the length of the line. As the voltage drop is dependent on the impedance, the voltage drop is dependent on the cable length. According to equation 2.5, the higher the impedance, the higher the voltage drop.

2.3   Power Systems with Distributed Generation

It was seen from equation 2.5 in the previous section that the active power at the load affects the voltage drop in a distribution network. What happens when active power at the load is no longer consumed, but actually produced? This can be the case of a grid connected distributed generator and the situation that arises from it will be investigated in this section.

Let us assume that the load of the circuit in figure 2.2 represents a farm, which decides to install photovoltaic panels on its rooftops. A quite realistic scenario in Germany, as explained in section 1. The new circuit that arises from this assumption is shown in figure 2.5. If the rooftop PV produces more power than the amount that is consumed at the farm, Ip ends up with a negative value and surplus power is fed into the grid.
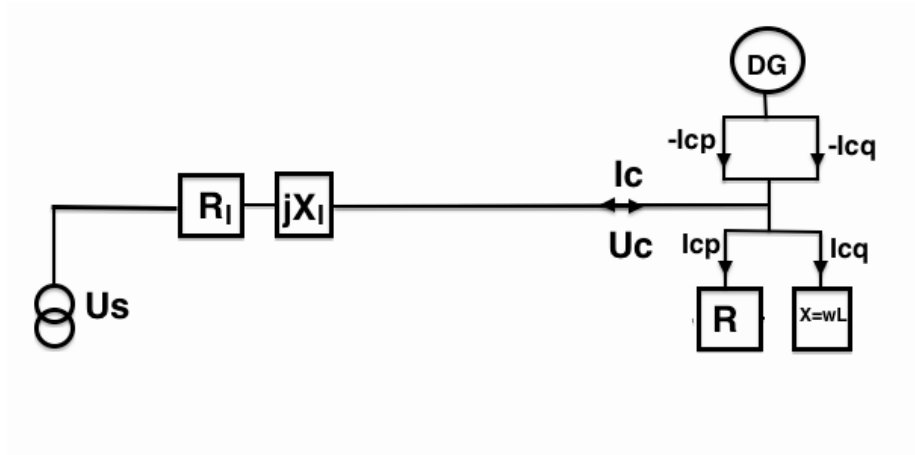
Figure 2.5: Line Diagram of Load with Generator

When the line current has a negative value like in this situation, it affects the voltages in the grid. The new situation will be illustrated through a phasor diagram. Both of the previous phasor diagrams could have been used as a basis, but here it has been found explanatory easiest to define the connection point voltage Uc as the reference phasor.

Assuming first, that no reactive power is present in the grid, no imaginary current Icq exists, and both the power produced and the injected current is purely real. The resulting situation is illustrated in phasor diagram 2.6. The current Icp changes direction and flows from the connection point to the substation. The direction of the voltage drop is reversed and it results in Us being smaller than Uc. In other words there is a *voltage rise* over the line from the substation to the connection point. This voltage rise that occur subsequent to active power injection is due to the low $\frac{X}{R}$-ratio in low voltage networks, and the high dependency of the PR-product [11]. This induced voltage rise may exceed the permissible voltage variations introduced in section 2.1. It can be problematic for other reasons as well, which will be explained in the next section.
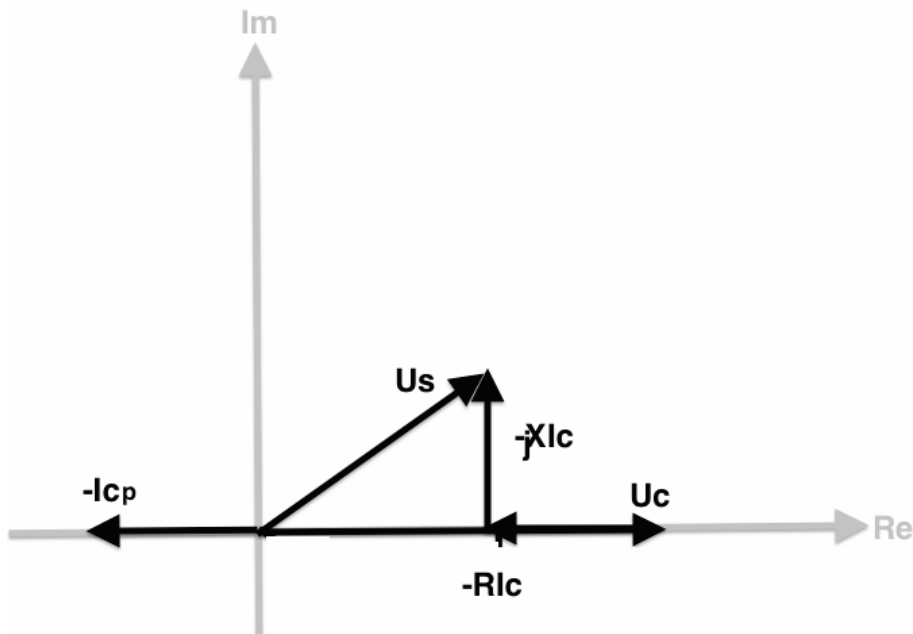
Figure 2.6: Phasor Diagram of Generator Producing Active Power

2.3.1 Physical Impact of Distributed Generation

Distributed generators introduce several new issues that could impact the power quality. Many aspects could be elaborated and investigated, but in this thesis the focus will be on the reversed power flow, the subsequent voltage rise, and the possible overvoltages that distributed generators can cause.

Overvoltages

Generally speaking, overvoltages are unwanted, as they can lead to damage of equipment and challenge proper operation of the system which are designed and optimized for operation at a certain voltage level [17]. Consequences of too high voltages can be[6]:

- Break down of insulation in electrical and electronic equipment (E.g. capacitors, transistors, surge protection)

- Excessive heat and losses which can damage equipment and lead to fire hazard

- Accelerated ageing due to high temperatures

- Malfunction and tripping

To avoid these consequences, the voltage variation limits presented in section 2.1 must be followed.

Other Issues

In addition to the already mentioned overvoltages, the reversed power flow can introduces other challenges for the grid operators. Examples of such challenges are:

- Reverse power flow may interact with the power system's voltage-regulating devices in an unfavourable way, which can lead to incorrect operation of control equipment. [15]

- Conventional fault detection systems may fail, e.g. fuses that are designed to protect the current carrying capability of a line, will not detect power injection of power downstream from a fuse, leading to a potential for overload. [16]

Influence on Undervoltages

The longer the network feeder, the higher the feeder impedance, and the larger the voltage drop along the feeder. Thus, for especially remote network users, the voltage drop could be undesirably large and result in an undervoltage. In Germany, there haven't been any issues regarding undervoltages up to now. The situation could be different in other countries. An analysis of 142 customer complaints in Norway, presented in [6], showed that 14% of the complaints was related to undervoltages. This was the second largest share of the complaints. Only the complaints on voltage dips, which accounted for 23%, was higher. The grid operators are obliged to keep the voltage at the customers connection points within a certain range, as was explained in section 2.1. Network user appliances are typically designed to tolerate supply voltages in this range and if the voltage drop is too large, it will have consequences for the network users appliances. Examples of such consequences are [6]:

- The power in ohmic appliances (e.g. heaters, stoves, light) is dependent on the square of the voltage ($P = \frac{U^2}{R}$). Low voltages in ohmic equipment will lead to reduced output power/light.

- Some electronics will compensate for reduced voltage by increasing the current to keep the power constant ($P = U * I$). This can lead to overheating and damage of equipment.

- Asynchronous motors do not come up to speed.

- Malfunction and tripping

Adding a generating unit on a line with significant voltage drop, could give a voltage rise and thus improve the voltage profile. This way some of the mentioned issues could be avoided.

Chapter 3

Statistics and Probability

An important aim for the master's thesis is to construct a Monte Carlo simulation model. The model will help estimate the network impact from a high penetration of distributed photovoltaics and electric vehicles. As the Monte Carlo model is a statistical problem solving technique, a theoretical foundation of statistical terms and methods is necessary before further explanations are carried out.

3.1 Collection of Data; the Sample

The collection of data for statistical testing results is what we call a *sample*. Samples are collected from *populations* that are collections of all individuals or individual items of a particular type and the sample size n is the number of elements in the sample [18]. As the whole population seldom is available for analysis, a sample is taken from the population, and with the help of statistical methods and elements of probability, the sample allows us to draw conclusions about the population as a whole.

3.1.1 Sample Mean

The sample mean is a numerical average, denoted by $\bar{x}$ and defined as:

$$\bar{x} = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i \tag{3.1}$$

It gives a quantitative measure of where the data center of the sample is and serves as en estimate for the population mean [18].

3.1.2 Sample Standard Deviation

A common way to measure spread or variability of a sample is the *standard deviation*. The real standard deviation is denoted σ_x. The sample standard deviation is denoted by S_x and defined as:

$$S_x = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \overline{x})^2} \tag{3.2}$$

3.2 Probability

3.2.1 Sample Space and Events

Consider the simple experiment: *Tossing of two coins*. There are two possible outcomes per coin: Heads (H) or Tails (T). The set of all possible outcomes of a statistical experiment is called the sample space and is represented by the symbol S [18]. For the coin example, the sample space would be

$$S = \{HH, HT, TH, TT\} \tag{3.3}$$

All the outcomes are subsets of the sample space. In statistical theory this is called an *event*. Each of the outcomes in S is equally likely to occur, if the coin is balanced [18].

3.2.2 Probability of an Event

The probability of an event A is found by summing the probabilities assigned to the sample points in A [18]. E.g. if A represent the event "At least one head will occur when two coins are tossed", then

$$A = \{HH, HT, TH\} \tag{3.4}$$

$$P(A) = \frac{1}{4} + \frac{1}{4} + \frac{1}{4} = \frac{3}{4} \tag{3.5}$$

3.3 Random Variable

A random variable is a function that associates a real number with each element in the sample space [18]. For the example with the two coins, a random number can be e.g. the number of heads that occur per toss. The values of this random number, let us call it R, can be 0, 1 or 2. If the coins are tossed 16 times, and the result is 4 tosses with no head, 7 tosses with one head and 5 tosses with two heads, then the value of R would be:

$$R = (0) * \frac{4}{16} + (1) * \frac{7}{16} + (2) * \frac{5}{16} = 1.06 \tag{3.6}$$

As most of us would probably guess, it is close to 1.

3.3.1 Mean of a Random Variable

The calculated R in the previous section, represents the average number of heads per toss of the two coins for 16 tosses. The *relative frequencies* of the different values of R (0, 1 and 2) are the fractions in equation 3.6. According to [18], in the long run, the relative frequencies equals the probabilities of the

given events. Thus, if two fair coins were tossed, it follows that

$$P(R = 0) = P(TT) = \frac{1}{4} \tag{3.7}$$

$$P(R = 1) = P(HT) + P(TH) = \frac{1}{4} + \frac{1}{4} = \frac{1}{2} \tag{3.8}$$

$$P(R = 2) = P(HH) = \frac{1}{4} \tag{3.9}$$

From this we can calculate the expected value of R:

$$E(R) = (0)*P(R = 0) + (1)*P(R = 1) + (2)*P(R = 2) = 0*\frac{1}{4} + 1*\frac{1}{2} + 2*\frac{1}{4} = 1$$

That is, tossing of two fair coins a very large number of times, gives on the average one head per toss.

3.4 Monte Carlo Simulation

In this thesis, a Monte Carlo simulation model is developed. A general explanation of the concept follows here. A Monte Carlo simulation is a very broad term. It encompasses every method based on the use of random numbers and probability statistics to investigate problems. A definition of the method is proposed in [19]:

Definition 3 *A Monte Carlo simulation is a problem solving technique used to approximate the probability of certain outcomes by running multiple trial runs, called simulations, using random variables.*

The simulation method is named after the city Monte Carlo in Monaco, known for its casinos with gambling games. These gambling games, such as roulette, dice, or slot machines, are based on random behaviour.

The method can be found applied in many different areas, but common for them all is that they try to model the future. The future is unknown, but past data and experience are known. The input data to the model is based on the known past with a certain range of possibilities. This range of possibilities is the basis for the random number generation. The model performs its calculation using the random number. The calculation is repeated several times, each time with a new random number. This is referred to as repeated sampling.

The output will be a large number of results with a certain range. Based on this range, the likelihood (mean, standard deviation etc.) of the future projection can be found. According to [20], this is exactly the key feature of the Monte Carlo simulation: based on how you create the ranges of estimates, it can tell you how likely the resulting outcomes are.

Part III

Analysis

Chapter 4

Overview of the Analyzed Grid

4.1 Background

The city Garmisch-Partenkirchen is for the time being subject to several research studies, due to the German government funded project *Modellkommune Elektromobilität Garmisch-Partenkirchen* or simply *e-GAP*. This project was started in July 2012 as a part of a five-point strategy for developing better electro mobility in the Bundesland Bavaria in Germany. A part of this strategy is *Smart Grid - Basis einer elektromobilen Zukunft*, a research project that deals with the creation of a smart grid in Garmisch-Partenkirchen. The city already has a large share of photovoltaic plants connected to the low voltage grid and the installation of photovoltaic capacity is expected to increase significantly. This, in addition to the planned growing electro mobility in Garmisch-Partenkirchen, introduces possible challenges.

4.2 Garmisch-Partenkirchen

Garmisch-Partenkirchen is a mountain resort town located in the very south of Germany with 27 890 inhabitants. The city will from here on be referred to as GaP. The electricity network of GaP covers a geographical area of 200.55 km^2 and on November 26th 2013 at 16:45 the utility of GaP registered its peak power of 26 611 kW [21]. Some basic data for the electricity network of GaP is provided in table 4.1. Numbers are and based on the situation 31.12.2013.

Table 4.1: Data for the electricity network in GaP

	0.4 kV network	10 kV network
Length over head lines	0 km	6.835 km
Length cables	233.459 km	182.338 km
Energy delivered in 2013	98 804 418 kWh	21 972 373 kWh
Number of supply points	22 391	10

4.3   Current Energy Situation in Garmisch-Partenkirchen

All figures in this section are based on numbers from the utility of GaP ([22]). Figure 4.1 gives an overview of the current energy sources of the power generation in GaP. As seen, the largest share is generated from photovoltaics.
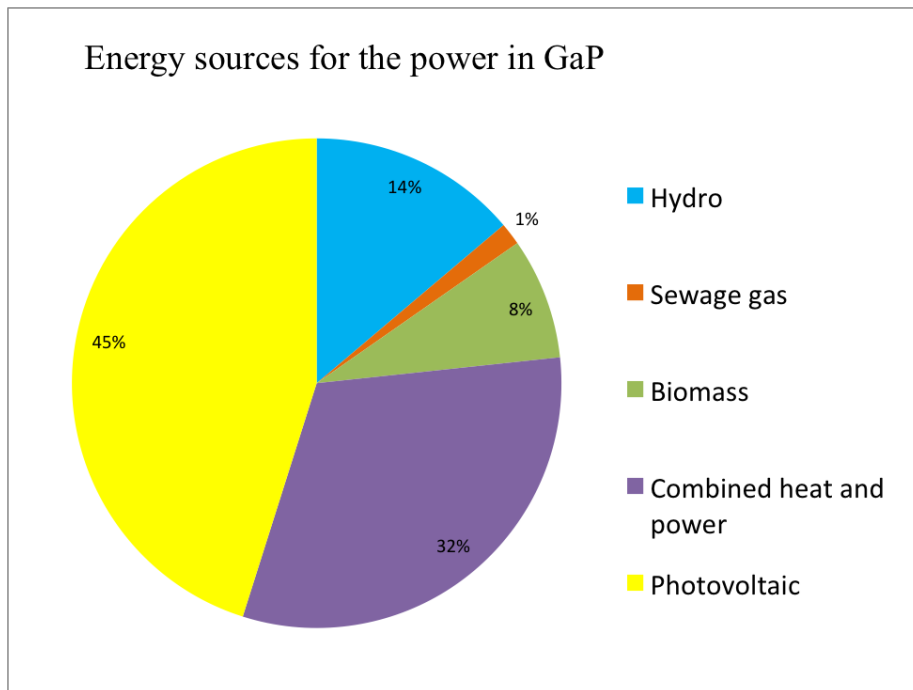
Figure 4.1: Energy Sources GaP

The German Renewable Act led to a rapid increase in installation of photovoltaics in GaP. This is clearly shown in 4.2, where all the solar units in GaP are sorted after their commission date.
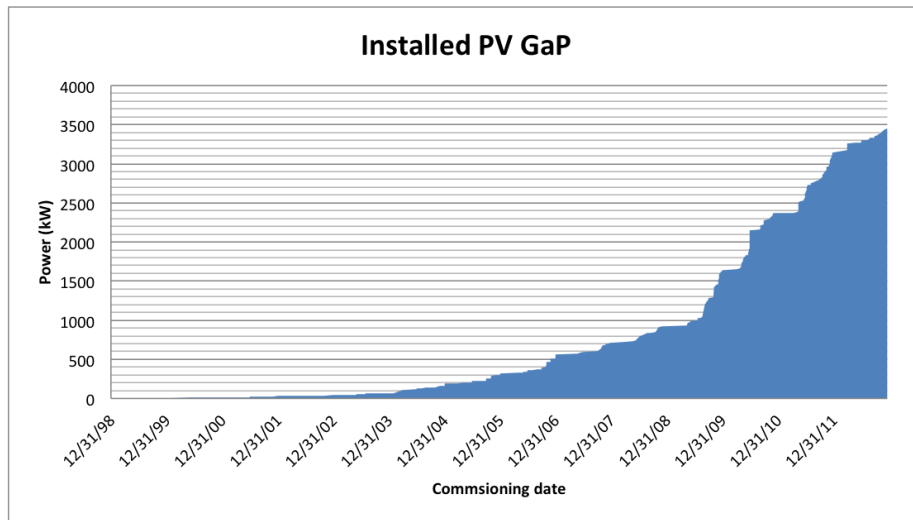
Figure 4.2: PV installation in LV grid of GaP after commissioning date

After a steady yearly increase since the act came into force, 295 PV units are by the end of 2012 connected to the low voltage distribution grid, together making up a photovoltaic capacity of 3451.418 kW of electricity. In addition, two larger PV parks are connected to the medium voltage grid. An overview is given in table 4.2

Table 4.2: Sum installed photovoltaic capacity in GaP

Low Voltage Grid (0.4 kV)	3451.418 kW
Medium Voltage Grid (10 kV)	1756.520 kW
Total	**5207.938 kW**

Chapter 5

Power Flow Simulation Model

5.1 OpenDSS

For the power flow simulations, Matlab and the open-source simulation tool Open Distribution System Simulator (OpenDSS) were chosen. OpenDSS is provided by Electric Power Research Institute and developed for the purpose of distribution grid simulations. It is possible to use OpenDSS as an independent program or as a *Component Object Model* (*COM*). The latter variant is what is used in this thesis and it allows the integration of OpenDSS in Matlab. OpenDSS is script-based, which means that the grid is given as several (*.dss) files. These files contain power system circuits with it's lines, transformers, loads and generators. These are all modelled as three phase symmetrical components. Figure 5.1 shows an example on how the different components are defined in OpenDSS.

```
New Transformer.S03-Trafo Windings=2 XHL=4.038056 phases=3
~ wdg=1 conn=wye Bus=MV_bus_S1 kV=10.0 kVA=400.0 %r=0.355
~ wdg=2 conn=wye Bus=LV_bus1 kV=0.4  kVA=400.0 %r=0.355

 New Line.L1004 bus1=LV_bus1 bus2=N184 linecode=NAYY4x50 length=0.007 phases=3

 New Load.3011 Pf=0.894 bus1=N628 kW=2.0 kvar=1.0 phases=3

 New Generator.PV_SnowStreet29 phases=3 bus1=N406 kv=0.4 kw=5.000000 pf=1.000000 model=1
```

Figure 5.1: Example of the component definition in a (*.dss) file

In order to do simulations on the grid of Garmisch-Partenkirchen, the grid had to be digitalized and presented as (*.dss) files like in figure 5.1. The files were constructed based on real grid values provided by the utility of Garmisch-Partenkirchen. The digitalization process that was performed prior to the power flow simulations is explained in the following section.

5.2 Grid Digitalization

To start with, the networks were depicted as two-dimensional drawings in the form of (*.dwg) files (DraWinG-files). Converting the grids from (*.dwg) to (*.dss) is a process which requires several steps. First, the data is put to a network calculation program. Version 5.5.1 of NEPLAN has been found suitable for this task. With NEPLAN it is possible to manually build the grid with the correct component characteristics and correct topology. Two different kinds of (*.dwg) files of the grid are provided by the utility of GaP:

- The *Connection Plan* containing cable characteristics and switch and fuse states.

- The *Network Plan* containing cable lengths and the correct topology of the grid. In addition the streets and houses are visible, which means that it is possible to identify the position of the loads (customers).

The *Network Plan* is integrated in NEPLAN as a background image and the grid was build in NEPLAN by drawing the cables exactly on top of the background image. Also a coordinate system had to be specified, in order for NEPLAN to calculate the exact lengths and the positions of the cables and the loads. This is done by looking up the XY - coordinates of two different locations (e.g. corners of two different houses) in the (*.dwg) file, and add these coordinates to the exact same locations in the NEPLAN file. Figure 5.2 shows how the NEPLAN elements will look in the network model. Typically, each distribution transformer with it's underlying grid are digitalized one at a time and saved as separate (*.dss) files.
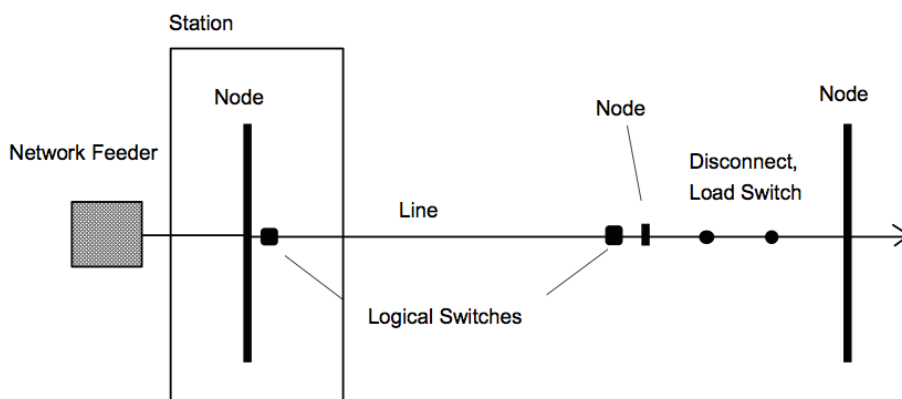
Figure 5.2: Example of the components in a NEPLAN grid. [34]

The cable types of each line segment has to be found in the *Connection Plan* and then registered in NEPLAN by looking up the cable type in the NEPLAN Cable Library. In addition, the loads, i.e. the customer connections, and the fuse boxes have to be drawn in the same network model and assigned the correct customer connection ID. As later chapters will show, the Monte Carlo simulations results in a large excel file containing all the buildings in GaP and

information on whether they have a rooftop PV or not. In this excel file, the buildings are named after their customer connection ID's. This excel file is the basis for the power flow simulations and the MatLab power flow script will add the PV plants from the excel sheet as generators to the (*.dss) file. In order for the power flow model to know on which roof tops to place the generators, it is important that the customer ID's in the (*.dss) file correspond to the customer ID's in the Monte Carlo excel result file.

When the drawing of the low voltage distribution grid and it's components is finished, a power flow is run as a test in NEPLAN. The power flow verifies whether everything is properly connected. If the power flow converges, the project file is ready for export.

Four files need to be exported from NEPLAN in order to obtain a (*.dss) file in the end. The four files are:

- loads and feeders(*.ndb)

- topology data(*.zdb)

- buses (*.ndt)

- element table (*.edt)

The Forschungstelle für Energiewirtshaft in Munich has developed a tool that can convert these four files into one single (*.dss) file. The tool is constructed by scripts in python programming language. This is the very last step of the digitalization process.

As the grid of Garmisch-Partenkirchen is subject to a lot of research, the digitalization is an ongoing process. At the time of the power flow calculations, the following parts of the grid were digitalized:

- Complete 10 kV distribution network

- The following 0.4 kV distribution network parts (Numbered after their 10/0.4 kV transformer): 3, 23, 44, 56, 57, 60, 63, 65, 66, 75, 84, 85, 88, 92, 100 and 147.

The author of this thesis stayed two weeks in Munich to participate in the digitalization process. During this stay, three transformers were digitalized: Number 23, 44 and 100. These were chosen in order to have a selection of each sector, in addition to cover areas with different properties. Transformer 23 lies in a suburban area and has a high share of industrial loads. Transformer 44 is located in a rural area and supplies a particular high share of agricultural loads. Transformer 100 is placed in the city centre and supplies 100% residential loads.

In general, the low voltage distribution grids that are digitalized up to now represent examples from the three different sectors in GaP and they cover rural, suburban and urban areas.

The 10 kV distribution network, is connected to the higher voltage levels through three substations transforming the voltage between 110 kV and 10 kV. These all have a rated power of $S_N = 31.5 MVA$. The slack bus is defined at the 110 kV bus.

5.3 Load Modelling

The customer connection points are modelled as loads in the (*.dss) file (see example in figure 5.1). The power demand of the loads can be scaled using load profiles, assigned a constant power or set inactive to clearly see the effect of the loads in conjugation with distributed generation. In this thesis there will be no time simulation due to the large amount of data. Instead, snapshots will be taken at two different dates and times and the situation at these times will be studied.

Two assumptions are made. First, all loads are assigned the same average power. As 91 % of the buildings belong in the residential sector, the loads are calculated based on numbers for an average residential building in Garmisch-Partenkirchen. Second, many buildings are connected to distribution grids that are not yet digitalized. This is solved in the following manner: the number of buildings connected to the same transformer are aggregated and the respective transformer is assigned the total load value of all the buildings. In this case, the transformer itself is considered a large load.

5.3.1 Active Power of Loads

On the web page of the utility of GaP, both production and normalized load profiles for the year of 2011 are available, see [35]. The standardized load profiles represent the average over many households in GaP and are given as 15 minutes averages. The loads are normalized based on a yearly energy consumption of 1 GWh. E.g., from the standardized load profile given in figure 5.3, the energy consumption at the 15 minute interval beginning at 14.00 is found to be 34.66 kWh. This means that in the 15 minutes between 14.00 and 14.15 an average household in GaP will consume $\frac{34.66kWh}{1GWh}$ of the total energy it consumes in one year. From this, it is possible to calculate the average power of the customer for the different 15 minutes intervals. The calculation will be explained in the following paragraphs. All graphs are constructed from [35].

Summer Day Scenario: Minimum Consumption - Maximum Production

A mid-July day will serve as a good example of a low load - high production scenario. Figures 5.3 and 5.4 represent the situation in GaP on July 19th 2011, and it can be seen that 14.00 is a time where the load is low and the production from PV is high. Thus, the average load for this time frame will be calculated and used as the value in power flow calculations for *summer day scenario*.

In the standardized load profile of figure 5.3, it can be seen that the energy consumption at the 15 minute interval beginning at 14.00 is 34.66 kWh.

$$W_{14.00} = 34.66kWh \tag{5.1}$$

The load profile is normalized with a yearly energy consumption of 1 GWh, so a normalization factor a can be defined:

$$a = \frac{W_{14.00}}{W_{year}} = \frac{34.66kWh}{10^6 kWh} = 0.00003466 \tag{5.2}$$
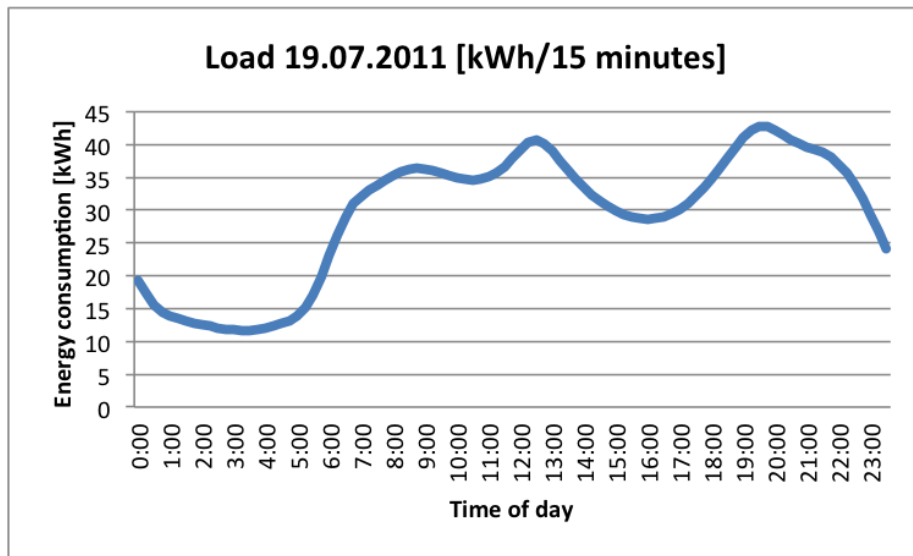
Figure 5.3: Average household load in GaP on summer day
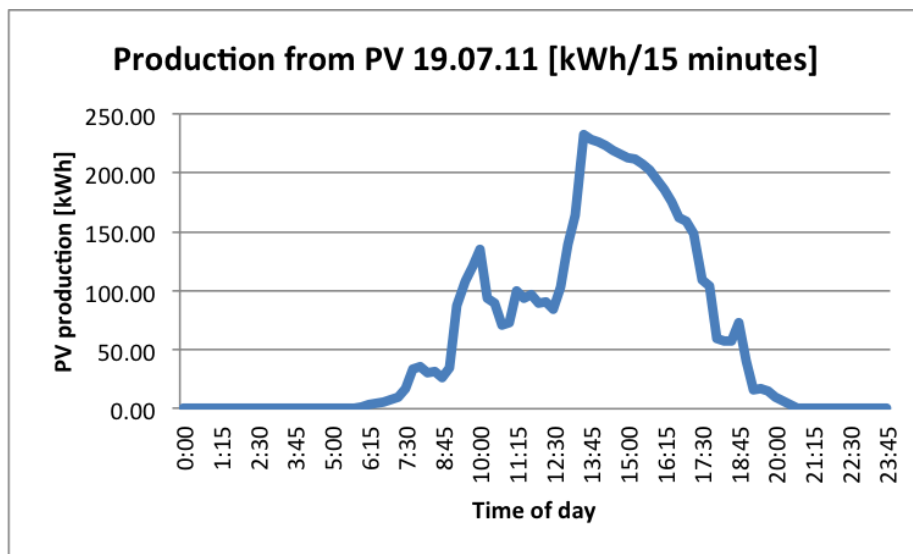
Figure 5.4: Production from PV in GaP on summer day

The annual energy consumption of an average household in Germany is assumed to be 4000 kWh, so the consumption for one household in 15 minutes is:

$$W_{15min} = a * 4000kWh = 138.64Wh \tag{5.3}$$

The average power within this 15 minutes time frame can then be found as:

$$P_{15min} = \frac{W_{15min}}{\Delta T} = \frac{138.64Wh}{0.25h} = 555W \qquad (5.4)$$

That is, for the *summer day scenario*, all loads are assigned $P = 555W$.

Winter Evening Scenario: Maximum Consumption - Minimum Production

For a high load - low production scenario, profiles from January 8th 2011 will be used. They are depicted in figures 5.5 and 5.6. The highest load is registered at 19.00 and at this time, there is no production from photovoltaics. Thus, the average load for this time frame will be calculated and used as the value in power flow calculations for *winter evening scenario*.
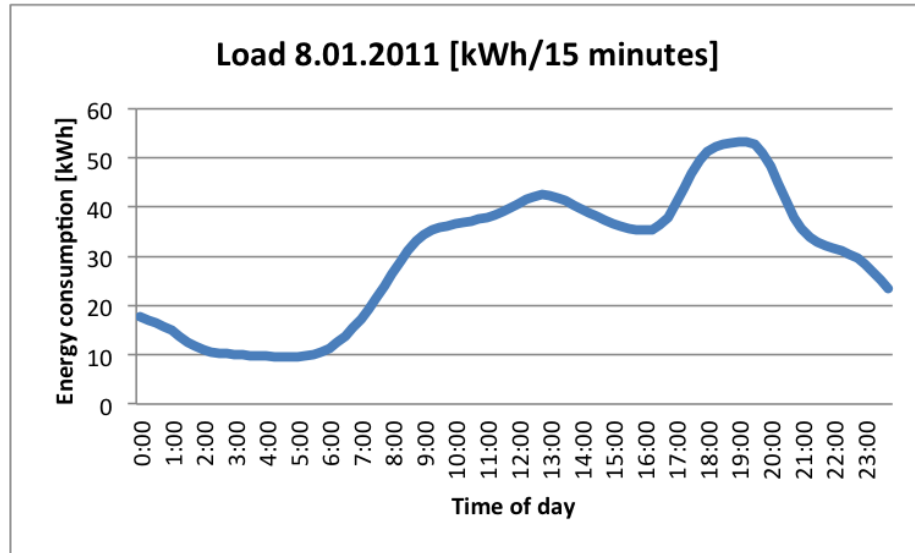
Figure 5.5: Average household load in GaP on winter day

In the standardized load profile of figure 5.5, it can be seen that the energy consumption at the 15 minute interval beginning at 19.00 is 53.399 kWh.

$$W_{19.00} = 53.399kWh \qquad (5.5)$$

The load profile is normalized with a yearly energy consumption of 1 GWh, so a normalization factor a can be defined:

$$a = \frac{W_{19.00}}{W_{year}} = \frac{53.399kWh}{10^6kWh} = 0.0000534 \qquad (5.6)$$

The annual energy consumption of an average household in Germany is assumed to be 4000 kWh, so the consumption for one household in 15 minutes is:

$$W_{15min} = a * 4000kWh = 213.60Wh \qquad (5.7)$$

The average power within this 15 minutes time frame can then be found as:
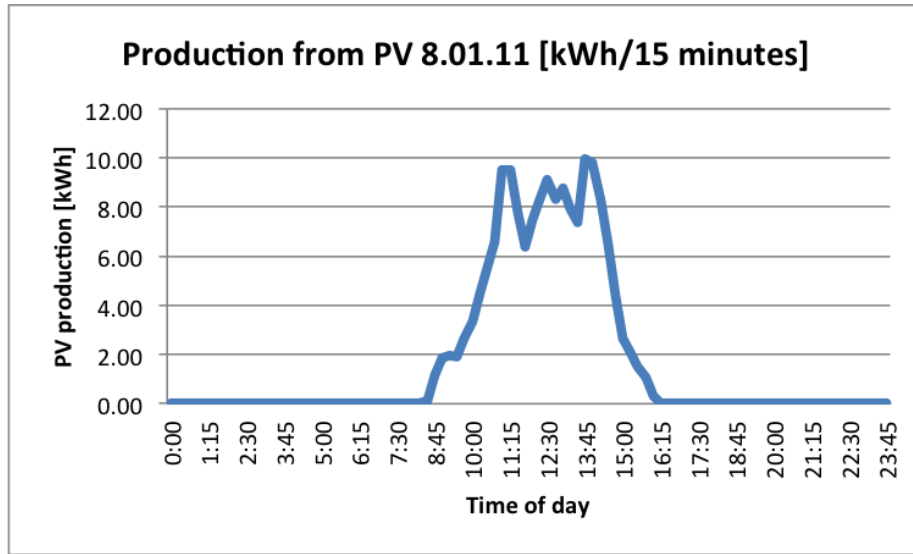
Figure 5.6: Production from PV in GaP on winter day

$$P_{15min} = \frac{W_{15min}}{\Delta T} = \frac{213.60Wh}{0.25h} = 854W \tag{5.8}$$

That is, for *winter evening scenario*, all loads are assigned $P = 854W$.

5.3.2 Reactive Power of Loads

As explained in section 2, the voltage is highly influenced by the amount of reactive power. It is therefore important to define the reactive power Q when characterizing the loads.

The power factor (pf) is a common way to express the active power share of the load apparent power, and thus indirectly also the reactive power share. Referring to the phasor diagrams of section 2, the power factor is defined as the cosine of the phase angle between the voltage and current phasor [37]. This equals the active power share of the apparent power and can be expressed as in equation 5.9.

$$pf = \cos(\phi) = \frac{P}{S} \tag{5.9}$$

The tangent of the phase angle is the fraction of Q to P

$$\tan(\phi) = \frac{Q}{P} \tag{5.10}$$

Combining equations 5.9 and 5.10 leads to the result in 5.11. With this equation, the reactive power of the load connection point can be found, if P and pf are known.

$$Q = P * \tan(\arccos(pf)) \tag{5.11}$$

As the share between reactive and active power in a household changes throughout the day, the power factor will also vary. In addition, the power factor differ

from load to load. It is therefore not evident which power factor that should be chosen for the loads. According to [36], when households are modelled as loads in a power system in Germany, a power factor of 0.9 can be assumed. As elaborated in section 2.2, the lagging power factor is due to the many inductive appliances present in households. In other countries, such as Norway, where heating mainly is electricity based, the ohmic loading share is much larger. This means a less inductive power factor of the house as a whole, and certain utilities in Norway use a standard power factor as high as 0.97 for households.

Referring to table 2.1, inductive loads consume reactive power. A high power factor, means a less inductive load and thus less consumption of reactive power. As was explained by the phasor diagram in figure 8.1, consumption of reactive power at the point of connection for a generating unit (e.g. a rooftop PV plant) will help mitigate the voltage rise that the distributed generator can cause. The choice of power factor in load modelling can thus give tremendous effects. For this reason, it was decided to measure the power factor of the grid in GaP to get higher level of accuracy in the simulations. Measurements of the loads when no PV is present have been performed and they show that the power factor always is higher than 0.95. For this reason, pf = 0.95 is chosen and for simplicity, the same power factor is applied for all loads.

Chapter 6

Studies on Photovoltaics

6.1 Future Scenario of Photovoltaic Development in GaP

6.1.1 Potential and Assumed Development

The installed PV power in GaP has rapidly increased every year since the introduction of the German Renewable Act, as was seen in figure 4.2. An important question is how this trend will develop in the future. A study conducted by C. Gerdiken at the Hamburg University of Applied Science ([23]) in cooperation with Forschungstelle für Energiewirtschaft (FfE), addresses this topic in detail. The study has found the effective usable roof area in GaP and thus the accessible rooftop PV potential of the municipality. The existing building mass was analysed in the context of the following properties:

- Azimuth angles

- Shadowing

- Rooftop types (e.g. flat, hipped, saddelback or tent roofs)

- Building type (e.g. block, detached or semi-detached houses)

- Building sectors (e.g. residential, agriculture, industry

The work resulted in an extensive database of all the houses in GaP, showing a large theoretical potential of 70 MW of rooftop PV. But to which extent will this potential be exploited? The study found that economical incentives are the most important parameter for the decision whether to install a PV-plant or not. Thus, the development of the German Renewable Energy Act and its planned future economical incentives is important to find future scenarios of PV-installation. On basis of this, the study found reasonable expansion scenarios for the different sectors in GaP and the result can be seen in figure 6.1. The figure is extracted from the data found in the study and illustrates the expected expansion of installed rooftop photovoltaics in GaP.
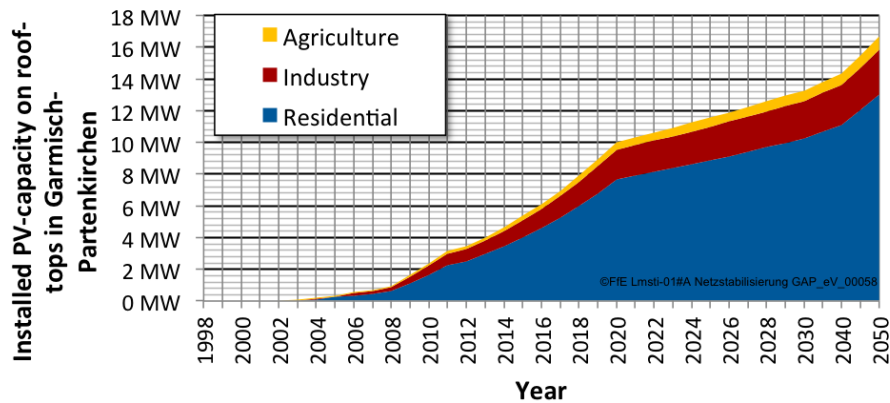
Figure 6.1: Installed PV-capacity on rooftops in GaP [23]

6.1.2 Spatial Distribution of the Expected PV

The database of the houses in GaP serves as a basis for this thesis. From this, the theoretical PV potential per house is known, in addition to the assumption of the expected increase of total PV capacity per sector. However, exactly where in the grid the expected PV will be installed, is unknown. As explained in section 2, distributed generation might influence the grid in an undesirable way. Some parts of the grid can be more vulnerable than others, e.g. it is more critical if a high number of PV plants are connected at long lines or remote feeder ends, as the impedance and thus the line length has a large impact on voltage profile along the line. Thus, for examining whether the expected PV-increase in GaP will cause grid problems or not, investigations of the location of the PV-plants is important. In this thesis a probabilistic approach to the spatial distribution of PV plants will be developed. Using MatLab, a Monte Carlo simulation model is created.

6.2 Monte Carlo Simulation Model for Photovoltaics

The number of different possible combinations for the distribution of the forecast among the houses in GaP is infinitely large. For this reason, a large number of simulations will be performed and the results will be saved, in order to statistically examine the effect on the grid with regard to voltage issues, transformer and/or line overloading.

The main unknown parameter for the model is on which buildings rooftop PV plants will be installed and on which will not. This is a choice of the house owner, which depends on numerous unknown factors, e.g. the house owner's economy or environmental consciousness, to mention some. For this reason, to simplify the complexity of the model, the parameter is treated as random. Running a large number of simulations of a system with such random variation of input parameters is the concept of *Monte Carlo simulation*, which was explained in section 3.4. For every house, a random number r is generated and this random number represents the choice whether to install *(r = 1)* or not install *(r = 2)* a PV plant on the respective rooftop. The probability of the random number r is dependent on the sector, to which the house belongs. The sectors in Garmisch-Partenkirchen are defined to be:

- Residential

- Industrial

- Agricultural

The forecast in [23], has taken the three sectors into account, and numbers for the PV capacity exist for both present and future buildings mass. The probabilities of the random numbers will be calculated in section 6.2.2 in order to reach the total expected PV in GaP.

6.2.1 Size of Photovoltaic Generators

On November 7th 2006, the first amendment of the Renewable Energy Act came into effect. This inhibited some important changes regarding the data publications of the renewable energy installations. From this date on, the network operators were obliged to publish information on their website regarding all grid connected producers of electricity from renewable energy sources. The information to be published, were among others, the power, the site and energy source of the electricity production. Based on this information, Die Forschungstelle für Energiewirtschaft (FfE) in Munich developed a database containing all renewable energy sources in Germany. From this data base, values for typical rooftop PV installations has been calculated. When the Monte Carlo model runs through all the buildings in GaP, it checks the sector of the building and assigns the random number with a probability that will be calculated in section 6.2.2. If the building in question is assigned a random number $r = 1$, a PV plant will be installed. In this case, the model verifies the rooftop size of the respective house, and assigns one of three values, dependent on what the roof can fit: no PV, a small PV plant or a large PV plant. For each sector, common values for small and large rooftop PV installations are defined and given in table

6.1. These are based on the data base from FfE of presently installed PV plants in Germany.

Table 6.1: Typical sizes for rooftop PV installations

Sector	PV plant large [kW]	PV plant small [kW]
Residential	10	3
Agricultural	30	10
Industrial	75	20

6.2.2 Probability Tuning

The forecast of future PV installation in GaP were depicted in figure 6.1. The numbers for year 2030 are given in table 6.2, together with the installed capacity of year 2012. The numbers for 2030 are the amount that the Monte Carlo model is going to distribute among the houses in GaP.

Table 6.2: Installed PV in LV grid of GaP

Sector	PV capacity in 2012 [MW]	PV capacity in 2030 [MW]
Residential	2.44026	10.2465
Agricultural	0.15408	0.6610
Industrial	0.851	2.3726
Total	**3.4513**	**13.2801**

The exact number of houses in GaP that can fit the different configurations in table 6.1 are known due to the studies in [23]. In order to reach the forecast per sector for the year 2030 by assigning the predefined PV-sizes in table 6.1, the random number must be calculated. The following equation can be applied:

$$\frac{P_{\text{increase sector}}[kW]}{N_{\text{sector}}} = probability*$$

$$\left[(\frac{N_{\text{P<Psmall}}}{N_{\text{sector}}} * 0kW) + (\frac{N_{\text{Psmall<=P<Plarge}}}{N_{\text{sector}}} * P_{\text{small}}[kW]) + (\frac{N_{\text{Plarge<=P}}}{N_{\text{sector}}} * P_{\text{large}}[kW]) \right]$$

Solving for the probability gives equation 6.1.

$$prob = \frac{P_{\text{increase sector}}[kW]}{(N_{\text{Psmall<=P<Plarge}} * P_{\text{small}}[kW]) + (N_{\text{Plarge<=P}} * P_{\text{large}}[kW])} \quad (6.1)$$

$P_{\text{increase sector}}$ is found by subtracting the already installed PV capacity from the forecast (column two from column three in table 6.2). N is the number of houses that can fit the different PV configurations. The houses that already have an installed rooftop PV are not included in N, as it is assumed that they will remain the same installation also in the future. In equations 6.2 - 6.4, the probability per sector is calculated.

$$probability(residential) = \frac{(10246.5 - 2440.26)kW}{(3689 * 3kW) + (1796 * 10kW)} = 0.2689 \quad (6.2)$$

$$probability(agricultural) = \frac{(661 - 154)kW}{(88 * 10kW) + (16 * 30kW)} = 0.3727 \qquad (6.3)$$

$$probability(industrial) = \frac{(2372.6 - 851)kW}{(96 * 20kW) + (8 * 75kW)} = 0.6038 \qquad (6.4)$$

This leads to the probability distribution in figure 6.2.
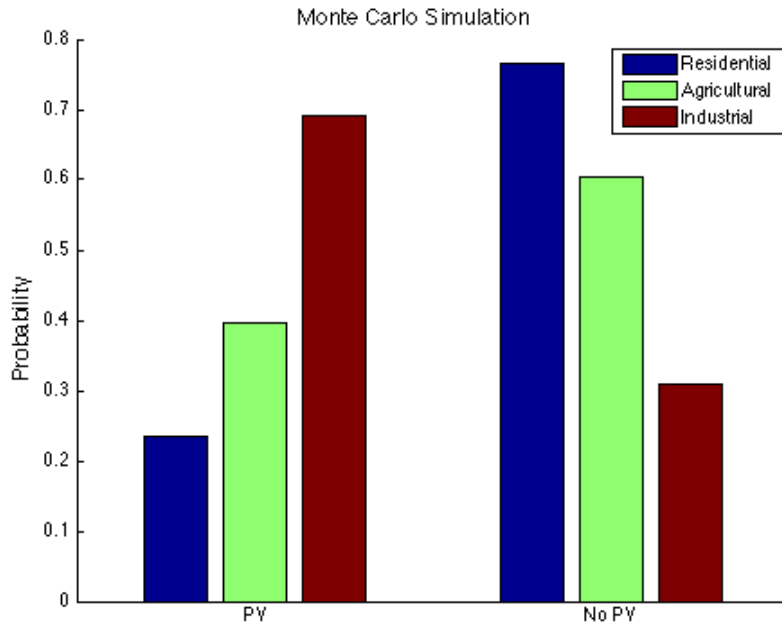
Figure 6.2: Probability distribution for Monte Carlo simulation

The assignment of the random number to each house can be considered as tossing a coin, where one toss equals one house. As was elaborated in section 3.3.1, the properties of tossing a coin several times is:

- There is two possible outcomes of every toss

- Each toss is independent from the outcome of the other tosses

The same two principles yield when assigning the random number to a house. Each house is assigned a random variable with two possible outcomes (PV or no PV) which is independent from the outcome of the other houses. Thus, *when repeated several times*, the mean or the expected value of the random variable is given by the probability distribution of the two outcomes. Since there is a high number of houses in every sector (equivalent to a high number of tosses), the mean of the random variable will end up close to the calculated number from

equations 6.2 - 6.4. It will later be shown that the percentage error between the forecast and the calculated values is less for the residential sector which has a much higher number of buildings, than for the two other sector which has a smaller number of buildings.

Thus, when all houses install a PV plant based on the random variable, which again are based on the forecast and the actual number of houses in each sector that can fit the different PV sizes, the total PV installation should aggregate up to approximately the forecast.

Figure 6.3 shows a plot of 100 Monte Carlo simulations, where the random numbers of figure 6.2 are applied. Each simulation gives exactly one distribution of PV plants amongst the rooftops of GaP. This figure shows that the total aggregated power is quite close to the numbers in table 6.2 every time.
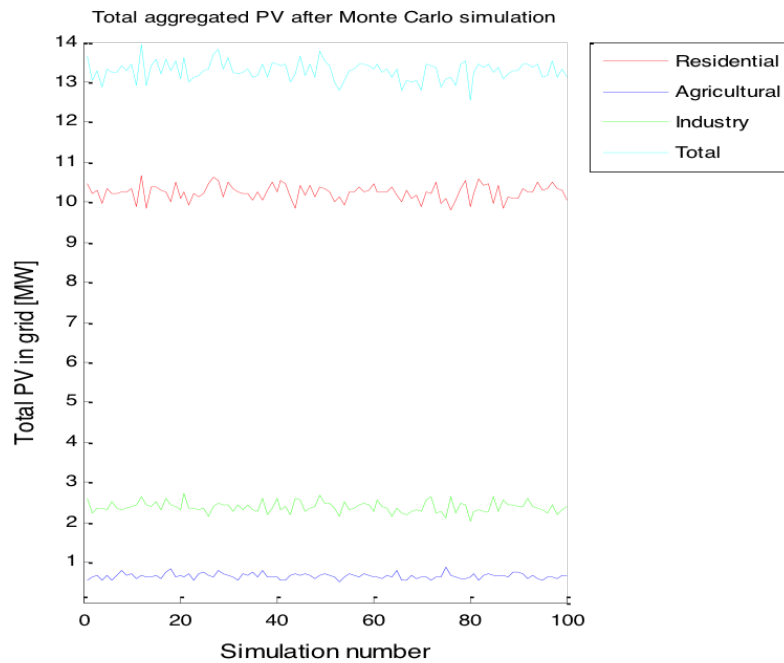
Figure 6.3: Aggregated PV per simulation

6.2.3 Required Number of Simulation Runs

In the previous section, it was shown that running the Monte Carlo model with the random numbers of figure 6.2 will give values around the forecasted photovoltaic installation in GaP in 2030. But how many simulations are needed in order to obtain a satisfying accuracy in the result? There are nearly 8000 buildings in GaP and each is assigned a random number. Running a loop through all the houses exactly one time equals one iteration and gives exactly one result of the PV distribution. The explanation of the statistical terms *sample* and *population* was given in section 3.1. The number of iterations in the Monte Carlo simulation are a collection of data, and serve as our *sample*. The real

mean value of the population, from where our sample origin, is denoted μ_x and in our case, μ_x is the forecast which we try to reach. From our sample we can calculate the sample mean (called *the average*) and denoted $\bar{X}$. The higher the number of iterations, the closer $\bar{X}$ will be to the real mean value. However, the amount of data in our case is very large, and the number of iterations that can be performed is limited due to time and computational power constraints. In the following paragraphs, the dependency between some statistical parameters and the number of iterations will be investigated. To illustrate this dependency, a loop with 200 Monte Carlo simulations is performed. Each of the 200 simulations has a different number of iterations, increasing from 1 to 200.

Mean

The mean of a Monte Carlo simulation sample comprising of n iterations can be calculated by equation 6.5 from section 3.1.1:

$$\bar{x} = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i \qquad (6.5)$$

In figures 6.4 - 6.6, $\bar{x}$ is plotted versus the number of iterations per Monte Carlo simulation. This must not be confused with the results in figure 6.3, where the total PV in GaP is plotted for every time the model has run through all the 8000 houses exactly one time. The average of all the iterations in figure 6.3 would give exactly one data point (the data point corresponding to 100 on the x-axis) in figures 6.4 - 6.6. It can be seen, that the mean gets closer to the real mean from table 6.2 as the number of iterations increases.
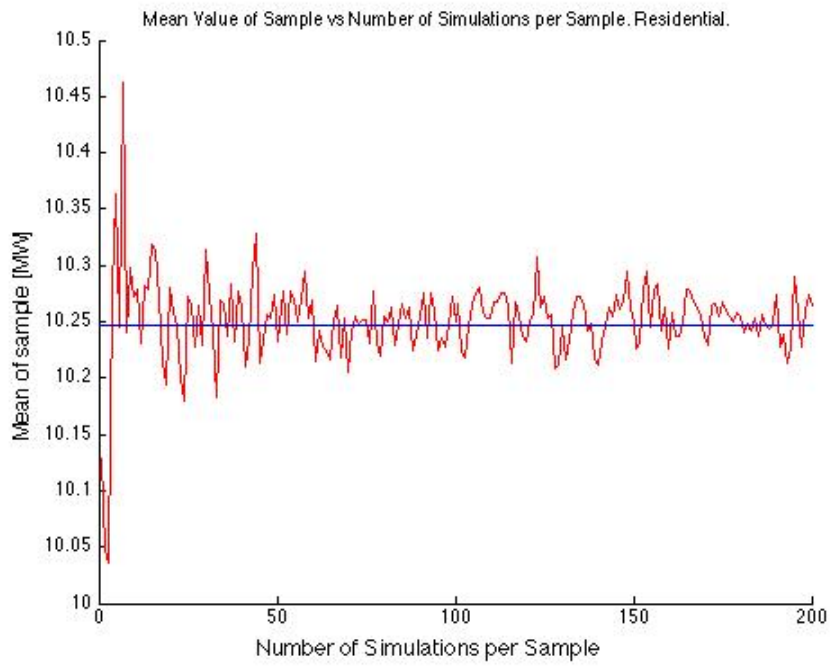
Figure 6.4: $\bar{x}$ versus simulation number
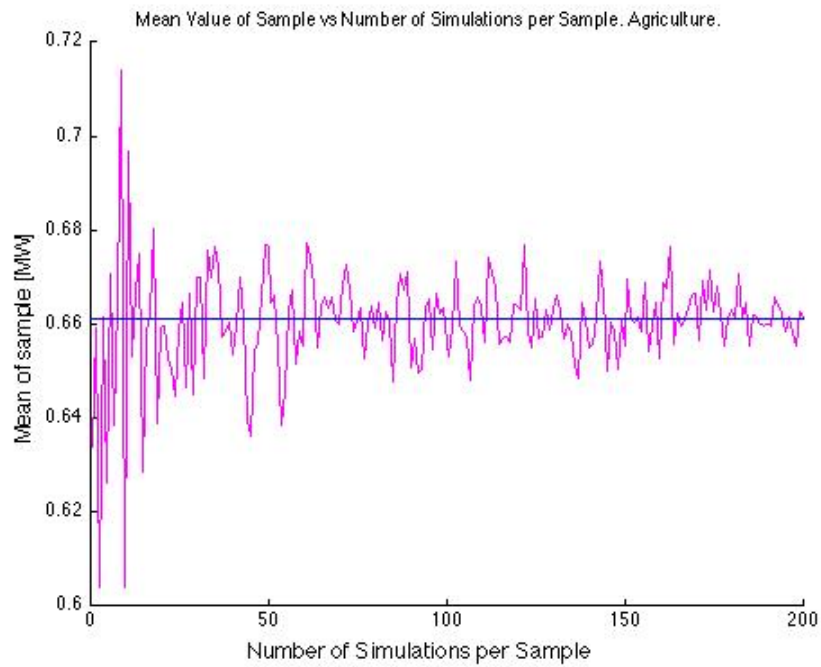
Figure 6.5: $\bar{x}$ versus simulation number
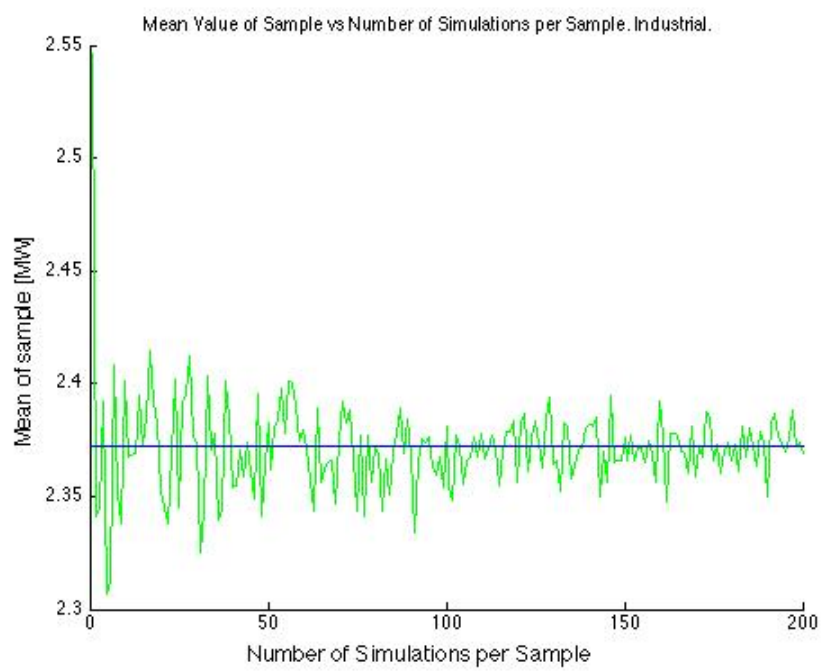
Figure 6.6: $\bar{x}$ versus simulation number

Standard Deviation

From our sample we can calculate the sample standard deviation S_x by equation 3.2 from section 3.1.2

$$S_x = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2} \qquad (6.6)$$

We do not know the real standard deviation for our population (σ_x). But according to [25], $\sigma_x \approx S_x$ when running a large number of iterations. A plot of the sample standard deviations versus the number of iterations (figure 6.7), shows that S_x is quite stable as the number of iterations increase, thus the value for our sample standard deviation is taken from a simulation with a large number of iterations.
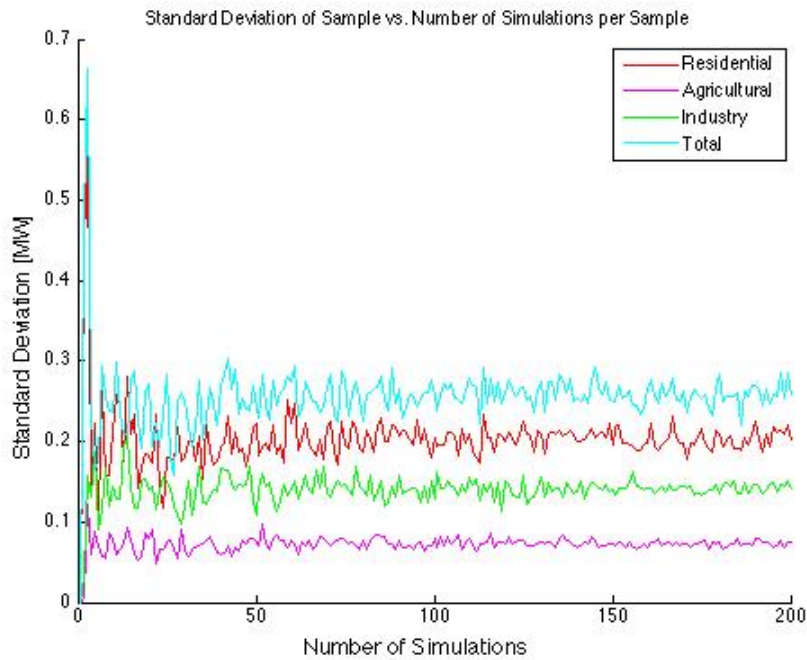
Figure 6.7: S_x versus simulation number

Confidence Interval

If the Monte Carlo simulation is performed a high number of times times, with a sample size n, it can be shown that the expected mean of $\bar{x}$ is the real mean μ_x and that the expected standard deviation of $\bar{x}$ (if large sample size) is:

$$\sigma_{\bar{x}} = \frac{\sigma_x}{\sqrt{n}} \qquad (6.7)$$

We can use a *confidence interval* to indicate the reliability of this estimate. An upper (U) and lower (L) limit of the deviation has to be defined. The range between U and L represent the interval where we expect the value of $\bar{x}$ to be. The interval can be written as:

$$L, U = \mu_{\bar{x}} \pm z_c * \sigma_{\bar{x}} = \mu_{\bar{x}} \pm z_c * \frac{S_x}{\sqrt{n}} \qquad (6.8)$$

where z_c is the confidence coefficient, and it's value is given in statistical tables (e.g. [24]) according to the confidence level. Since the population variance is unknown, the approximation $\sigma_x \approx S_x$ explained in the previous section is applied in the equation.

The confidence level has to be chosen in respect to the maximum error that the model is allowed to have. Figure 6.8 gives an idea on the percentage error that this Monte Carlo model inhibits. As seen, the error shrinks when the number of simulation per sample rises.
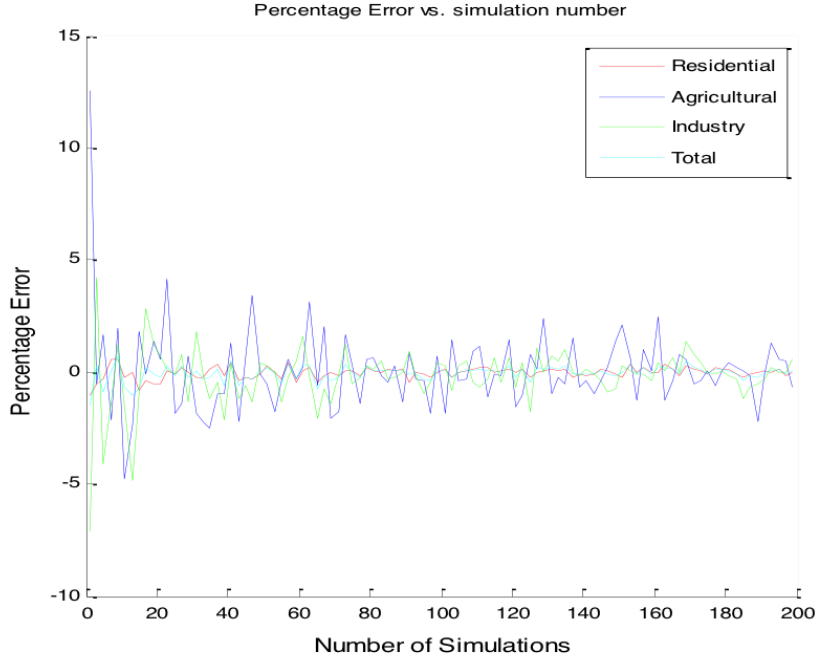
Figure 6.8: Percentage error versus simulation number

To find a reasonable maximum error per sector when approaching a large number of simulations, the largest simulations must be investigated. Figure 6.9, is a close up of the error simulations from figure 6.8.

It can be seen that the residential sector has an overall lower percentage error than the two other sectors. As explained in section 6.2.2 by the coin-tossing example, the less percentage error in residential sector is due to a much higher number of buildings in this sector than in the other two sectors and thus the expected value of the random number is closer to the real probability
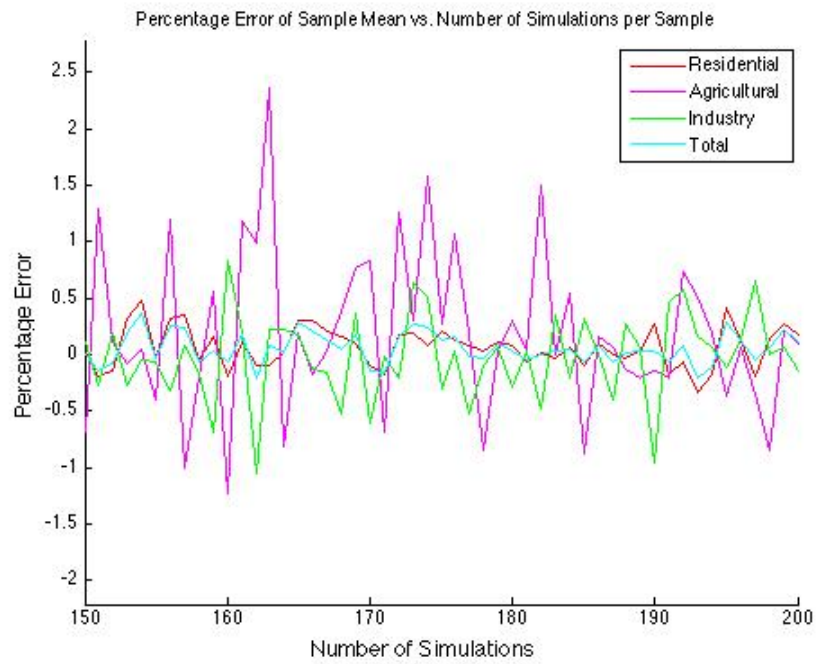
Figure 6.9: Close up of percentage error versus simulation number

distribution. The closer the generated mean value is to the real mean value, the less the percentage error.

Like the confidence interval, the error can occur on both sides of the mean value. Thus, it is possible to set the confidence interval to represent twice the error.

$$Error_{max} = z_c * \frac{S_{\bar{x}}}{\sqrt{n}} \tag{6.9}$$

53

Estimating required number of simulations

A study on the required number of iterations for Monte Carlo simulations are presented in [25]. In the study, equation 6.9 is rewritten, so that the acceptable error is expressed as the percentage error of the mean. This gives equation 6.10.

$$Error_{max}[\%] = 100 * z_c * \frac{S_{\bar{x}}}{\bar{x} * \sqrt{n}} \qquad (6.10)$$

Solving this equation with respect to n, an expression for the number of required iterations can be found, based on the choice of percentage allowable error, the standard deviation, the mean and the confidence coefficient:

$$n = \left[\frac{100 * z_c * S_{\bar{x}}}{\bar{x} * Error_{max}[\%]} \right]^2 \qquad (6.11)$$

Equation 6.11 will now be applied to the three sectors of GaP, in order to find the number of required simulations. The input values for the equation is listed in table 6.3. To verify how the numbers are found, an explanation of sector residential follows:

An reasonable error value for residential sector would be, according to figure 6.9, set at around 0.5%. This means a confidence level of $100 - 0.5 = 99.5$. Using a statistics table (e.g. [24]), the z_c is found for this confidence interval. The standard deviation is found by running a Monte Carlo simulation with a large number of iterations, in order for the approximation $\sigma_x \approx S_x$ to be applicable, as explained in section 6.2.3. The $\bar{x}$ is the real mean value, i.e. the forecast for year 2030.

Table 6.3: Input values for equation 6.11

Sector	$E_{max}[\%]$	$ConfidenceLevel$	z_c	$S_{\bar{x}}[MW]$	$\bar{x}[MW]$
Residential	0.5	99.5	2.807	0.2087	10.2465
Agricultural	1.5	98.5	2.451	0.0713	0.661
Industrial	1	99	2.576	0.1297	2.3726

Applying the numbers of table 6.3 to equation 6.11, gives the required iterations for the Monte Carlo model in this thesis:

$$n_{res} = 130$$
$$n_{agr} = 310$$
$$n_{ind} = 199$$

As the same number of iterations must yield for all sectors, an iteration number that is higher than the n_{ind} will be applied. The Monte Carlo Model is run with 1000 iterations and the result saved as excel files, ready to be used for power flow simulations.

It should be mentioned that to have a power flow simulation result that covers the entire voltage range between possible extreme voltage drops and

voltage rises, the number of necessary power flow simulations must also be calculated. In this case the mean value would be the voltage value, and most probably the result would be that a very large number of simulations is required. Due to time limitation, this was not performed in the thesis.

6.3 Results for Photovoltaic Studies

6.3.1 Monte Carlo Simulation Results

Figure 6.10 depicts a histogram of the total PV per simulation in the low voltage grid of GaP. As seen, the occurrence is high around the forecasted value of 13.28 MW. All the simulations end up with a total PV capacity between 12.4 MW and 14.2 MW
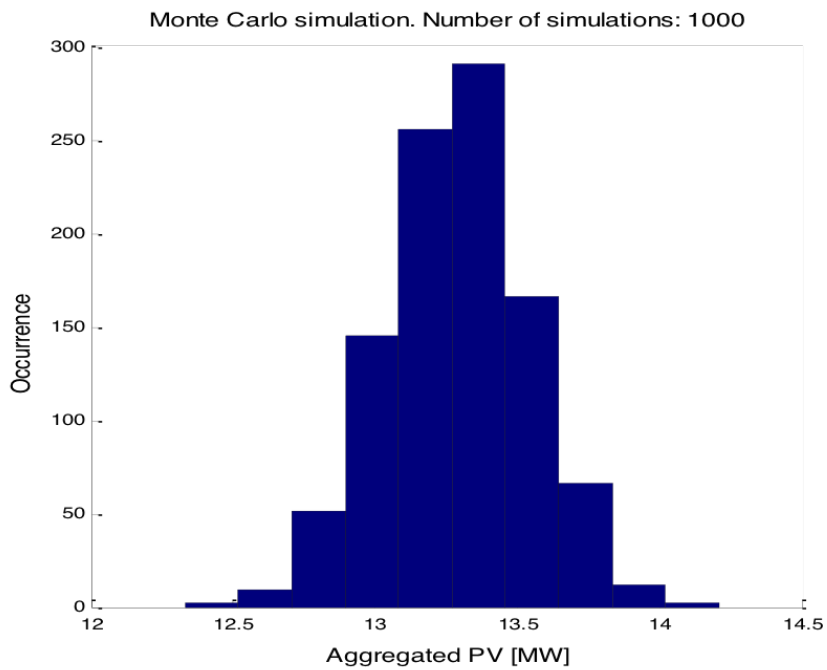
Figure 6.10: Histogram of total PV

The variation from simulation to simulation in the Monte Carlo model lies not only in the total PV in the grid, but of course also in the spatial distribution of the PV plants. This will be thoroughly investigated by the power flow analysis. Figure 6.11 shows the average aggregated PV of all the simulations per transformer station and can give an idea of which transformer that will get high penetration of PV.
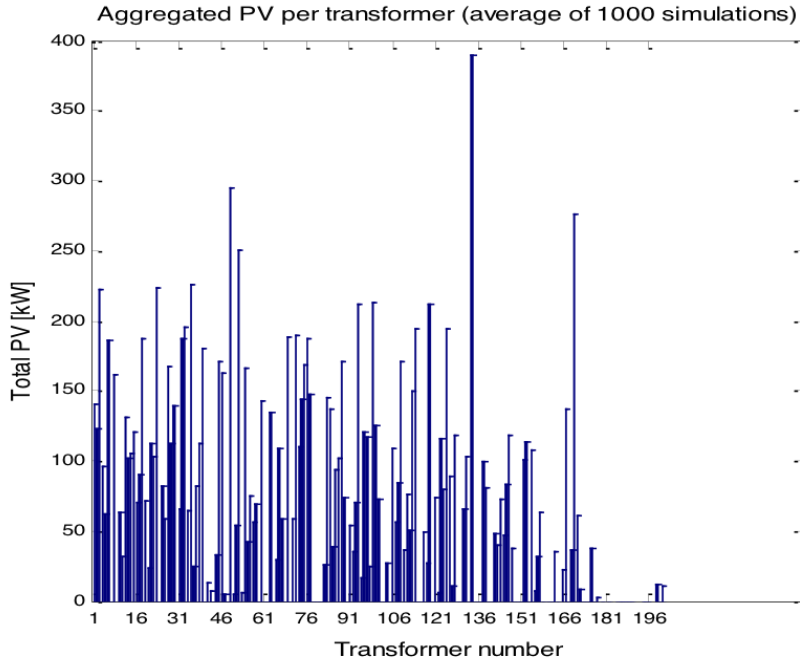
Figure 6.11: Average aggregated PV per transformer station

6.3.2 Power Flow Simulation Results

Winter Evening Scenario: High Load - Zero Generation

This scenario is based on the load for GaP on the situation a January evening, which was calculated in section 5.3.1. As opposed to the summer day scenario, where a large amount of power flow simulations must performed to see the different effects among the Monte Carlo spatial distributions, only one single power flow is necessary to see the status of this scenario. No production from PV is present in the grid and the loads are modelled after the standardized load profile for the 15 minute interval beginning at 19.00. The single power flow is runs and the resulting bus voltages are displayed as a box plot in figure 6.12. The squared box represents the central 50% of the bus voltages. The green line indicates the median of the bus voltages. The two blue lines which extend vertically from the square, indicate the remaining bus voltages outside the central 50%, but extend maximally to $\frac{3}{2}$ times the height of the squared box. All bus voltages larger or smaller than $\frac{3}{2}$ times the central 50% are considered *outliers* and these are marked with red crosses.

The slack bus voltage is set to 1.0 pu. The left data set depicts the situation in the 400 V network. The majority of these buses have voltages between 0.992 and 0.982 pu, which means a voltage drop between 0.8% and 1.8%. The outliers have voltages down to 0.975 pu (2.5% voltage drop). The outliers are buses, which after closer investigations, turns out to be almost exclusively located on transformer 85. This transformer has a rating of 250 kVA and supplies 64 loads,

where 62 of them are residential and 2 agricultural. The 2 agricultural loads are located in the end of a long line, which again are connected to a radial feeder supplying residential buildings. All the most extreme outliers are registered at this feeder, with the largest voltage drop in the very end of the feeder.

The right data set depicts the situation in the 10 kV network. The majority of the buses has voltages between 0.99 and 0.9975 pu, which means a voltage drop between 0.25% and 1%. The outliers have voltages down to 0.97 pu (3% voltage drop). These outliers are found to be buses located mainly far south in the medium voltage grid, on long lines supplying lifts and railways in the GaP skiing area.
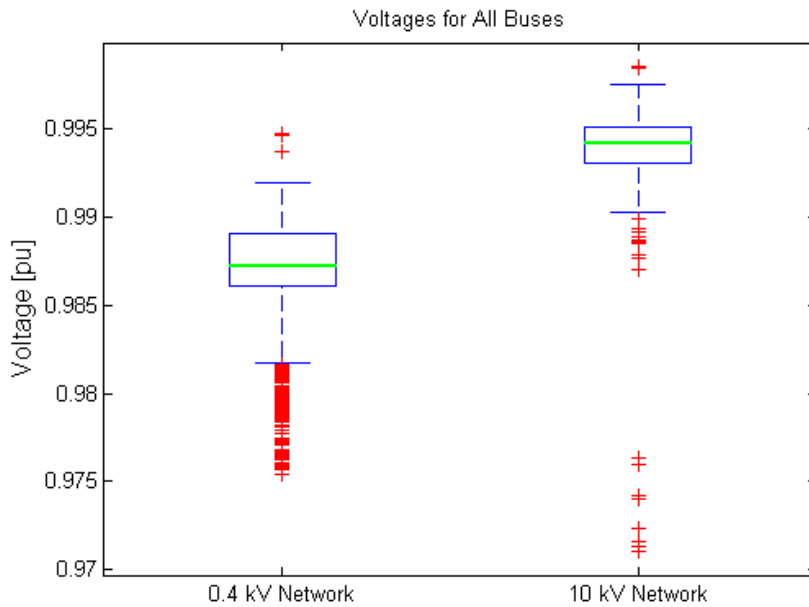
Figure 6.12: Voltages for Winter Evening Scenario

The result of the winter evening is in compliance with the theoretical foundations in section 2.2: the voltage drop increases with the line length because the line impedance Z increases with the line length. Thus the lowest voltages are in the end of long feeders. However, it can be concluded that for this scenario, all voltages are inside the permissible limits.

Summer Day Scenario: Low Load - Max Generation

Basis for the power flow simulations on the summer day scenario is the Monte Carlo simulation results where the expected PVs are distributed among the houses in GaP. These simulations results represent a large amount of different spatial distributions. It exist an infinite number of distributions, but the power flow simulations are time demanding. For this reason, a limit of 1000 has been set, that is, a power flow snapshot simulation of the grid is performed for each

of the 1000 different Monte Carlo results. This takes approximately 40 hours. Graphical illustration of the 1000 power flow results for the summer day scenario must be done in a slightly different manner than the single power flow result from the winter evening scenario.

Voltages As explained in section 2.3, distributed generating units like rooftop PV plants can cause overvoltages if the power production from the PV exceeds the consumption of the customer. As this situation inhibits low load and max generation, overvoltages are expected. The 10 kV network consists of 338 buses. In the winter evening scenario, all of these were below nominal voltage. In the summer day scenario, on the other hand, the situation is different. The histogram in figure 6.13 shows the occurrence of 10 kV buses with voltages above nominal voltage. From this histogram it can be seen, that in 800 of 1000 simulations, 182 of the 338 10 kV buses has a voltage above nominal voltage.
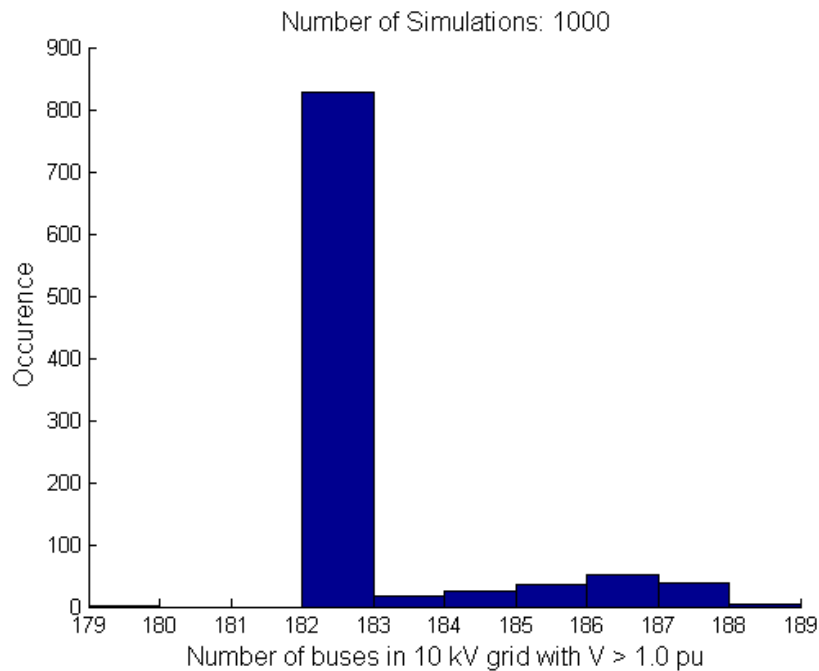
Figure 6.13: Voltages in 10 kV Network for Summer Day Scenario

The actual value of the buses are of great interest, thus a plot of the bus voltage magnitude from the histogram are presented in figure 6.14. As seen, some of the buses get a voltage close to 1% above nominal voltage. This is inside the permissible voltage limits for medium voltage grids in Germany. Two large PV plants are connected to the 10 kV network through transformers. They have a rated power of 1212.2 kW and 544.32 kW. The highest voltages occur at the connection point of these two transformers and influence parts of the grid, creating high voltages on several buses.
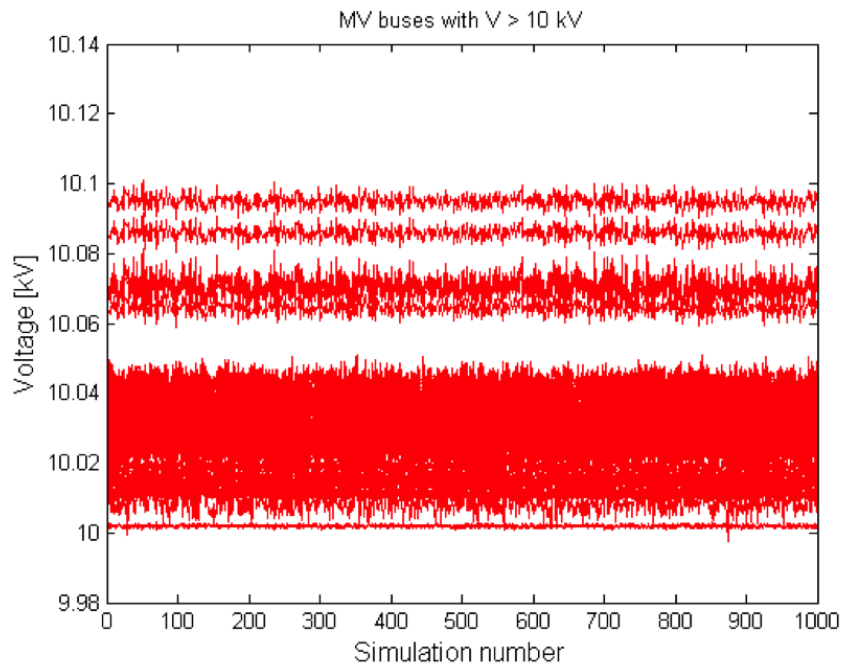
Figure 6.14: Voltages in 10 kV Network for Summer Day Scenario

The 16 low voltage parts of the network are made up of together 2487 buses. In the winter evening scenario, all of these were below nominal voltage. On a summer day the situation is quite different. The histogram of figure 6.15 shows the occurrence of 0.4 kV buses with voltages above nominal voltage for the summer day scenario. In around 450 of the 1000 simulations, every single of the 2487 low voltage buses are above nominal voltage. There is never less than 2150 buses with voltage above nominal voltage. This means that for 100% of the 1000 different spatial distributions, minimum $\frac{2150}{2487} * 100\% = 86\%$ of the buses have a voltage above nominal voltage. The voltage magnitude of these buses are plotted in figure 6.16. A reference line for the 3% voltage rise limit set by the VDE directive mentioned in section 2 is also included in the figure, and as seen the limit is equivalent to a voltage of 412 V. Since the LV grids consists of 2487 buses, it is not easy to graphically illustrate all the voltages. However, it can be seen from the graph 6.16, that the +3% limit is violated in several of the simulations. By closer analysis, it was clear that exactly 37 buses violated the limit.
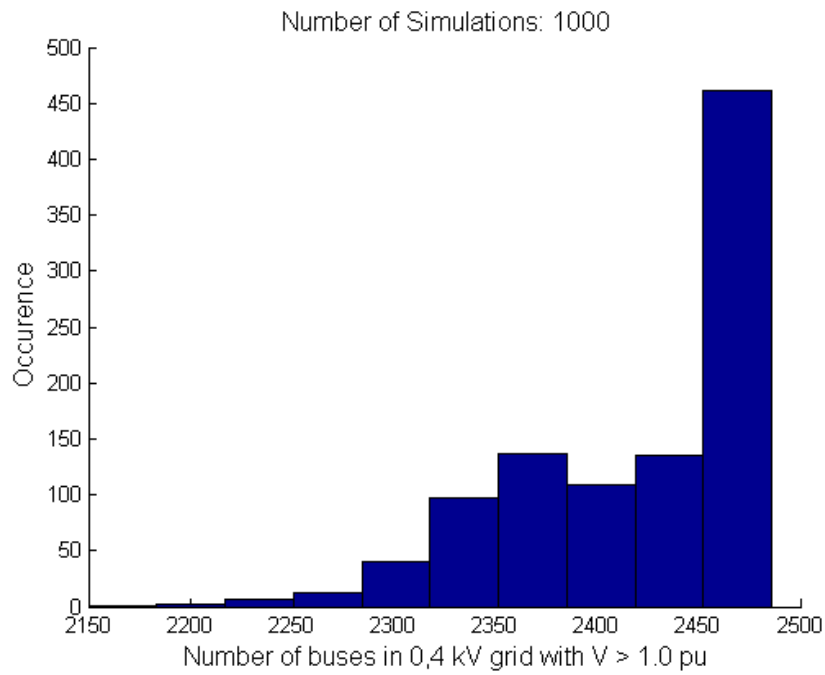
Figure 6.15: Voltages in 0.4 kV Network for Summer Day Scenario
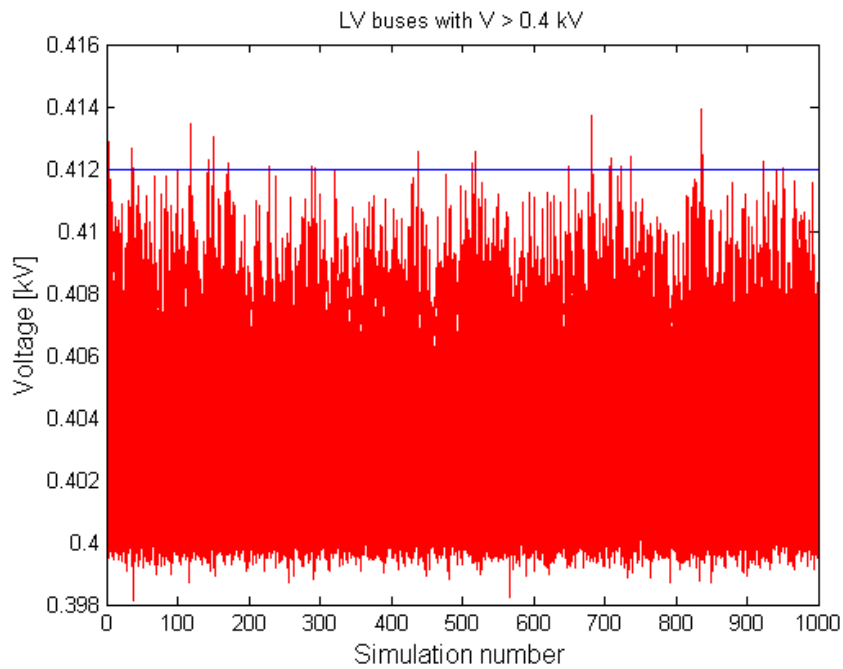
Figure 6.16: Voltages in 0.4 kV Network for Summer Day Scenario
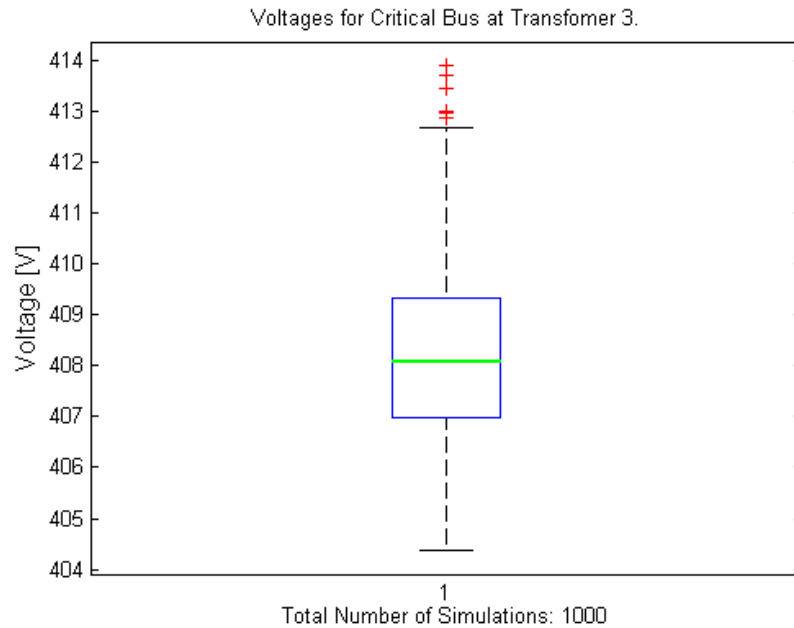
Figure 6.17: Voltages for critical bus in Summer Day Scenario

Of the 37 overvoltages occurring in the low voltage grids, 4 instances of a voltage above 413 V was recorded. These over voltages were all occurring at the same bus. A box plot of this bus is shown in 6.17. The top outlier represent the highest bus voltage of 1000 Monte Carlo simulations for this building. This outlier has a value of 413.9 V which means a voltage rise of 3.45 % above nominal voltage. The green line is the median and it has a value of 408 V. In addition, it can be seen that 100% of the 1000 simulations result in a voltage above 404.4 V.

The bus is the connection point of a residential house connected to transformer 3. The transformer supplies an urban area that is densely populated and the building with the highest voltage is one of the buildings that is located furthest away from the transformer.

Overloading An important part of power flow analysis is to check whether the grid infrastructure can physically accommodate such large power flows that is the consequence of 13.28 MW of grid connected photovoltaics. Two aspects regarding overloading will be examined here. First, the power at all the 10 kV/0.4 kV transformers are measured and compared to their rated power. Second, currents of all line segments will be measured and compared to the cable type's thermal limit.

The apparent power at all transformers were measured for the 1000 Monte Carlo distributions, and by comparison to the transformer rating, it was shown that never was any of the transformers overloaded.

In the line definition of the (*.dss)-files, the cable types of each line are in-

cluded. In the cable type specification a value called "NormAmps" lies, which is the thermal limit of the respective cable type. After the power flow simulations, it is possible to export a list of all lines, with the thermal limit of the cable type and the current each line segment is actually carrying. Thus, it is possible to check whether any of the lines are overloaded with regard to the cable type's thermal limit. The results showed that for the 1000 Monte Carlo distributions, zero line segments were overloaded in regard to their thermal limit.

6.4 Worst Case Scenario

6.4.1 Concept

The Monte Carlo results represent a certain amount of distributions generated by random numbers. These are most-likely scenarios and can give a picture of the influence that the PV could cause. However, with almost 8000 buildings, there exists an infinite larger number of combinations for the PV plants distribution. The 1000 spatial distributions from the Monte Carlo simulation will only cover a tiny fraction of the total number of different distributions possible.

The probability that a worst-case scenario exist in any of the 1000 distributions from the Monte Carlo simulation model is therefore small. For this reason, a spatial distribution representing a worst case scenario is manually constructed.

Considering voltage rise, the worst case arises from the combination of a high line impedance and a large amount of PV power on this line. Therefore, to find the worst case scenario, the low voltage distribution grids must be investigated, looking for long lines with large rooftop areas.

Transformer 23 has a mix of residential and industrial buildings, and supplies in total 67 loads. It has a rating of 630 kVA. The lines from this transformer are not particular long, but have some very large buildings connected and thus a large theoretical rooftop PV potential. Figure 6.11 from section 6.3.1 depicted the average PV per transformer for 1000 Monte Carlo simulations. From this it was seen that transformer 23 on average was assigned 224 kW. However, a plot of the PV per simulation for this particular transformer (figure 6.18), shows that the PV is ranging from 75 kW to 350 kW.
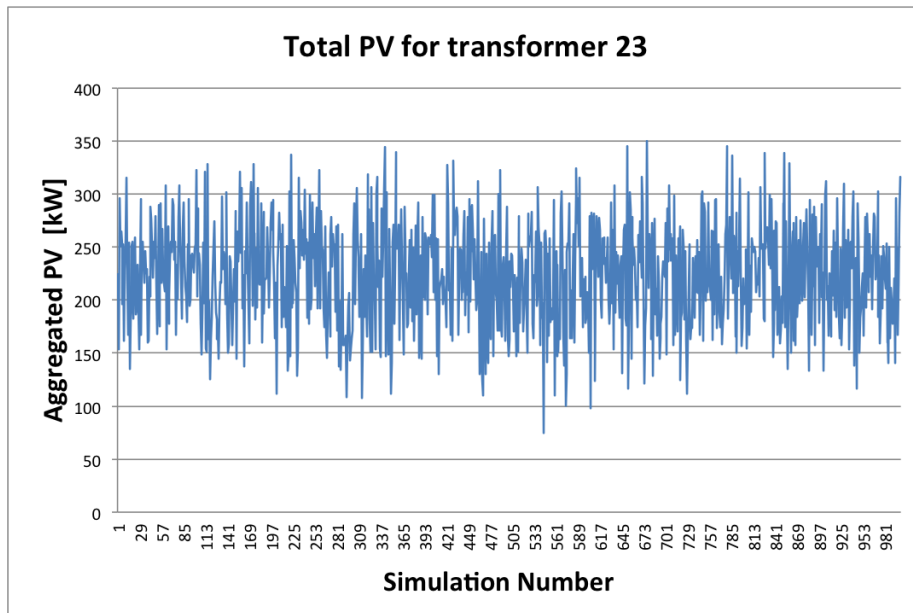
Figure 6.18: Aggregated PV per simulation

Among the lines from transformer 23 is a radial feeder which supplies eight

industrial buildings, some with very large rooftop areas. The eight buildings are from here on named A-H and their PV potential are listed in table 6.4. A line diagram of the feeder is shown in figure 6.19. The small circles on the feeder represent fuse boxes where, in reality, several cables are connected.
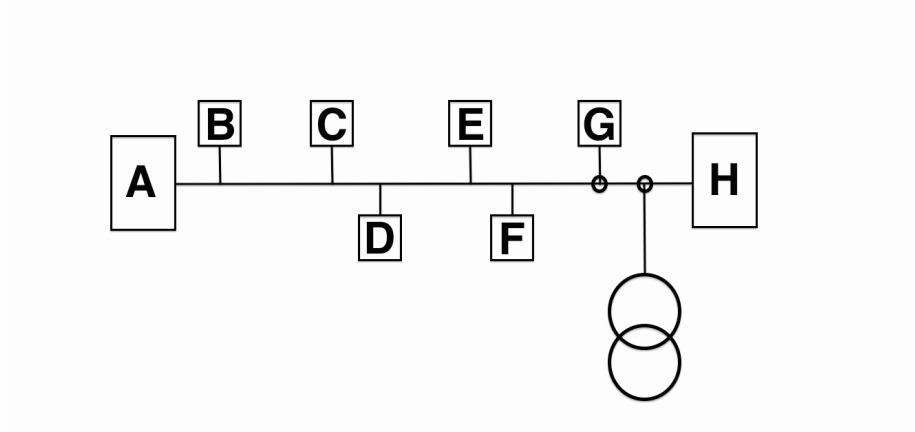
Figure 6.19: Feeder at transformer 23

Table 6.4: PV Potential for buildings A-H. Numbers from [23].

Building	PV potential [kW]
A	128,31
B	57,99
C	59,78
D	57,02
E	57,00
F	12,41
G	14,16
H	118,67

A simulation model to perform the worst case scenario is created in MatLab. It assigns rooftop PV plants to the houses on the critical feeder one at a time. For every house a power flow is run, until all the 8 houses on the feeder have PV plants installed. Since the scenario is representing a worst case, the entire roof is filled with solar panels. That is, a generator is added to the bus where the house is connected, with the power from table 6.4. The power flow is performed with the same assumptions as the summer day scenario that was presented in section 5.3.1, that is, max production from the PV plants and minimum consumption of the loads.

6.4.2 Simulation Results for Critical Feeder

Referring to the line diagram in figure 6.19, the PV plants are added to the houses one at a time, starting in the end of the feeder at building A, ending

with building H. Figure 6.20 is a plot of the total loading of the transformer, measured on the secondary side of the transformer after every power flow. As seen, before any PV is added to the grid, the active power has a negative value of 31,7 kW, which means the loads are consuming power. At the instant the large PV plant is added at building A, the power turns positive, which means that the power flow has changed direction and is flowing from the feeder to the transformer. The reactive power in the grid is present due to the power factor of the loads (pf = 0.95). Through equation for Q_{load} the value for the situation before any PV is added can be verified:

$$Q_{load} = P_{load} * \tan(\arccos(pf)) = -31,7kW * \tan(\arccos(0.95)) = -10.42kVAr$$

This value corresponds to the measured reactive power value represented by the blue line plotted in graph 6.20.

Figure 6.21 is a plot of the voltage at the connection points for the 8 buildings on the critical feeder. Before any PV is added to the grid, the voltage is between 396 and 397 Volt for all the loads on the feeder. When the voltage is below the nominal voltage, it means that there is a voltage drop from the transformer to the loads. When the PV plant is added at A, the voltage rises above nominal voltage at the connection point for this building. Also the other buildings on the feeder are affected by the PV plant at A. It can be seen, that the further away from the transformer, the higher the voltage rise becomes.
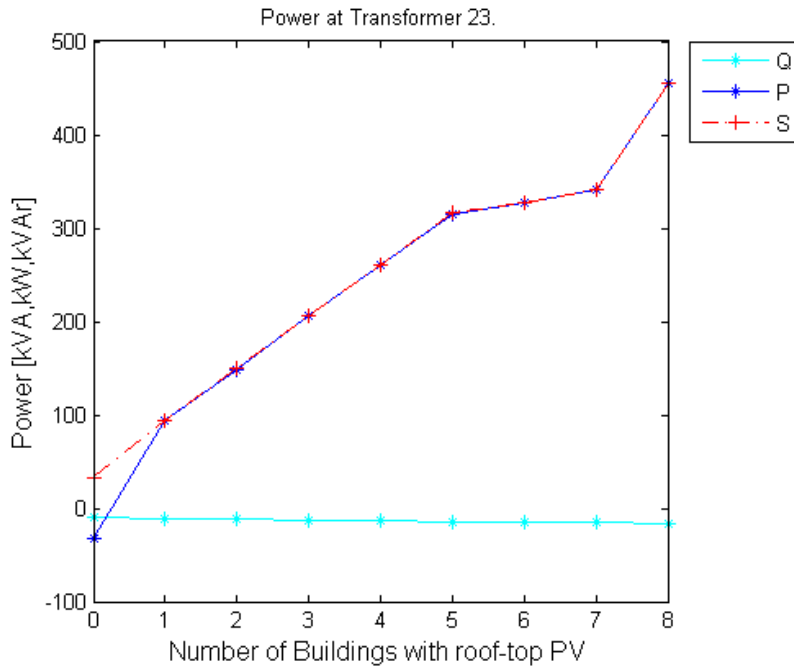
Figure 6.20: Power for transformer 23

As mentioned, the Monte Carlo model assigned a maximum of 350 kW to transformer 23. This amount is reached when buildings A-E has filled their
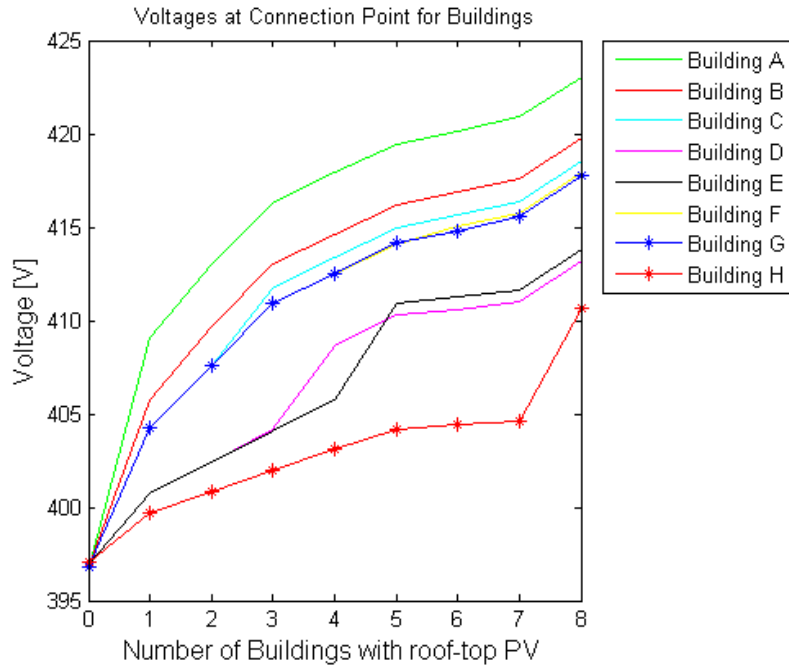
Figure 6.21: Voltage for critical feeder on transformer 23

roofs with solar panels. The voltage rise at this point is 419.5 V, which is 4.9% above nominal voltage and it already represents a violation of the 3% limit. The probability that this exact spatial distribution was included in the Monte Carlo results is extremely small. It could however happen in reality and when looking at the voltage rise that this exact distribution can cause, it shows the importance of defining and investigating such special-case scenarios.

However, the Monte Carlo results are not definitive. It is not known how much PV that will end up for the different transformers. Continuing to install PV plants for buildings F,G and H result in a total of 505 kW PV power. When subtracting the load, the total power is 455 kW, as shown in figure 6.20. It leads to a violation of the 3% limit for all buildings, except H, which is located close to the transformer.

The underlying grid of transformer 23 consists of 182 line segments. Measurements of currents of all line segments were compared to their cable type's thermal limit. From this it was seen, that when all the houses on the critical feeder have installed PV plants, two lines are carrying a current larger than their thermal limit. Location of the overloaded line segments is a complex operation, as they have to be identified among hundreds of other lines, where the only information is which buses they are connected between. However, in this situation, where only one feeder is assigned PV plants, it can be assumed that the overloaded line segments are somewhere on this feeder, thus the search for the overloaded line segments is easier. Closer investigations confirm the assumptions. Two aluminium line segments (NAYY 4x120) between the two fuse

boxes are overloaded. This cable type has a thermal limit of 245 A, but the two line segments carry 303 A and 367 A, respectively. These cables are overloaded because they must carry large amount of PV power. In addition, the copper cable (NYYx50) between the large building H and the right fuse box is close to its thermal limit. The limit of this cable type is 188 A, and the measured current is 166 A. This can be verified by a simple calculation: From table 6.4, it can be seen that building H has a max power production of 118.67 kW. The load at 14:00 is 0.555 kW, as calculated in section 5.3.1.

From this, the current in this line segment can be approximately calculated using equation 2.3 from section 2.2:

$$Ic = \frac{P}{Uc} = \frac{118.67kW - 0.555kW}{0.4kV * \sqrt{3}} = 170A$$

Overloading of Transformer

In addition to the eight buildings on the critical feeder in figure 6.19, there are 59 more loads connected to the transformer. Next, the simulation model will assign PV plants to rooftops until the rated power of the transformer is exceeded. The power plot is shown in figure 6.22 and the dark blue reference line shows the rating of the transformer. When 19 buildings are assigned their full PV potential, it can be seen from the plot in 6.22 that the transformer rating is exceeded. The voltage development for these 19 buildings are plotted in figure 6.23. The highest voltage rise is still at building A, which is located in the end of the feeder. Even though the new PV plants are added at other places in the underlying grid of transformer 23, the critical feeder is influenced and the voltage rise gets even higher at all the buildings at this feeder. Building A now has a voltage of 427,8 V, which is 7% above nominal voltage and a clear violation of the limit stated by the VDE directive. It was seen in figure 2.1 that the medium voltage grid can have a voltage above nominal voltage, which means that there is already a voltage rise present before these 7%. If that is the case, it would mean that this worst-case scenario is close to violating the 10%-limit from EN 50160. In addition, the simulation results show that five line segments are carrying a current larger than their thermal limit.
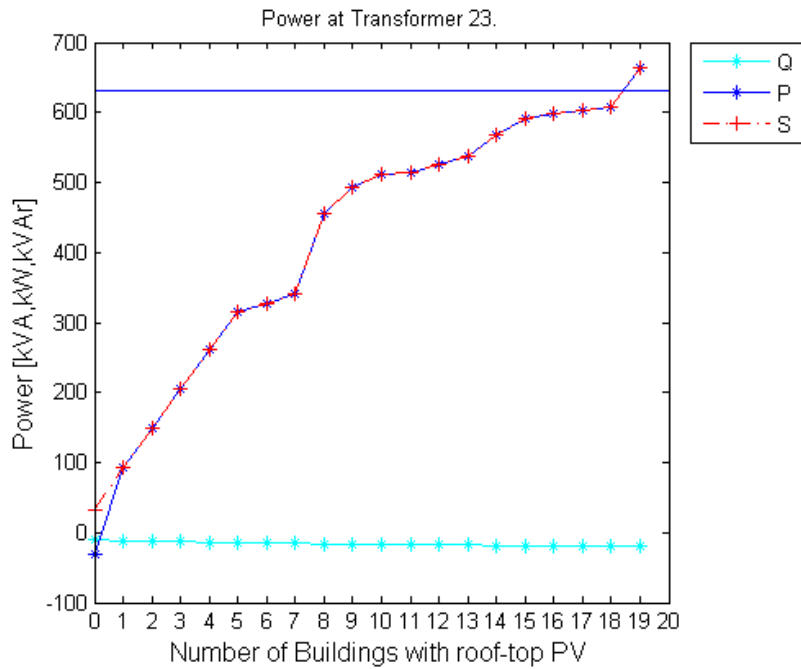
Figure 6.22: Power for transformer 23
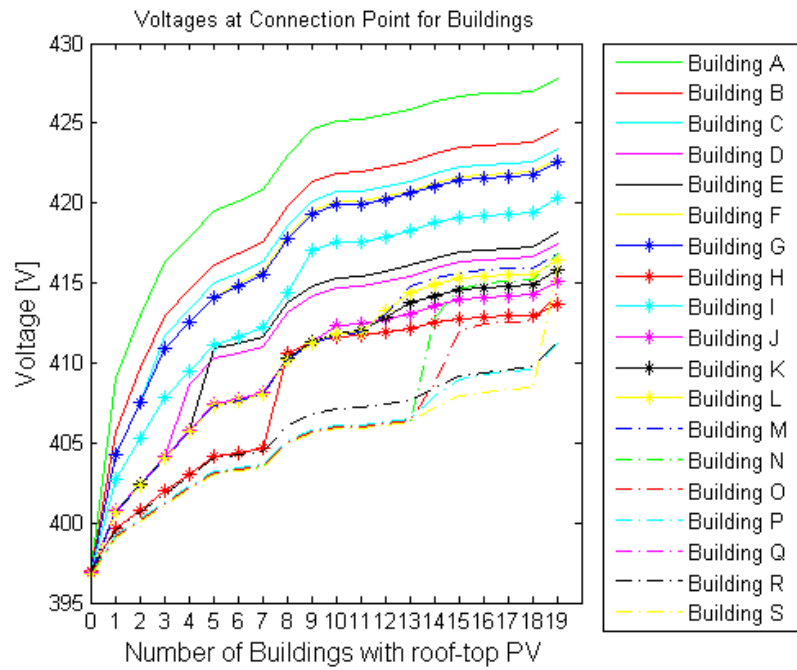
Figure 6.23: Load voltages at transformer 23

Chapter 7

Studies on Electro Mobility

7.1 Future Scenario of Electromobility Development in GaP

GaP is subject to research also in the field of electro mobility. A forecast of the number of electric vehicles in the city is however not available. Some studies regarding future e-mobility situation in GaP is therefore carried out here.

According to the National Platform for Electric Mobility in Germany, 6 million electric vehicles can be expected in Germany by 2030 [26].

In a study published by the *Rheinisch-Westfälischen Instituts für Wirtschaftsforschung* ([27]) on population decline and car ownership in Germany, some useful forecasts are presented: The number of inhabitants decline, but the size of the households (i.e.: the number of people per house) is decreasing. That means the number of households keeps almost constant and in 2030 the number of households in Germany is expected to be around 41 millions, according to this study.

If the national goal of 6 million electric vehicles in Germany by 2030 is reached and the forecast of households in [27] is correct, the percentage of households with electric vehicles can be calculated to be 14.6%, see equation 7.1.

$$\frac{ElectricVehicles}{Households} * 100\% = \frac{6000000}{41000000} * 100\% = 14.6\% \tag{7.1}$$

Assuming the same number of households in GaP in 2030, the number of electric vehicles can be found in equation 7.2.

$$ElectricVehicles = \frac{Households}{100\%} * 14.6\% = \frac{7992 households}{100\%} * 14.6\% = 1167 \tag{7.2}$$

7.2 Monte Carlo Simulation Model for Electro Mobility

The Monte Carlo simulation of electric vehicles, will result in a spatial distribution of the future amount of electric vehicles in the grid of GaP, calculated in section 7.1. It is through their charging process that electric vehicles can have an impact on the grid. Picture a feeder, supplying a neighbourhood where all the inhabitants own an EV and charge their vehicle when returning from work in the evening. The resulting power on this feeder could in this situation exceed the maximum load that the grid was designed for. The charging process of an electric car will vary in several ways, through time of charge, elapsed charging time, charging technology and power and place of charging. Thus, it exists countless different charging scenarios. It is therefore necessary to define certain assumptions around the charging process before running the Monte Carlo model.

7.2.1 EV Charging Power

International Electrotechnical Commission (IEC) has developed standards that specify plugs and sockets for the charging of electric vehicles and the effects that these require. The specifications can be found in the standards *IEC 62196-1* and *IEC 62196-2*. The purpose of the standards is to avoid that manufacturers in different regions develop solutions incompatible to other regions [32], in addition to ensure good communication between the grid and the battery electronics for safe charging and to adapt to future smart grid scenario where current and power can be exchanged between electric vehicle batteries and the grid [28].

The two standards are based on the electric vehicle charging system standard *IEC 61851-1: Electric road vehicles and electric industrial trucks* [31]. This standard encompasses four different charging modes:

- Mode 1 (AC): slow charging from a household-type socket-outlet

- Mode 2 (AC): slow charging from a household-type socket-outlet with an in-cable protection device

- Mode 3 (AC): slow or fast charging using a specific EV socket-outlet with control and protection function installed

- Mode 4 (DC): fast charging using an external charger (e.g. charging of a vehicle in under 10 minutes via an off-grid DC charger such as a high voltage battery [31])

In addition to the *Mode*, it is common to specify the *Type* when referring to plugs. IEC 62196-1 contains the general requirements for charging, while IEC 62196-2 standardizes the three types of plugs, known as Types 1, 2 and 3. These are defined in IEC paper [32] to be:

- Type 1: single phase vehicle coupler (vehicle connector and inlet)

- Type 2: single and three phase vehicle coupler and mains plug and socket-outlet without shutters

- Type 3: single and three-phase vehicle coupler and mains plug and socket-outlet with shutters

In Germany, the Type 2 plug from the German manufacturer *Mennekes*, is used. Type 2 is sometimes referred to as simply *the Mennekes plug*. The European Commission has announced, that this plug will be the common standard for charging electrified vehicles across the European Union [33]. The plug enables mode 3 charging from 16 A/single-phase up to 63 A/three-phase [28]. The plug has 7 pins: 3 for charging, 1 neutral, 1 and 2 for communication and an external connector [30]

The amount of charging power from a vehicle using this standard plug can be categorized as follows:

- AC normal charging: 11 kW

- AC semi-fast charging: 22 kW

- AC fast charging: 43 kW

The trend is seen to go towards faster charging [29]. In Germany, a large number of houses are already equipped with three-phase AC semi-fast charge, which enables a 80% charge of 30 minutes with 22-24 kW [30]. For the year 2030 it is therefore reasonable to assume that most cars use either semi- or fast charging technology.

7.2.2   Scenarios for Electro Mobility

Assuming that the numbers for the 2030-situation presented in section 7.1 would yield also for GaP, then approximately 14.6% of the GaP households would have an electric vehicle. For simplicity, it is not differentiated between sectors in this simulation. All buildings are treated as a household and thus, 14.6% is equal to a total of 1167 electric vehicles in total in GaP.

Two electro mobility scenarios will be created by Monte Carlo simulations and studied by power flow simulations. These two scenarios are:

- Semi-Fast Charging Scenario: All electric vehicles use semi-fast charging of 22 kW.

- Fast Charging Scenario: All electric vehicles use fast-charging and charge with the maximum power that the standard plug Type 2 allows, equal to 43 kW.

Except for the new input values, the simulation model will be principally the same as for the PV Monte Carlo model. Each house in GaP is, with a probability of 14.6%, assigned an EV. If the EV is placed to charge at this house, the charging power is either 22 kW or 43 kW.

The simulations are run assuming an evening load to create a worst case scenario. The load value calculated in section 5.3.1 is applied here. The charging vehicles are added to the power flow model as an increase in the customer's load.

7.2.3 Required Number of Simulation Runs

To find the required number of simulations for the EV Monte Carlo model, equation 7.3 from section 6.2.3 can be applied.

$$n = \left[\frac{100 * z_c * S_{\bar{x}}}{\bar{x} * Error_{max}[\%]} \right]^2 \tag{7.3}$$

The $\bar{x}$ is the real mean value, i.e. the expected number of electric vehicles for year 2030, which is 1167. The other input values z_c, $S_{\bar{x}}$ and $Error_{max}$ need to be calculated. This is done in MatLab, the same way as for the PV model, by a loop of 200 Monte Carlo simulations. Each of the 200 simulations have a different number of iterations, increasing from 1 to 200. Each simulation represent a sample, and the sample mean is calculated in every simulation. From this, it is possible to see how the parameters stabilizes around certain values when reaching a large number of iterations. As seen in figure 7.1, the larger the number of iterations per sample, the closer the sample mean gets to the real mean.

According to figure 7.2 the error doesn't exceed 0.5% when the number of iterations gets large. This means it is reasonable to apply a confidence level of $100 - 0.5 = 99.5$. Using a statistics table (e.g. in [24]), the z_c for this confidence level is found to be $z_c = 2.807$. According to figure 7.3, the standard deviation is approximately 31 cars. The values are summarized in table 7.1.
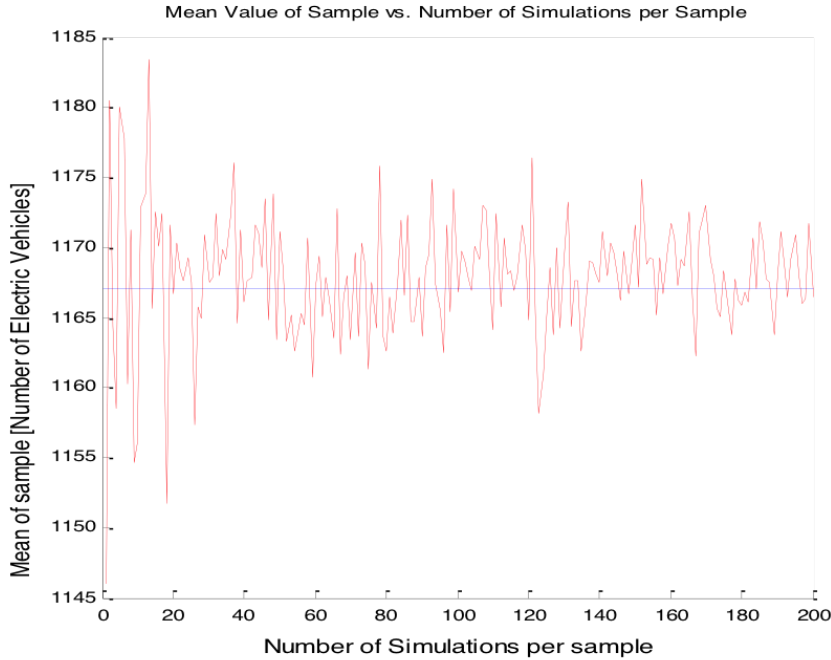
Figure 7.1: $\bar{x}$ versus simulation number

Applying the numbers of table 7.1 to equation 7.4, gives the required iterations for the Monte Carlo model for EV in this thesis:
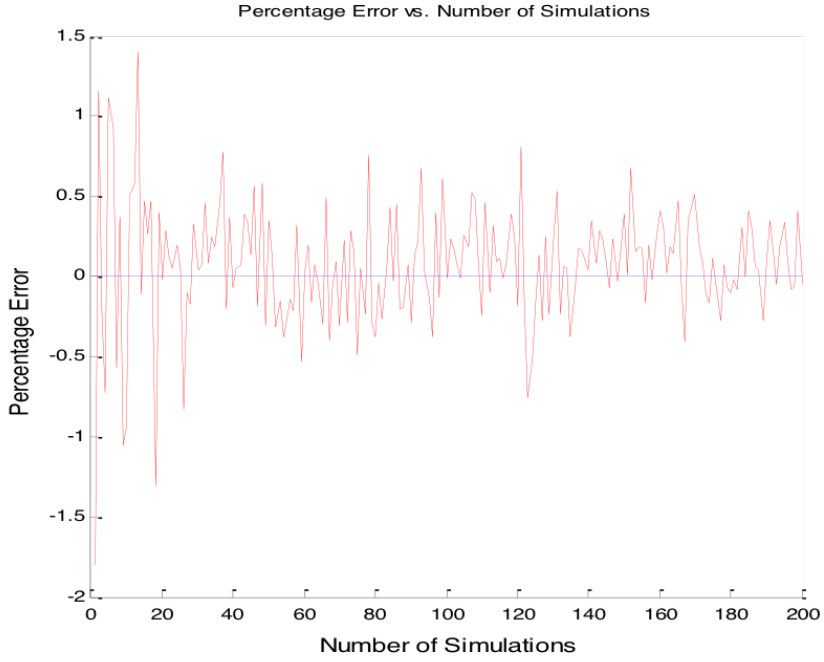
Figure 7.2: Percentage error versus simulation number

Table 7.1: Input values for equation 7.4

EV	$E_{max}[\%]$	$ConfidenceLevel$	z_c	$S_{\bar{x}}[MW]$	$\bar{x}[MW]$
EV	0.5	99.5	2.807	28	1167

$$n = \left[\frac{100\% * z_c * S_{\bar{x}}}{\bar{x} * Error_{max}[\%]} \right]^2 = \left[\frac{100\% * 2.807 * 31}{1167 * 0.5\%} \right]^2 = 222 \qquad (7.4)$$

It can be seen that also for the Monte Carlo e-mobility model, a run of 1000 iterations is more than enough to reach the forecast. The result for the two different scenarios is saved as excel files, ready to be used for power flow simulations.
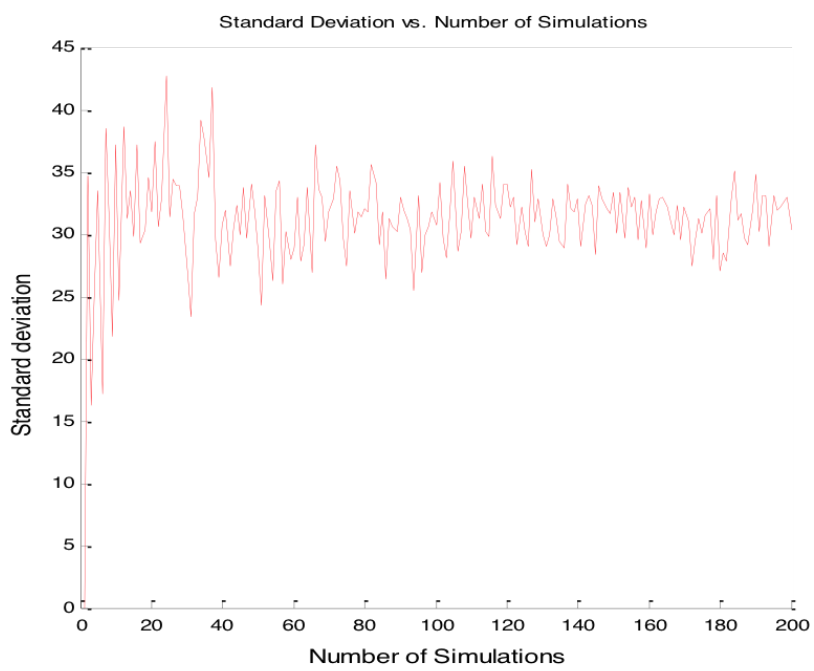
Figure 7.3: S_x versus simulation number

7.3 Results for Electro Mobility Studies

7.3.1 Monte Carlo Simulation Results

For a Monte Carlo simulation with 1000 iterations, the amount of total charging power in the GaP-grid for the two scenarios of 22 kW and 43 kW charging power per vehicle end up as shown in figures 7.4 and 7.5. As for
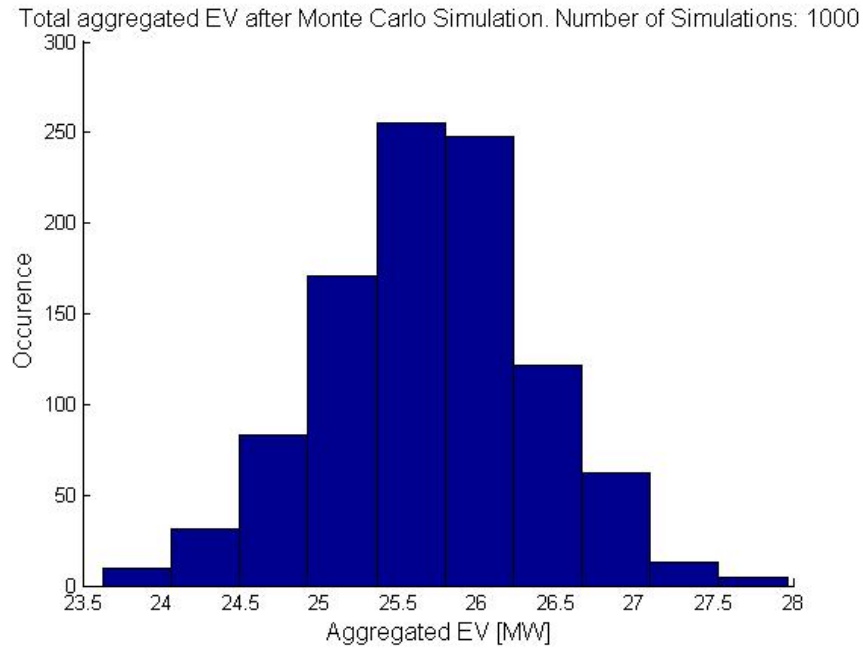
Figure 7.4: Histogram of total EV charging power for semi-fast charging

the case of PV, the spatial distribution of the charging vehicles is the core of interest. Figure 7.6 shows a plot of average power per transformer for the semi-fast scenario and figure 7.7 represent the case for the fast charging scenario.
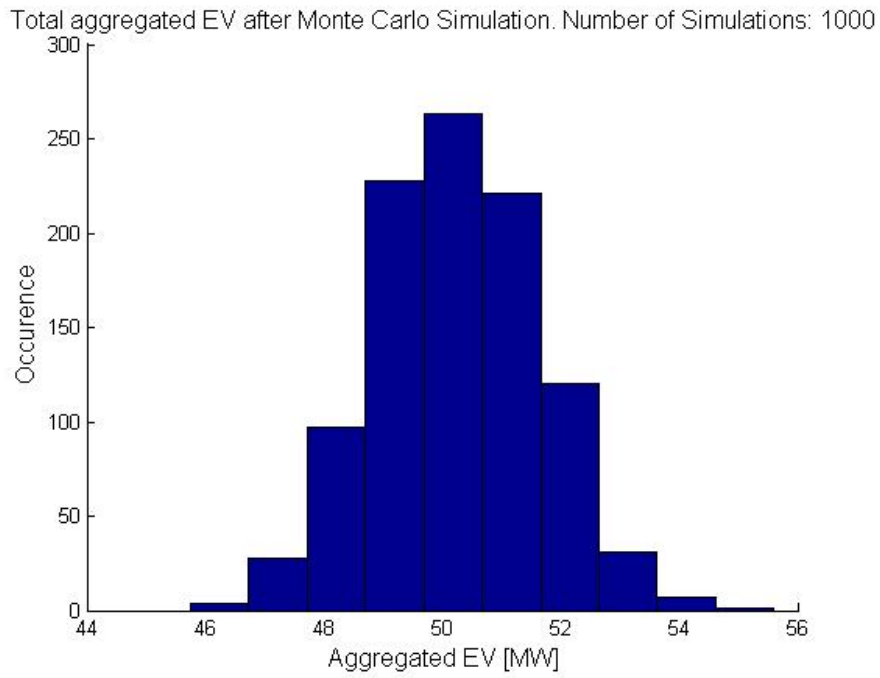
Figure 7.5: Histogram of total EV charging power for fast-charging
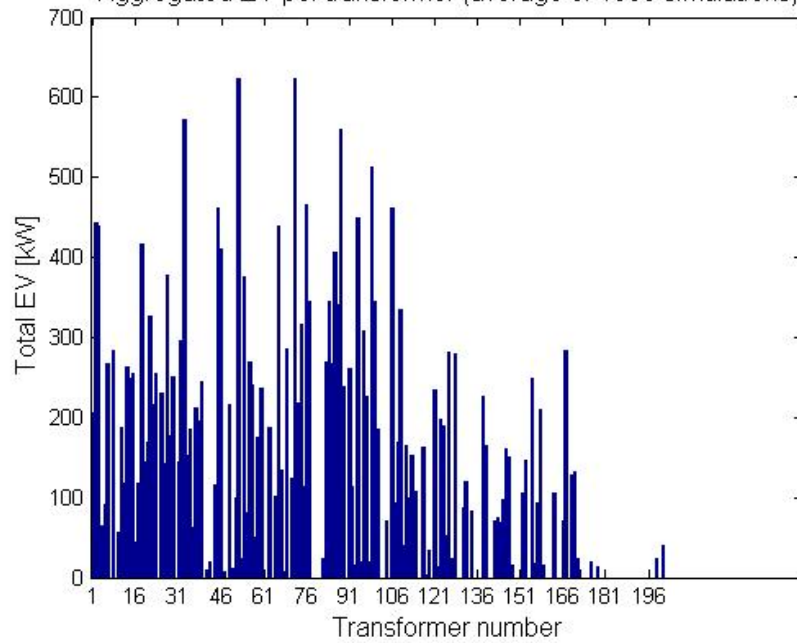
Figure 7.6: Average EV charging power per transformer station. Semi-fast charging.
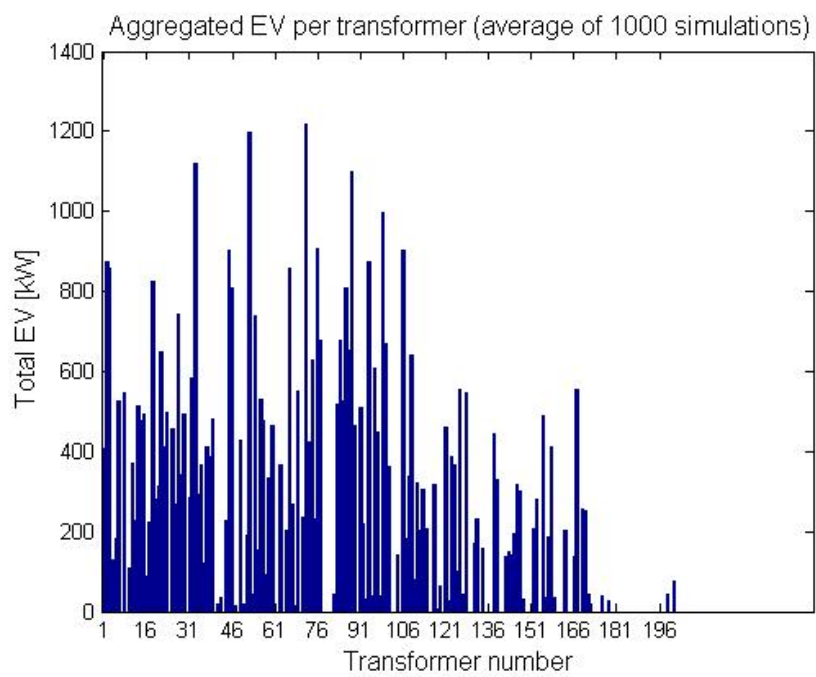
Figure 7.7: Average EV charging power per transformer station. Fast charging.

7.3.2 Power Flow Simulation Results

As was shown in figure 7.1, the Monte Carlo simulation model adds around 1167 charging vehicles randomly to the grid in GaP. The Monte Carlo simulations represent 1000 different spatial distributions of these charging vehicles. A power flow is run for every one of the 1000 distributions. The scenarios are considered to be in the evening, which means that no PV production is present in the grid and that all houses have the evening load calculated in section 5.3.1. This would represent a worst-case for the electro mobility studies, as the charging vehicles are added to the laready high evening load.

Semi-Fast Charging Scenario

Voltage Considerations When all of the electric vehicles charge with a power of 22 kW, it can be seen from figure 7.8 that there is a large voltage drop for many of the buses in the 0.4 kV grid. The -10% limit is violated for some of the buses. These undervoltages are located on buses where there is a charging vehicle in the end of a long feeder. A voltage drop of 11.6% is registered at the connection point of a residential building with a charging vehicle. This building is placed in the end of a radial line connected to transformer 84. Transformer 3 are also experiencing voltage drops of more than 10%. These are located at loads with charging vehicles far from the transformer. The 10 kV grid handles the increased loading better, but also here a voltage drop of nearly 3% is present, as seen in figure 7.9.
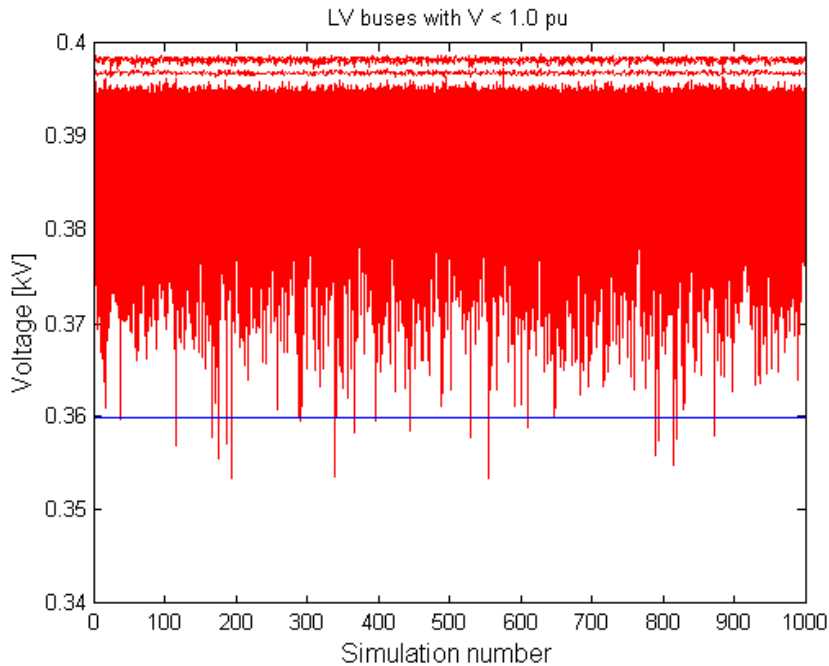
Figure 7.8: Voltages in 0.4 kV Network for Semi-Fast Charging Scenario
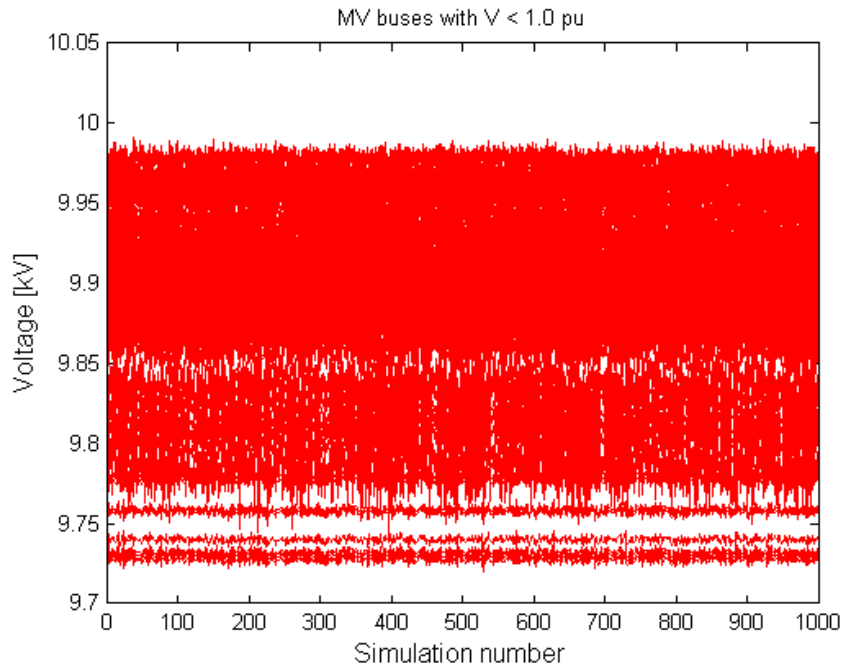
Figure 7.9: Voltages in 10 kV Network for Semi-Fast Charging Scenario

Overloading Considerations Measurements of the line currents were performed and compared to each line segments respective carrying capacity. A plot of the percentage number of lines with a current exceeding their thermal limits is shown in figure 7.10. As seen, some of the simulations leads to overloading of lines. As the current measurement and comparison is a time consuming iteration process, only the first 100 simulations were considered, due to time limitations.

The power at all 10 kV/0.4 kV transformers was recorded for all 1000 simulations, and the result of this is shown as a boxplot in figure 7.11. As seen, every single simulations lead to overloading of some transformers and the median is around 7% overloaded transformers.

80
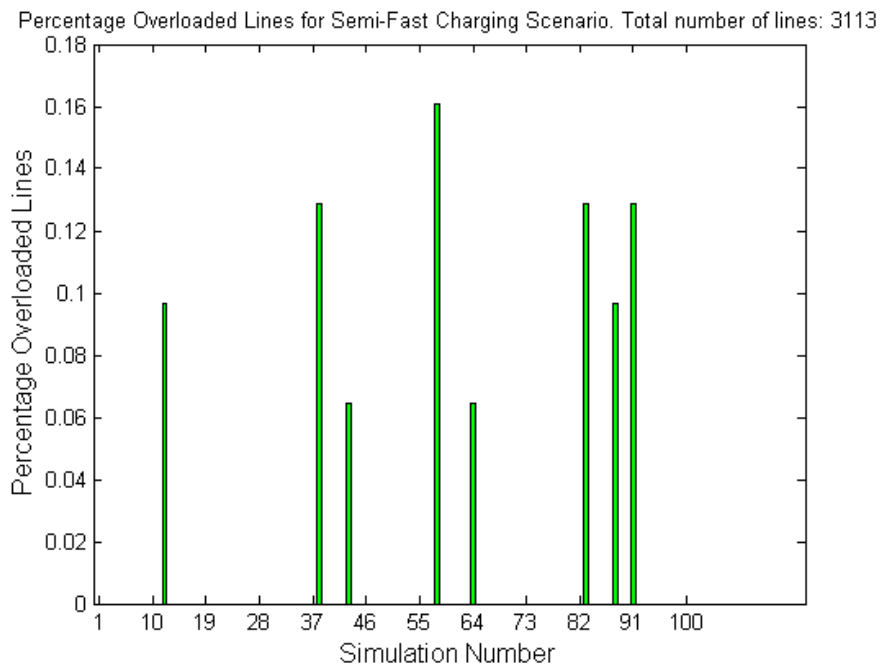
Figure 7.10: Overloaded Lines for Semi-Fast Charging Scenario
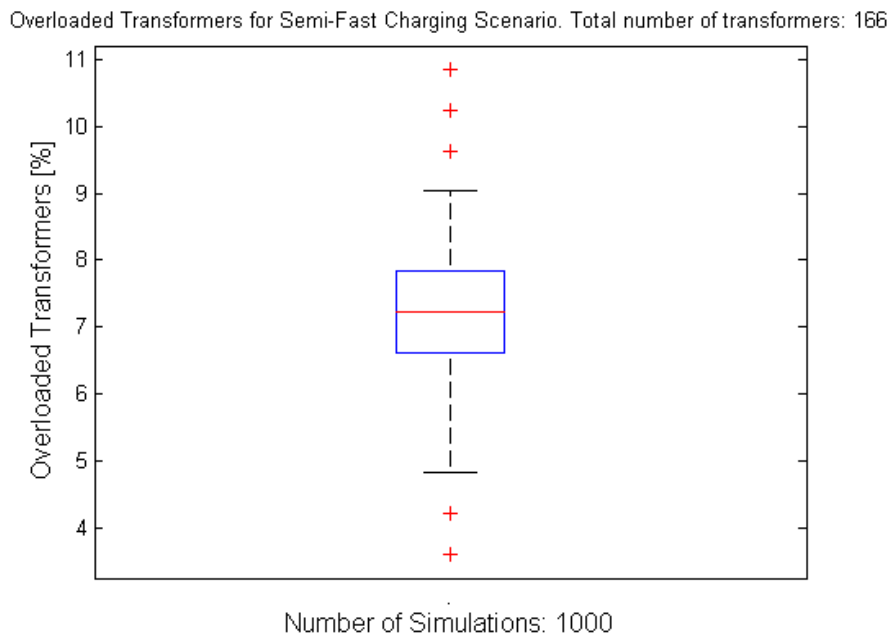
Figure 7.11: Overloaded Transformers for Semi-Fast Charging Scenario

81

Fast Charging Scenario

Voltage Considerations   Assuming that all the electric vehicles charge with a power of 43 kW, the voltage drop gets even larger, see figure 7.12 of the low voltage network grid and figure 7.14 of the medium voltage network. The -10% and the -15% limit are both violated for several buses in the low voltage grid. To see exactly how many buses that violate the -10% limit, a histogram of the occurrence of undervoltages is shown in figure 7.13. It can be seen, that for some of the distributions, up to 350 buses violate the permissible voltage drop limit. The lowest voltage is registered at a bus, which has two buildings connected it. The bus is located near the end of a line supplying residential houses on transformer 84. Closer investigations show that only one of the buildings connected to this bus are assigned a charging vehicle by the Monte Carlo model. The second house has a normal evening load of 0,854 kW. Even so, the voltage at this bus is 290,01 V, meaning a voltage drop of 27,5%. Also for this scenario, the 10 kV grid handles the increased loading better, but a voltage drop of nearly 6% is present, see figure 7.14.
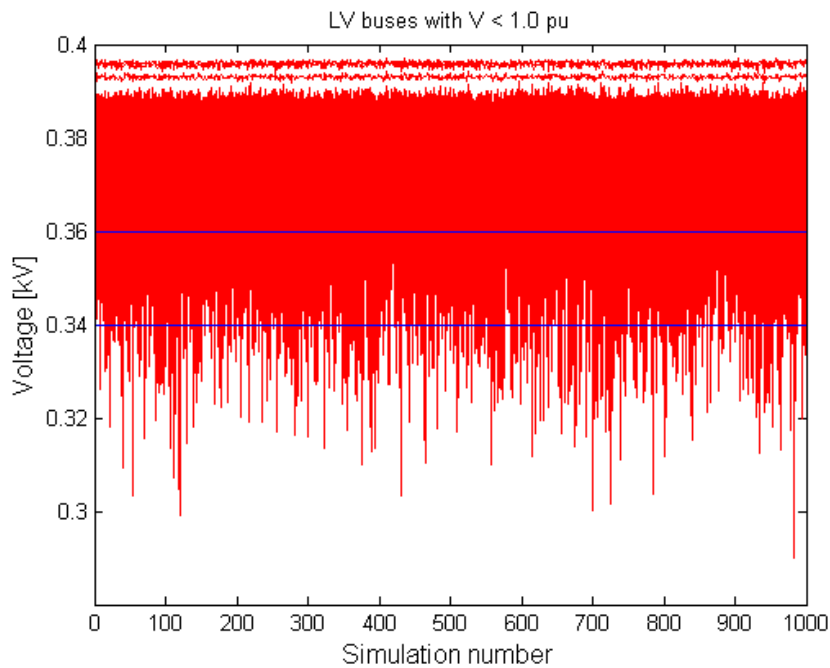
Figure 7.12: Voltages in 0.4 kV Network for Fast Charging Scenario
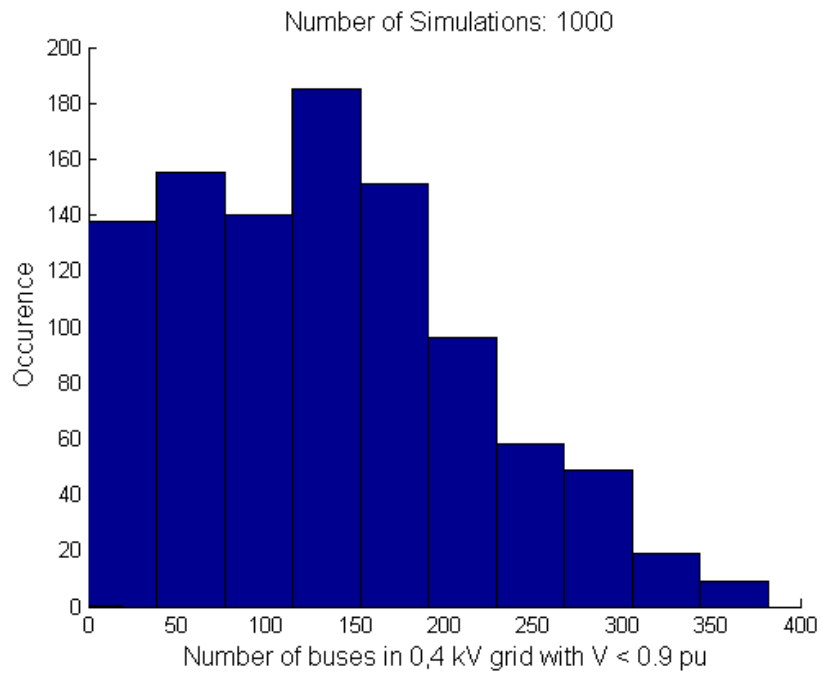
Figure 7.13: Under voltages in 0.4 kV Network for Fast Charging Scenario
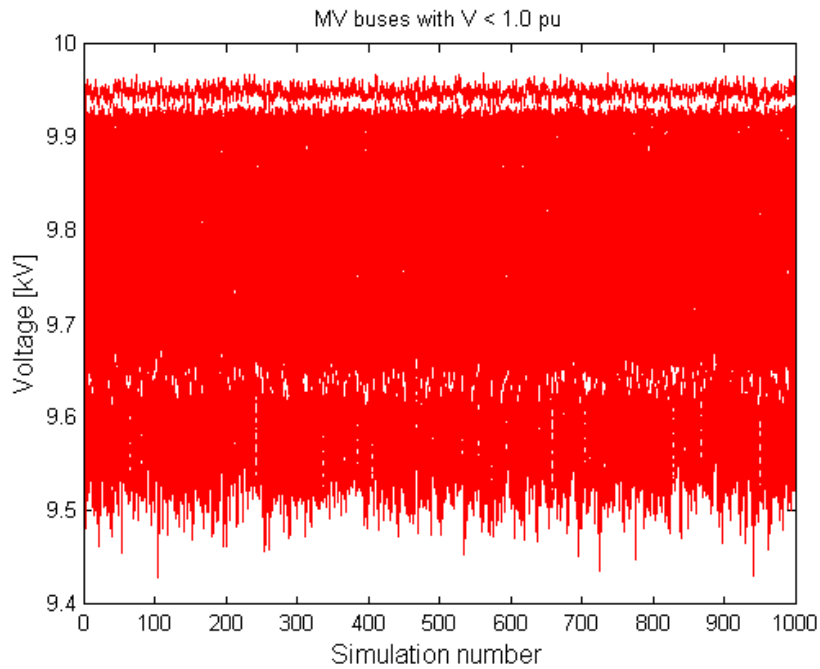
Figure 7.14: Voltages in 10 kV Network for Fast Charging Scenario

Overloading Considerations Line currents were measured and compared to the line segments carrying capacity. Also for this scenario, only 100 simulations were considered. A plot of the number of lines with a current exceeding their thermal limits is shown in figure 7.15. It can be seen, that every simulation results in overloaded lines, some with 2.5% overloaded lines.
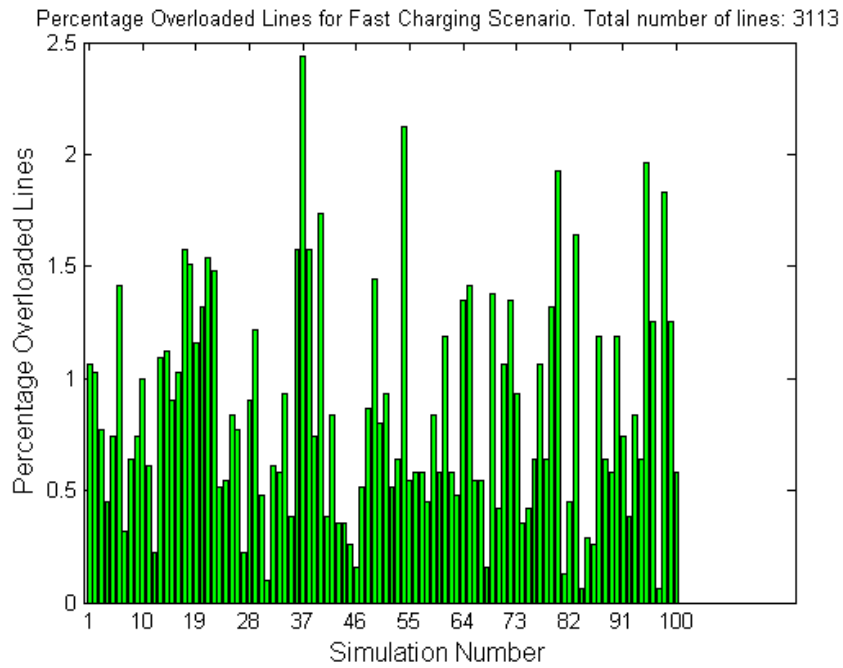
Figure 7.15: Overloaded Lines for Fast Charging Scenario

Also for this scenario, the power at the 10 kV/0.4 kV transformers were measured after every power flow simulation. The result is shown in figure 7.16 and shows that among the 166 transformers in the grid of GaP, up to 36% are overloaded when all electric vehicles simultaneously charge with 43 kW.
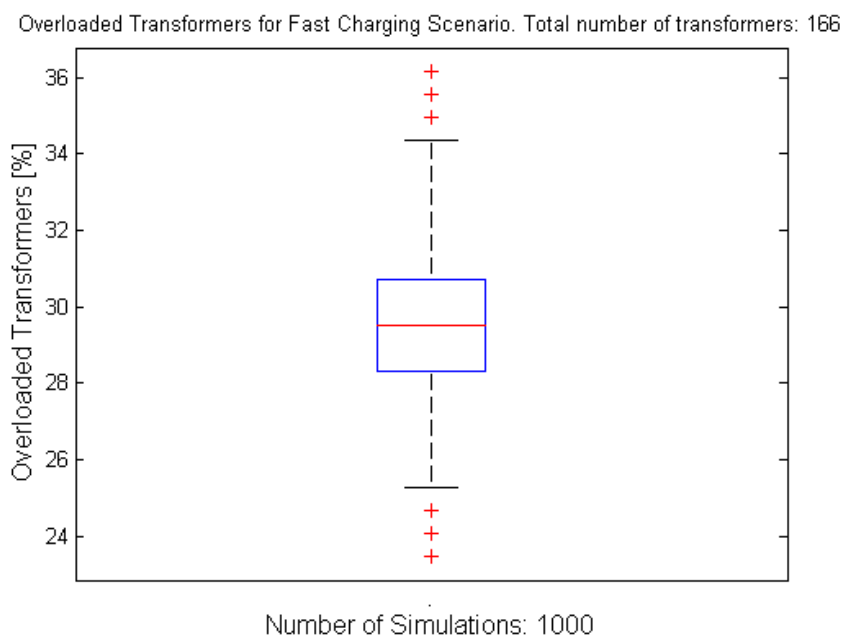
Figure 7.16: Overloaded Transformers for Fast Charging Scenario

Chapter 8

Summary and Discussions

The first part of work in this master's thesis consisted in programming a Monte Carlo simulation model in MatLab. This model creates spatial distributions of rooftop photovoltaic distributed generators and electric vehicles in the low voltage grid of Garmisch-Partenkirchen. The model can assign any desired amount of total generating capacity or charging power, but here the situation in year 2030 has been studied. Forecasts for the year have been defined and the Monte Carlo model was tuned in order to assign this projected amount of generating capacity and electric vehicles randomly among the nearly 8000 houses in Garmisch-Partenkirchen.

The second part of work in this thesis consisted in statistically examining the results from the Monte Carlo model and analyze the effect these have on the grid. This was done by running power flow simulations on the spatial distributions from the Monte Carlo results.

8.1 Photovoltaic

The number of required iterations that was calculated for the photovoltaic and electro mobility Monte Carlo models gave the necessary number of simulations in order to reach the forecast of future PV and EV in the grid. The resulting number was a couple of hundred simulations, yielding for both EV and PV, and thus 1000 simulations was enough to get an average value close to the forecast.

1000 different distributions from the Monte Carlo model were analyzed by running a power flow of every of the 1000 network situations. The overall result regarding voltage issues is that at day time, when the load is low and the production from the photovoltaic generators is high, at least 86% of the 0.4 kV buses have a voltage above nominal voltage. Some of the spatial distributions lead to overvoltages which violate the +3%-limit at certain buses. The summer day scenario is in clear contrast to the evening load scenario, where 100% of the 0.4 kV network nodes have a voltage below nominal voltage.

However, none of the lines are overloaded in regard to their thermal limits. Nor are any of the transformers overloaded with respect to their ratings.

These 1000 different spatial distributions are only a fraction of the number of possible distributions. The probability that any extreme situations, e.g. that a

large amount of PV or EV end up on the same remote feeder ends, is extremely low. As the time frame for this thesis was limited, the maximum number of simulations was set at 1000 and instead a worst case scenario was manually created. With this scenario, it was possible to clearly see the possible impact from distributed generation on the network operating system. The scenario was constructed based around transformer 23, by filling the entire roof of buildings along a single radial feeder with PV plants, starting from the feeder end. It was seen that the generation exceeds the total consumption for the transformer and as a result, active power is injected to the overlying 10 kV network. The generators cause large overvoltages at their point of connection and violate the thermal limits of some cables.

The findings from the photovoltaic studies show that on a most probable basis, this distribution network can handle most of the projected increase in photovoltaic capacity for the year of 2030. However, if large shares of the projected capacity is concentrated around a small area, e.g. on a single feeder, this can introduce several challenges. Before every new installation of photovoltaics among the grid connected customers, in can be wise to perform calculations regarding the network situation in order to verify that no lines or transformers will be overloaded.

If the network status has reached a point where the voltage at the customer connection point is no longer within the limits, voltage regulation could be an alternative. The mitigation of voltage rise is a large field of study which covers numerous technologies. In the specialization project that was written prior to this thesis, different regulation methods for mitigation of voltage issues caused by distributed generation were investigated. Due to time limitations, no regulation methods were tested in this thesis. However, two important voltage control alternatives to cope with the voltage rise will be briefly mentioned at the end.

8.2 Electro Mobility

As was shown in equation 2.5, the voltage drop along a line is dependent on both the power at the loads and the impedance of the line. In the electro mobility studies in this thesis, the loads were assigned electric vehicles with two different charging powers. In both cases the voltage drop increases significantly compared to the standard evening scenario. It was also seen that the voltage drop becomes larger when the charging power of the vehicles increases from semi-fast to fast charging. This confirms the theoretical foundations, that a voltage drop is dependent on the power of the receiving end.

The semi-fast charging scenario causes certain violations of the -10% voltage drop limit defined in the standard EN 50160. In addition, for every simulation there is overloading of transformers with a median value of around 7% overloaded transformers. Some simulations reach a value of 11% overloaded transformers. For a few lines the thermal limits are violated

Among the total 2487 low voltage buses, the fast charging scenario causes up to 350 buses to violate the -10% voltage drop limit. In addition, a large share of the bus voltages drop below the -15% - requirement. Up to 36% of the 166 transformers are overloaded, with a median of 29.5% overloaded transformers. In addition, thermal limits are violated for up to 2.5% of the lines. With a total

of 3113 lines in the distribution grid, this corresponds to up to 80 overloaded lines.

It is seen as a general trend, that the remote network users are especially vulnerable in regard to voltage drops. The lowest voltages are recorded at loads near the end of radial feeders or far from the transformer. This confirms what was elaborated in the theoretical foundations: the voltage drop increases with line length.

It was shown that the distribution grid will experience difficulties handling large scale integration of semi-fast and fast charging electric vehicles. Under-voltages will be a issue, as well as overloading of lines and transformers. Some possible solutions will be briefly discussed in the next section.

8.3   Voltage Regulation

Reactive Power Compensation   One solution for voltage rise mitigation can be to control the amount of reactive power at the point of connection. This is done by operating the generator in an under excited mode, at a leading power factor. Referring to table 2.1, the generator will then consume reactive power. The phasor diagrams from sections 2.2 and 2.3 are used to illustrate the concept. The phasor representing the imaginary part of the current $-Icq$ is drawn along the positive imaginary axis and the new situation can be seen in figure 8.1.
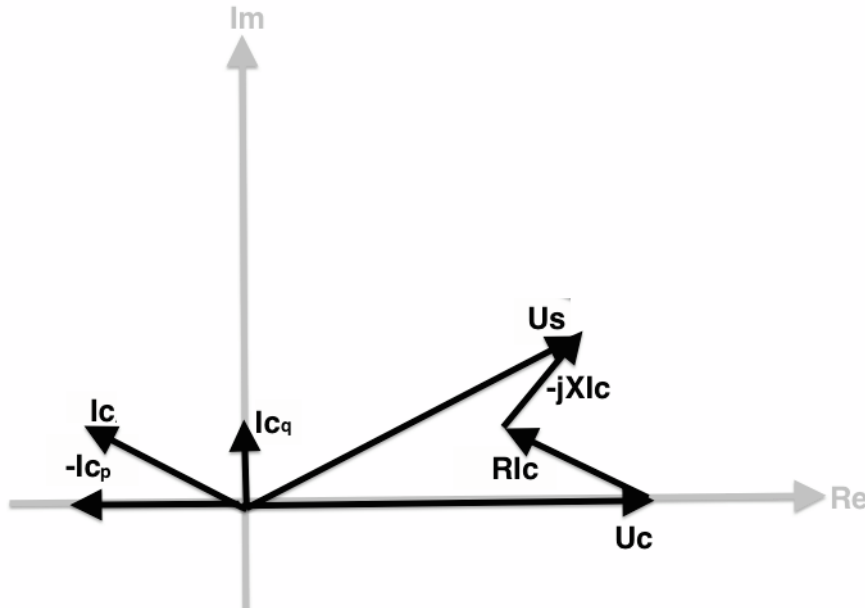
Figure 8.1: Phasor Diagram of Generator with Active Power production and Reactive Power control

The addition of $-Icq$ shrinks the difference between Us and Uc and mitigates the voltage rise. Thus, reactive power can be a solution to the overvoltages that distributed generation can cause.

Tap Changing Control   Distribution transformers have conventionally been designed in such a way that the tap settings cannot be changed under load. Just arising in Europe is on load tap changing mechanisms also for distribution transformers. With taps in the windings that enable changes in the turns ratio, it is possible with adjustments of a certain percentage above or below the nominal voltage rating of the transformer while the transformer is under load. The tap changing control can be integrated on the main transformer, or a new tap changing transformer can be added to a feeder. The latter is often used by power companies on power lines whose voltage varies widely with the load, due to high impedance between generators and the particular load [12]. On-load tap changers can also be operated in combination with line drop compensation, in which the turns ratio are adjusted depending on the actual voltage level in the network. Line drop compensation can be utilized by measuring the impedance and length of the line between the transformer and the critical generating unit/load [38]. In addition to this, the current through the transformer must be measured. Knowing the current and impedance of the line, the voltage rise can be found by voltage drop equation 2.1 presented in section 2.2. It should be mentioned that if tap changer control shall be utilized in networks with distributed generation, it is crucial that the controller have the capability to detect change in power flow direction.

Comparing the photovoltaic and electro mobility scenarios, large voltage differences between day time with PV generation and evening load with EV charging are seen. Even without the most extreme voltage rises and drops, these differences alone may necessitate voltage regulation.

Grid Reinforcement   It has been shown here that semi-fast and fast charging of electric vehicles leads to a clearly overloading of both lines and transformers. For the network to be able to integrate this amount of extra loading, grid reinforcement might be a necessary solution.

It can be mentioned that in Norway, the network is seen to handle electric vehicle charging better than in the study conducted in this thesis due to the fact that the Norwegian electricity network is dimensioned to supply heavy loading from electricity based heating.

Part IV

Conclusions and Outlook

Conclusions

After the theoretical foundations are carried out in part II, the analysis follows in part III. Chapter 4 gives an overview of the medium and low voltage distribution network in Garmisch-Partenkirchen in Germany. The city already has a large share of its electricity generation based on solar energy and is expecting a further increase in small scale photovoltaic installations. Due to former studies of the city, a database containing all buildings and their rooftop PV potential exist.

The power flow simulations in this master's thesis are performed using a network model based on the 10 kV and 0.4 kV distribution grid in Garmisch-Partenkirchen. The process of creating this network model in a suitable format is an ongoing process. This process, as well as the load modelling are elaborated in chapter 5.

All studies regarding photovoltaics are collected in chapter 6. Based on a projected increase in photovoltaic capacity, in addition to the database of the buildings and their PV potential, a Monte Carlo model is created. The model distributes rooftop photovoltaic generators randomly among the buildings in the 0.4 kV network of Garmisch-Partenkirchen. Subsequently, a large amount of power flow simulations are run on the same network model, but with new locations of the generating units for every simulation. Thus, it is possible to examine the different spatial distributions effect on the grid, i.e.: with regard to voltage issues, transformer and line overloading. The power flow simulations show large differences between the situation during winter evenings contra summer days. On a summer day, when the load is low and generation from photovoltaics is high, the active power feed-in from the distributed generators causes a voltage rise from the transformer to the connection points of the PV's. The voltage rises are in certain cases violating the 3% voltage rise limit set by the German VDE directive. This is in contrast to the conventional networks, where a continuously decreasing voltage along the distribution feeder can be assumed because only loads are connected.

Chapter 7 studies the effect from an increased integration of electric vehicles among the households in Garmisch-Partenkirchen. With some adjustments, the same Monte Carlo model was applied also for electro mobility simulations. Houses were randomly assigned charging vehicles in order to produce a large set of different spatial distributions. These were subsequently analyzed by power flow simulations. In the case of future electro mobility scenarios in Garmisch-Partenkirchen, no thorough studies have been carried out. For this reason, certain assumptions regarding future electro mobility situations in Garmisch-Partenkirchen were defined. The power flow simulations on the network with increased loading due to charging electric vehicles have shown that large undervoltages are expected. Some simulations resulted in buses with voltages 27.5% below nominal voltage. In addition, line overloading in certain parts of the grid is seen, as well as a high number of overloaded transformers. A total of 2.5% overloaded lines and 36% overloaded transformers are registered for some simulations.

To summarize, a Monte Carlo model that can add either generators or loads at random customer connection points in a distribution network has been created. Through statistical verification, it was shown that by tuning the probabilities of the random numbers, the model can assign the desired amount of

total PV or EV. Further, the thesis has investigated how an increase in small-scale photovoltaics and electro mobility could have an impact on the low voltage distribution grid. This is done by simulations on a network model constructed based on a real 10 kV/0.4 kV grid in Garmisch-Partenkirchen in Germany.

However, all the results in this thesis are based on assumed projections of the future. Even though national plans and goals lie behind the forecasts, assumption regarding a situation in 2030 are highly uncertain. It must be emphasized that the results here only serve as a rough estimate and an illustration of the impact an increase in PV and EV may have.

Outlook

At the very end, some thoughts regarding further work are proposed. First of all, there is yet large parts of the low voltage grid in Garmisch-Partenkirchen that haven't been digitalized. In this thesis, the simulations were performed on the 10 kV distribution network in addition to 16 low voltage networks, i.e.: 16 distribution transformers and their underlying 0.4 kV networks. However, there are 166 10 kV/0.4 kV transformers, so in order to find all critical buses and feeders, the remaining network parts should also be digitalized and studied.

If the entire voltage range from minimum voltages to maximum voltages should be included in the power flow simulation results, simulations on a very large number of spatial distributions must be performed. The statistical evaluation around this subject, i.e.: how to calculate the number of necessary power flow simulations, has yet to be studied. This leads to another point, that possible problematic spatial distributions could be closer investigated. In this thesis only one worst-case scenario was defined and analyzed, but the combination of long lines and large roofs are probably found on several places in the grid. Thus, further studies and investigations of critical network parts could be an option.

Numerous parameters could have been studied in regard to voltage quality and network characteristics. One of them is the power factor, which in these simulations was chosen to be 0.95. An option for further studies could be to evaluate the impact on the power quality due to the power factor. In addition, the power losses in the different spatial distributions could perhaps be of particular interest. When electricity is generated locally, and the need of power transfer decreases, the losses associated with this power transfer naturally reduces. This benefit of reduced losses would only yield up to the point where the local generation is so large that an equally amount of power must be transferred in the opposite direction. Thus, the loss impact from a growing number of distributed generators could be an object of future research.

If large scale implementation of distributed generation and/or electro mobility is going to succeed, the voltage and overloading issues that have been seen to occur, need to be solved. Some solutions were briefly mentioned here, but the implementation and simulation of these are an important field of research.

Bibliography

[1] Intergovernmental Panel on Climate Change. 2013. *Fifth Assessment Report (AR5) - Summary for Policymakers.*

[2] Dr. Suess, M. 2013. *Tour around the globe. Deep dive in 11 selected countries - Germany: the energy system today AND tomorrow.* Lecture notes from course "Challenges in Energy Markets" at the Technical University of Munich.

[3] Die Bundesregierung, 2010. *Electro mobility - moving towards the future.* [Online] Available at `http://www.bundesregierung.de/Content/EN/Artikel/2010/05/2010-05-03-elektromobilitaetsgipfel_en.html?nn=709674` [Accessed: 04.06.2014]

[4] Deutschlands Informationsportal zu Erneuerbaren Energien. *Erfolgsgeschichte EEG - das Erneuerbare-Energien-Gesetz.* [Online] Available at: `http://www.unendlich-viel-energie.de/themen/politik/erneuerbare-energien-gesetz-eeg/erfolgsgeschichte-eeg-das-erneuerbare-energien-gesetz` [Accessed: 20.01.2014]

[5] Deutschen Gesellschaft für Sonnenenergie. *EnergyMap Bundesrepublik Deutschland.* [Online] Available at: `http://www.energymap.info/energieregionen/DE/105.html`, [Accessed: 20.01.2014]

[6] Sand, K. 2013. *Supply Voltage Variations.* Lecture notes from class "Quality of Supply" at the Norwegian University of Science and Technology.

[7] VDE-AR-N 4105. 2011. *Erzeugungsanlagen am Niederspannungsnetz - Technische Mindestanforderungen für Anschluss und Parallellbetrieb von Erzeugungsanlagen am Niederspannungsnetz*, p.19

[8] BDEW (Bundesverband der Energie- und Wasserwirtschaft e.V.) 2007. *Technical Guideline: Generating Plants Connected to the Medium-Voltage Network*

[9] Hinz, A. 2012. *Der regelbare Ortsnetztransformator im Verteilungsnetz - Lösung aller Spannungsprobleme?* FGE Kolloquium, RWTH Aachen.

[10] Ulset, R., Holen, A.T., Wangensteen, I. 2010. *TET4155 Energisystemer.* University Compendium Norwegian University of Science and Technology, Trondheim

[11] Leiße, I. 2013. *Efficient Integration of Distributed Generation in Electricity Distribution Networks* Doctoral Dissertation. Department of Measurement Technology and Industrial Electrical Engineering. Lund University.

[12] Chapman, S. J., 2005, *Electric Machinery Fundamentals, Fourth Edition,*, International Edition, McGraw- Hill, United States of America

[13] Machowski, J., Bialek, J.W., Bumby, J.R. 2008. *Power System Dynamics - stability and control.* 2nd ed. Wiltshire: John Wiley and Sons.

[14] Dixon, J., Moran, L., Rodriguez, J. and Domke, R. *Reactive Power Compensation Technologies, State- of-the-Art Review*

[15] Bärthlein, E.M., 2012. *Development and Simulation of a Novel Concept for Reactive Power Compensation of Self-Induced Voltage Variations.* Diploma Thesis, Technische Universität München

[16] U.S. Department of Energy, 2007. *Solar Energy Grid Integration Systems.* Program Concept Paper [Online]. Available at `https://www1.eere.energy.gov/solar/pdfs/segis_concept_paper.pdf` [Accessed: 24.4.2014]

[17] Endegnanew, A. G., 2014. *Distribuert produksjon* [Email] Arnemo, C.S. (cecilsun@stud.ntnu.no) Date sent: 24. April 2014 15:19. [Accessed: 24.4.2014]

[18] Walpole, R.E., Myers, R. H., Myers, S.L. and Ye, K. 2007. *Probability and Statistics for Engineers and Scientists.* 8th ed. Upper Saddle River, New Jersey: Pearson Prentice Hall.

[19] Investopedia. 2014. *Monte Carlo Simulations.* [Online] Available at: `http://www.investopedia.com/terms/m/montecarlosimulation.asp` [Accessed: 3.3.2014]

[20] RiskAmp. *What is Monte Carlo Simulation?* Available at: `http://www.riskamp.com/files/RiskAMP%20-%20Monte%20Carlo%20Simulation.pdf` [Accessed: 02.05.2014]

[21] Gemeindewerke Garmisch-Partenkirchen. 2013, *Netzrelevante Daten.* `http://www.gemeindewerke-garmisch-partenkirchen.de/cms/index.php?option=com_content&task=view&id=17&Itemid=112` [Accessed: 7.05.2014]

[22] Gemeindewerke Garmisch-Partenkirchen. 2013, *Anlagenstatistik 2012.* Available at `http://www.gemeindewerke-garmisch-partenkirchen.de/cms/index.php?option=com_content&task=view&id=128&Itemid=161` [Accessed 28.06.2013]

[23] Gerdiken, C. 2013. *Analyse der Spannungsqualität für den Ausbau und die Netzintegration von Photovoltaikanlagen anhand der Modellkommune Elektromobilität Garmisch-Partenkirchen.* Master thesis, Hamburg University of Applied Science

[24] Institutt for Matematiske Fag, NTNU. 2000. *Tabeller og formler i statistikk.* Tapir Akademisk Forlag, ISBN 978-82-519-1595-3.

[25] Driels, M. R. and Shin, Y. S. 2004. *Determining the number of iterations for Monte Carlo simulations of weapon effectiveness.* Naval Postgraduate School, Montery, California

[26] Die Bundesregierung, n.d. *Leitmarkt und Leitanbieter für Elektromobilität.* [Online] Available at `http://www.bundesregierung.de/Webs/Breg/DE/Themen/Energiewende/Mobilitaet/podcast/_node.html` [Accessed: 05.05.2014]

[27] Ritter, N. and Vance, C. 2012. *Do Fewer People Mean Fewer Cars? - Population Decline and Car Ownership in Germany* Rheinisch-Westfälischen Instituts für Wirtschaftsforschung. Available at `http://www.rwi-essen.de/media/content/pages/publikationen/ruhr-economic-papers/REP_12_385.pdf` [Accessed: 01.06.2014]

[28] Ladestasjoner, n.d. *Type Elbilplugger.* Available at `http://www.ladestasjoner.no/ladehjelpen/teknologi/38-type-elbilplugger` [Accessed: 05.05.2014]

[29] Dorrestein, S. 2013. *EV Charging infrastructure.* International NoAE Project-Day E-mobility Aachen

[30] IFRI. 2010. *Electric Cars Charging Standards: Does it matter?* ifri - institut francais des relations internationales, Online Magazine.

[31] Pool, R., 2013. *Electric cars take off with new standards* IEC e-tech. Available at `http://www.iec.ch/etech/2013/etech_0413/tech-1.htm` [Accessed: 05.05.2014]

[32] Ehrlich, G. 2011. *EVs ready to charge ahead* IEC report. Available at `www.iec.ch/newslog/2011/doc/1611_EV_charging_standards.doc` [Accessed: 05.05.2014]

[33] Environment News Service, n.d. *Europe Sets Common Standard for Electric Vehicle Charging.* Available at `http://ens-newswire.com/2013/01/28/europe-sets-common-standard-for-electric-vehicle-charging/` [Accessed: 05.05.2014]

[34] NEPLAN. n.d. *NEPLAN Tutorial.* Available at: `http://www.deplan.gr/download/NEPLAN-Tutorial-Elec-Eng.pdf` [Accessed: 21.3.2014]

[35] Gemeindewerke Garmisch-Partenkirchen. 2014. *Normierte Profile: Standardlastprofile und Einspeiseprofile für die Gemeinde Garmisch-Partenkirchen.* Available at `http://www.gemeindewerke-garmisch-partenkirchen.de/cms/index.php?option=com_content&task=view&id=184&Itemid=234.` [Accessed: 24.3.2014]

[36] Heuck, K. and Dettmann, K. D. 2007. *Elektrische Energieversorgung - Erzeugung, Transport und Verteilung elektrischer Energie für Studium und Praxis.* Vieweg Verlag. p. 208.

[37] Schavemaker, P. and van der Sluis, L. 2008. *Electrical Power System Essentials*, Wiley, ISBN 978-0470-51027-8.

[38] Arnemo, C. S. 2013. *Simulation of Regulation Methods for Mitigation of Voltage Issues caused by Distributed Generation.* Specialization Project at the Norwegian University of Science and Technology in cooperation with General Electric Global Research.

Appendices

Appendix A

MatLab Script for Monte Carlo Simulation

```matlab
%Code to perform Monte Carlo Simulation on distribution among rooftop ↩
    PV
%Author: Cecilie S. Arnemo
%Last update: 17.05.14

clc; clear all;  format compact;
cd('M:\GaP_simulations')
RootPath = 'M:\GaP_simulations'; %for remote desktop NTNU
addpath('M:\GaP_simulations\GaP_matlabscripts');

%Read excel file containing houses
[house_numbers,house_char]=xlsread('/Excel_files/AllHouses_ha_idvs2');
id_ont_list = xlsread('\Excel_files\id_ont_list');
id_HA_list = xlsread('\Excel_files\id_HA_list');
MyMonteCarloInput.id_HA_list = id_HA_list;
PV_pot_all = house_numbers(:,16);
houses_IDs = house_numbers(:,1);
HA_IDs = house_numbers(:,1);
MyMonteCarloInput.HA_IDs = HA_IDs;
id_ont = house_numbers(:,3);
nr_houses = length(house_numbers(:,1));
category = house_char(2:length(house_char),9);
existing_pv = house_numbers(:,17);
existing_pv_logic = isnan(existing_pv);

%% ═══════════════ INPUT CHOICES ═══════════════ %%
%Decide which method to use for random number generation:
% method = 1:  Either P_typ, P_typ_redu or no PV is assigned
% method = 2: PV potential from normal distribution
method = 1;
nr_sim = 1000; %number of simulations
number_of_HA = 1577; %number of houses that are assigned a HA_ID
year = 2030;

forecast_2030 = [10.2465 0.661 2.3726 13.2801];
%specify calculated probabilities for Monte Carlo distribution
if method == 1 && year == 2030
    prob_res = [0.2692 (1-0.2692)];
    prob_agr = [0.3596 (1-0.3596)];
    prob_ind = [0.6052 (1-0.6052)];
elseif  method == 2 && year == 2030
        prob_res = [0.4 0.6];
        prob_agr = [0.4 0.6];
    prob_ind = [0.4 0.6];
end
%Define typical PV size
```

```matlab
PV_typ = 0;
PV_typ_redu = 0;
PV_typ_array = [10,30,75]; %Common values for larger PV plants
PV_typ_array_reduced = [3,10,20]; %Common values for smaller PV plants

%Run loop through all houses of GaP
%% ===== MAKE ARRAY of RANDOM NUMBERS for all houses === %%
MyMonteCarlo = MonteCarlo();
MyMonteCarloInput.method = method;
MyMonteCarloInput.category = category;
MyMonteCarloInput.nr_sim = nr_sim;
random_nr = 0;
random_numbers_houses = 0;
PV_all = 0;
std_dev_array = 0;
average_total_PV_array =0;
MyMonteCarloInput.id_ont = id_ont;
MyMonteCarloInput.id_ont_list = id_ont_list;
%%%%%%% to verify number of simulations needed %%%%%%%%%%%%%%
% iterations = 0;
% nr_iterations = 200;
% for t = 1:2:nr_iterations
%     iterations = iterations + 1;
%     nr_sim = t;
%     nr_simulations_array(iterations) = nr_sim
for j = 1:nr_sim
for i = 1:nr_houses

    PV_pot = PV_pot_all(i);
    %Input for Random Number Generator
    if(strcmp(category(i),'Wohnbau'))
        PV_typ = PV_typ_array(1);
        PV_typ_redu = PV_typ_array_reduced(1);
        prob = prob_res;
    elseif(strcmp(category(i),'Landwirtschaft'))
        PV_typ = PV_typ_array(2);
        PV_typ_redu = PV_typ_array_reduced(2);
        prob = prob_agr;
    elseif(strcmp(category(i),'GHD'))
        PV_typ = PV_typ_array(3);
        PV_typ_redu = PV_typ_array_reduced(3);
        prob = prob_ind;
    elseif(strcmp(category(i),'Militaer'))
        PV_typ = 0;
        PV_typ_redu = 0;
        PV_pot = 0;
        prob = 0;
    end %if(strcmp(category(i),'Wohnbau')

        MyMonteCarloInput.PV_typ = PV_typ;
        %For every house, generate a random number
        MyMonteCarloInput.prob = prob;
        temp = MyMonteCarlo.random_nr(MyMonteCarloInput);
        random_numbers_houses(i,j) = temp(1);
        size_prob = temp(2);
    if existing_pv_logic(i) == 0 && existing_pv(i) ~= 0  % check ←
        whether house has installed PV
        PV_all(i,j) = existing_pv(i); %assign the existing PV to the ←
            house
    else %%ASSIGN PV TO HOUSES ACCORDING TO THE RANDOM NUMBERS%%
     if method == 1
                if random_numbers_houses(i,j) == 1
                    if PV_pot < PV_typ && PV_pot <PV_typ_redu
                        PV_all(i,j) = 0;
                    elseif PV_pot < PV_typ && PV_pot >= PV_typ_redu
                        PV_all(i,j) = PV_typ_redu;
                    else
                        PV_all(i,j) = PV_typ;
                    end
                elseif random_numbers_houses(i,j) == 2
                    PV_all(i,j) = 0;
                end
     elseif method == 2
            if random_numbers_houses(i,j) == 0
```

```matlab
                                PV_all(i,j) = 0;
            else
                PV_all(i,j) = random_numbers_houses(i,j);

                        if PV_all(i,j) >= PV_pot && PV_all(i,j) >= ↵
                            PV_typ_redu
                            PV_all(i,j) = PV_typ_redu;
                        elseif PV_all(i,j) >= PV_pot && PV_all(i,j) <= ↵
                            PV_typ_redu
                            PV_all(i,j) = PV_pot;
                        end
                        if PV_all(i,j) < 0.2*PV_typ
                            PV_all(i,j) = 0;
                        end
            end
        end %if method == 1
    end %if existing_pv_logic == 0

    %% ═══════════════ ASSIGN PV TO HOUSES finished ═══════════════ %%
end %for i = 1:nr_houses
%%%PV per trafo per simulation %%
MyMonteCarloInput.PV_all = PV_all;
MyMonteCarloInput.simulation_number = j;
matrix_of_trafos(:,j) = MyMonteCarlo.find_PV_per_trafo_per_sim(↵
    MyMonteCarloInput);
MyMonteCarloInput.PV_HA = PV_all(1:number_of_HA,:);
MyMonteCarloInput.number_of_HA = number_of_HA;
temp = MyMonteCarlo.sum_ha_id(MyMonteCarloInput);
ha_id_and_ont = temp(:,1:2);
matrix_of_ha_id(:,j) = temp(:,3);
end %for j = 1:nr_simulations

    MyMonteCarloInput.PV_all = PV_all;
    %%%save the total aggregated power per simulation%%%
    temp = MyMonteCarlo.find_total_PV(MyMonteCarloInput)/1000;
    array_of_sums_res = temp(:,1);
    array_of_sums_agr = temp(:,2);
    array_of_sums_ind = temp(:,3);
    array_of_sums = temp(:,4);

%% ═══════ PLOTTING results of Monte Carlo ═══════════════ %%

MyMonteCarloInput.array_of_sums_res = array_of_sums_res;
MyMonteCarloInput.array_of_sums_agr = array_of_sums_agr;
MyMonteCarloInput.array_of_sums_ind = array_of_sums_ind;
MyMonteCarloInput.array_of_sums = array_of_sums;
%% ══ PLOT total aggregated PV after Monte Carlo simulation═══ %%
MyMonteCarlo.plot_total_PV(MyMonteCarloInput);
average_PV = MyMonteCarlo.find_average_total_PV(MyMonteCarloInput);

% %%%%%STATISTICS PLOT%%%%%%
% average_res(iterations) = average_PV(1);
% average_agr(iterations) = average_PV(2);
% average_ind(iterations) = average_PV(3);
% average_tot(iterations) = average_PV(4);
%
% std_deviation_res(iterations) = std(array_of_sums_res);
% std_deviation_agr(iterations) = std(array_of_sums_agr);
% std_deviation_ind(iterations) = std(array_of_sums_ind);
% std_deviation_tot(iterations) = std(array_of_sums);
%
% error_res(iterations) = (average_res(iterations) - forecast_2030(1))↵
%     /forecast_2030(1)*100;
% error_agr(iterations) = (average_agr(iterations) - forecast_2030(2))↵
%     /forecast_2030(2)*100;
% error_ind(iterations) = (average_ind(iterations) - forecast_2030(3))↵
%     /forecast_2030(3)*100;
% error_tot(iterations) = (average_tot(iterations) - forecast_2030(4))↵
%     /forecast_2030(4)*100;
%
% end %for t = 1:nr_iterations
    %%%═══════ plot statistical results ═══════════ %%%
%     MyMonteCarloInput.average_res = average_res;
%     MyMonteCarloInput.average_agr = average_agr;
```

```matlab
%       MyMonteCarloInput.average_ind = average_ind;
%       MyMonteCarloInput.average_tot = average_tot;
%
%       MyMonteCarloInput.std_deviation_res = std_deviation_res;
%       MyMonteCarloInput.std_deviation_agr = std_deviation_agr;
%       MyMonteCarloInput.std_deviation_ind = std_deviation_ind;
%       MyMonteCarloInput.std_deviation_tot = std_deviation_tot;
%
%       MyMonteCarloInput.error_res = error_res;
%       MyMonteCarloInput.error_agr = error_agr;
%       MyMonteCarloInput.error_ind = error_ind;
%       MyMonteCarloInput.error_tot = error_tot;
%
%       MyMonteCarloInput.nr_simulations_array = nr_simulations_array;
%       MyMonteCarlo.statistics_plot(MyMonteCarloInput);
nbins = size_prob; %number of equally spaced bins = number of random ↩
    numbers
MyMonteCarloInput.random_numbers_houses = random_numbers_houses;
MyMonteCarloInput.houses_IDs = houses_IDs;
MyMonteCarloInput.nbins = nbins;
MyMonteCarlo.plot_histogram(MyMonteCarloInput);
%MyMonteCarlo.plot_duration_curve(MyMonteCarloInput);
%MyMonteCarlo.plot_standard_deviation(MyMonteCarloInput);

% MyMonteCarloInput.prob_res = prob_res;
% MyMonteCarloInput.prob_agr = prob_agr;
% MyMonteCarloInput.prob_ind = prob_ind;
% MyMonteCarlo.plot_probability_distribution(MyMonteCarloInput);

%%Average PV per trafo %%
  MyMonteCarloInput.id_ont = id_ont;
  MyMonteCarloInput.id_ont_list = id_ont_list;
  list_of_trafos  = MyMonteCarlo.find_PV_per_trafo(MyMonteCarloInput);
  MyMonteCarloInput.list_of_trafos = list_of_trafos;
  MyMonteCarlo.plot_pv_per_trafo(MyMonteCarloInput);

%%====== plot std_dev vs. total PV =============== %%%%
% average_total_PV_array(x) = MyMonteCarlo.find_average_total_PV(↩
    MyMonteCarloInput)/1000; %MW
% MyMonteCarloInput.average_total_PV_array = average_total_PV_array;
% std_dev_array(x) = x;
% MyMonteCarloInput.std_dev_array = std_dev_array;

%%============== FIND the forecast for the specified year=========%%
% year = 2030;
% MyMonteCarloInput.year = year;
% forecast_2030 = MyMonteCarlo.find_forecast(MyMonteCarloInput)
%
% year = 2050;
% MyMonteCarloInput.year = year;
% forecast_2050 = MyMonteCarlo.find_forecast(MyMonteCarloInput)

%% == Save results of Monte Carlo to excel file  === %%
filename = ['Excel_files\Sim_results\PV_MC_',num2str(nr_sim),'↩
    sim_all_ONTs_HA.xlsx'];
headlines1 = [{'HC_ID'} {'Tr._ID'} {'PV[kW] per sim ->'}];
xlswrite(filename,headlines1,1,'A1:C1');
xlrange_HA_id = 'A2';
xlrange_ONT_id = 'B2';
xlrange_PV_all = 'C2';
xlswrite(filename,houses_IDs((number_of_HA+1):end),1,xlrange_HA_id);
xlswrite(filename,id_ont((number_of_HA+1):end),1,xlrange_ONT_id);
xlswrite(filename,PV_all((number_of_HA+1):end,:),1,xlrange_PV_all);
xlswrite(filename,ha_id_and_ont,3,'A2');
xlswrite(filename,matrix_of_ha_id,3,'C2');
xlswrite(filename,list_of_trafos,2,'A2');
xlswrite(filename,matrix_of_trafos,2,'D2');
headlines2 = [{'Tr._ID'} {'Houses per Tr.'} {'Average PV'} {'PV [kW] ↩
    per sim ->'}];
xlswrite(filename,headlines2,2,'A1:D1');
```

Appendix B

Matlab Script for Monte Carlo Simulation Functions

```matlab
%Functions for Monte Carlo RunSim script
%Author: Cecilie S. Arnemo
%Last update: 21.05.14

classdef MonteCarlo < handle
methods
    %constructor
    function obj = MonteCarlo()
        obj;
    end

function Output = random_nr(obj, MyMonteCarloInput)
    clear CLASSES;
    method = MyMonteCarloInput.method;
    prob = MyMonteCarloInput.prob;
    size_prob = 0;
    if method == 1
        r = rand; %uniformly distributed number between 0 and 1
        random_nr = sum(r >= cumsum([0, prob])); %generates integer ↩
            number with the specified probability
        size_prob = length(prob);
    elseif method == 2
        r = rand; %uniformly distributed number between 0 and 1
        random_nr = sum(r >= cumsum([0, prob])); %generates integer ↩
            number with the specified probability
        size_prob = length(prob);
        if random_nr == 1
            std_dev = MyMonteCarloInput.std_dev;
            PV_typ = MyMonteCarloInput.PV_typ;
            random_nr = PV_typ + std_dev*(randn(1));%generates a ↩
                number from normal distribution with mean = PV_typ and↩
                std=std_dev
        elseif random_nr == 2
            random_nr = 0; %no PV
        end %if random_nr
    end %if method
    Output = [random_nr size_prob];
end % random_nr

function plot_histogram(obj, MyMonteCarloInput)
    array_of_sums_res = MyMonteCarloInput.array_of_sums_res;
    array_of_sums_agr = MyMonteCarloInput.array_of_sums_agr;
    array_of_sums_ind = MyMonteCarloInput.array_of_sums_ind;
    array_of_sums = MyMonteCarloInput.array_of_sums;

    PV_all = MyMonteCarloInput.PV_all;
```

```matlab
            random_numbers_houses = MyMonteCarloInput.random_numbers_houses;
            method = MyMonteCarloInput.method;

            nbins = MyMonteCarloInput.nbins;
            houses_IDs = MyMonteCarloInput.houses_IDs;
            nr_sim = MyMonteCarloInput.nr_sim;
            if method == 1
                xvalues = [1,nbins];
            end
            %%==== plot histogram ======= %%
            histogram of the random numbers
            figure(1);
            if method == 1
            hist(random_numbers_houses', xvalues, nbins);
            else
            hist(random_numbers_houses');
            end
            ylabel('\fontsize{12} Occurence');
            xlabel('\fontsize{12} Distribution of random number');
            title([ '\fontsize{12} Monte Carlo,method ' num2str(method), '. ↵
                Number of Simulations: ' num2str(nr_sim)]);

            figure(2); hold on;
            ylabel('\fontsize{12} Occurence');
            xlabel('\fontsize{12} Size of PV plant [kW]');
            hist(PV_all');
            title([ '\fontsize{12} Monte Carlo,method  ' num2str(method), '. ↵
                Number of Simulations: ' num2str(nr_sim)]);

            figure(3); hold on;
            ylabel('\fontsize{12} Occurrence');
            xlabel('\fontsize{12} Aggregated PV [MW]');
            hist(array_of_sums);
            title([ '\fontsize{12} Monte Carlo simulation. Number of simulations↵
                : ' num2str(nr_sim)]);
        end % histogram

        function plot_duration_curve(obj, MyMonteCarloInput)
            PV_all = MyMonteCarloInput.PV_all;
            random_numbers_houses = MyMonteCarloInput.random_numbers_houses;
            method = MyMonteCarloInput.method;
            nbins = MyMonteCarloInput.nbins;
            houses_IDs = MyMonteCarloInput.houses_IDs;
            nr_sim = MyMonteCarloInput.nr_sim;
            l = length(PV_all(1,:));
            n = 0;
            xbins = 0;
            p = 0;
            P=0;
            %%==== plot duration curve ======= %%
            figure(4);
            [n,xbins] = hist(PV_all');
            p=n/l;
            P = cumsum(p);
            plot(cumsum(n),xbins);
            %set(gca,'ydir','rev');
            ylabel('\fontsize{12} Generation size [kW]');
            xlabel('\fontsize{12} Duration');
            title([ '\fontsize{12} MC method ' num2str(method), '. Number of ↵
                Simulations: ' num2str(nr_sim)]);
        end %duration curve

        function output=find_forecast(obj, MyMonteCarloInput)
            %%========== find the forecast for the specified year=========%%
            year =  MyMonteCarloInput.year; %year of forecast
            forecast = xlsread('20140128_Ausbauszenario_PV_je_Sektor');
            all_years = size(forecast);
            forecast_agricultural = 0;
            forecast_industry = 0;
            forecast_residential = 0;
            for i = 1:all_years(1)
                if forecast(i,1) == year
                    forecast_residential = forecast(i,2);
                    forecast_industry = forecast(i,3);
```

```matlab
                forecast_agricultural = forecast(i,4);
            end %forecast(i,1) == 2030
        end %i = 1:all_years
        PV_forecast_total_MW = forecast_agricultural+forecast_industry+↵
            forecast_residential;%kW
        output = [forecast_residential forecast_agricultural ↵
            forecast_industry PV_forecast_total_MW];
end

function output = find_total_PV(obj, MyMonteCarloInput)
    %find total aggregated PV among rooftops after Monte Carlo ↵
        simulation
    PV_all =  MyMonteCarloInput.PV_all;
    category = MyMonteCarloInput.category;
    size_PV_all = size(PV_all);
    nr_sim = size_PV_all(2);
    nr_houses = size_PV_all(1);
    %%============= find sum for every simulation ========%%
    array_of_sums = 0;
    array_of_sums_res = 0;
    array_of_sums_agr = 0;
    array_of_sums_ind = 0;
    for j = 1:nr_sim
        sum_PV = 0;
        sum_PV_res = 0;
        sum_PV_agr = 0;
        sum_PV_ind = 0;
        for i = 1:nr_houses
            if(strcmp(category(i),'Wohnbau'))
                sum_PV_res = sum_PV_res + PV_all(i,j);
            elseif(strcmp(category(i),'Landwirtschaft'))
                sum_PV_agr = sum_PV_agr + PV_all(i,j);
            elseif(strcmp(category(i),'GHD'))
                sum_PV_ind = sum_PV_ind + PV_all(i,j);
            end
            sum_PV = sum_PV + PV_all(i,j);
        end %i = nr_houses
    array_of_sums(j) = sum_PV;
    array_of_sums_res(j) = sum_PV_res;
    array_of_sums_agr(j) = sum_PV_agr;
    array_of_sums_ind(j) = sum_PV_ind;
    end %j = 1:nr_sim
    output = [array_of_sums_res', array_of_sums_agr', ↵
        array_of_sums_ind', array_of_sums'];
end %find_total_PV

function plot_total_PV(obj, MyMonteCarloInput)
        method =  MyMonteCarloInput.method;
    array_of_sums_res = MyMonteCarloInput.array_of_sums_res;
    array_of_sums_agr = MyMonteCarloInput.array_of_sums_agr;
    array_of_sums_ind = MyMonteCarloInput.array_of_sums_ind;
    array_of_sums = MyMonteCarloInput.array_of_sums;
        %% ============== PLOT ALL SUMS ============= %%%
        figure(5);
        nr_sim = MyMonteCarloInput.nr_sim;
        hold on
        plot(array_of_sums_res,'r')%,nr_sim,array_of_sums_agr/1000,'b↵
            ', nr_sim,array_of_sums_ind/1000, 'g',nr_sim, ↵
            array_of_sums_agr/1000, 'c')
        plot(array_of_sums_agr,'b')
        plot(array_of_sums_ind,'g')
        plot(array_of_sums,'c')
        set(gca,'ytick',[1:1:15])
        legend('Residential','Agricultural','Industry','Total' ,'↵
            Location', 'NorthEastOutside')
        title('Total aggregated PV after Monte Carlo simulation')
        ylabel('\fontsize{12} Total PV in grid [MW]');
        xlabel('\fontsize{12} Simulation number');
        hold off
%         figure(6)
%         plot(array_of_sums/1000,'c')
%         legend('Total', 'Location', 'NorthEastOutside')
%         title(['Total aggregated PV for MC method',num2str(method)])
%         ylabel('\fontsize{12} Total PV in grid [MW]');
```

```matlab
%          xlabel('\fontsize{12} Simulation number');
     end %plot_total_PV

function plot_standard_deviation(obj, MyMonteCarloInput)
     std_dev_array = MyMonteCarloInput.std_dev_array;
     average_total_PV_array = MyMonteCarloInput.average_total_PV_array;
     method =  MyMonteCarloInput.method;
     nr_sim = MyMonteCarloInput.nr_sim;
     figure(7);
     plot(average_total_PV_array,std_dev_array, 'r--*')
     title(['Total aggregated PV in GaP after Monte Carlo distribution'...
          ]);% method ',num2str(method)])
     xlabel(['\fontsize{12} Total PV [MW]. Every star represents ...
          average of ',num2str(nr_sim),' simulations']);
     ylabel('\fontsize{12} Standard deviation [kW]');
     set(gca,'ytick',[1:1:15])
end

function output = find_average_total_PV(obj, MyMonteCarloInput)
     %find average aggregated PV among rooftops after Monte Carlo ...
          simulation
     array_of_sums_res = MyMonteCarloInput.array_of_sums_res;
     array_of_sums_agr = MyMonteCarloInput.array_of_sums_agr;
     array_of_sums_ind = MyMonteCarloInput.array_of_sums_ind;
     array_of_sums = MyMonteCarloInput.array_of_sums;

     average_res = mean(array_of_sums_res);
     average_agr = mean(array_of_sums_agr);
     average_ind = mean(array_of_sums_ind);
     average_total = mean(array_of_sums);
     output = [average_res, average_agr, average_ind, average_total];
end %find_average_total_PV

function plot_probability_distribution(obj, MyMonteCarloInput)
     prob_res = MyMonteCarloInput.prob_res;
     prob_agr = MyMonteCarloInput.prob_agr;
     prob_ind = MyMonteCarloInput.prob_ind;
     method = MyMonteCarloInput.method;
     if method == 1
     %%==== bar plot of probabilities ====== %%
     figure(8); hold on;
     random_numbers_houses = MyMonteCarloInput.random_numbers_houses;
     method = MyMonteCarloInput.method;
     nbins = MyMonteCarloInput.nbins;
     houses_IDs = MyMonteCarloInput.houses_IDs;
     nr_sim = MyMonteCarloInput.nr_sim;
     xvalues = [1,nbins];
     prob = [prob_res;prob_agr;prob_ind];
     bar(xvalues,prob')
     set(gca,'xtick',[1:1:2])
     set(gca,'xticklabel',{'PV','No PV'}')
     ylabel('\fontsize{12} Probability');
     xlabel('\fontsize{12} Random Number');
     title(['\fontsize{12} Input for Monte Carlo Model']);
     legend('Residential','Agricultural','Industrial')
     elseif method == 2
     %%==== plot histogram ====== %%
     %histogram of the random numbers
      figure(9); hold on;
     hist(random_numbers_houses');
     ylabel('\fontsize{12} Occurence');
     xlabel('\fontsize{12} Distribution of random number');
     title(['\fontsize{12} Monte Carlo,method ' num2str(method), '. ...
          Number of Simulations: ' num2str(nr_sim)]);
     end
end %plot_probability_distribution

function output = find_PV_per_trafo(obj, MyMonteCarloInput)
          id_ont = MyMonteCarloInput.id_ont;
          PV_all = MyMonteCarloInput.PV_all;
          nr_houses = length(id_ont);
          nr_sim = MyMonteCarloInput.nr_sim;
          simulation_number = MyMonteCarloInput.simulation_number;
          matrix_trafo_and_PV = 0;
```

```matlab
            %%average PV per house:
              for i = 1:nr_houses
                    sum_PV_per_house = 0;
                 for j = 1:nr_sim
                    sum_PV_per_house = sum_PV_per_house + PV_all(i,j);
%                     if(strcmp(category(i),'Wohnbau'))
%                         sum_PV_res = sum_PV_res + PV_all(i,j);
%                     elseif(strcmp(category(i),'Landwirtschaft'))
%                         sum_PV_agr = sum_PV_agr + PV_all(i,j);
%                     elseif(strcmp(category(i),'GHD'))
%                         sum_PV_ind = sum_PV_ind + PV_all(i,j);
%                     end
                 end %j = 1:nr_sim
                        average_PV_per_house(i) = sum_PV_per_house/nr_sim;
                        matrix_trafo_and_PV(i,1) = id_ont(i);
                        matrix_trafo_and_PV(i,2) = average_PV_per_house(i);
                        %matrix_trafo_and_PV(i,simulation_number) = ↩
                           sum_PV_per_house
              end %i = nr_houses
    %%make list of pv per transformer:
    id_ont_list = MyMonteCarloInput.id_ont_list;
    for i = 1:length(id_ont_list)
        sum_pv_trafo = 0;
        nr_houses = 0;
    for j=1:length(matrix_trafo_and_PV) %look for trafo number
        if matrix_trafo_and_PV(j,1) == id_ont_list(i,1)
            sum_pv_trafo = sum_pv_trafo + matrix_trafo_and_PV(j,2);
            nr_houses = nr_houses + 1;
        end
    end %for j=1:length(matrix_trafo_and_PV)
        list_of_trafos(i,1) = id_ont_list(i,1);
        list_of_trafos(i,2) = nr_houses;
        list_of_trafos(i,3) = sum_pv_trafo;
    end %for i = 1:length(id_ont_list)

        output = list_of_trafos;

end %PV_per_trafo

function output = sum_ha_id(obj, MyMonteCarloInput)
        PV_HA = MyMonteCarloInput.PV_HA;
        nr_sim = MyMonteCarloInput.nr_sim;
        HA_IDs = MyMonteCarloInput.HA_IDs;
        nr_houses = MyMonteCarloInput.number_of_HA;
        id_ont = MyMonteCarloInput.id_ont;
        %for s = 1:nr_sim

        simulation_number = MyMonteCarloInput.simulation_number;
        matrix_HA_and_PV = 0;
        %%PV per row:
          for i = 1:nr_houses
                    matrix_HA_and_PV(i,1) = HA_IDs(i);
                    matrix_HA_and_PV(i,2) = id_ont(i);
                    matrix_HA_and_PV(i,3) = PV_HA(i,simulation_number);
          end %i = nr_houses
     %%make list of pv per HA_id:
    id_HA_list = MyMonteCarloInput.id_HA_list;

    for i = 1:length(id_HA_list)
        sum_pv_HA = 0;
    for j=1:length(matrix_HA_and_PV) %look for trafo number
        if matrix_HA_and_PV(j,1) == id_HA_list(i,1)
            sum_pv_HA = sum_pv_HA + matrix_HA_and_PV(j,3);
            this_ONT =  matrix_HA_and_PV(j,2);
        end
    end %for j=1:length(matrix_trafo_and_PV)
        list_of_HA(i,1) = id_HA_list(i);
        list_of_HA(i,2) = this_ONT;
        list_of_HA(i,3) = sum_pv_HA;
    end %for i = 1:length(id_ont_list)

    output = [list_of_HA];
end %sum_ha_id
```

```matlab
function output = find_PV_per_trafo_per_sim(obj, MyMonteCarloInput)
    id_ont = MyMonteCarloInput.id_ont;
    PV_all = MyMonteCarloInput.PV_all;
    nr_houses = length(id_ont);

    simulation_number = MyMonteCarloInput.simulation_number;
    matrix_trafo_and_PV = 0;
    %%PV per house:
      for i = 1:nr_houses
                matrix_trafo_and_PV(i,1) = id_ont(i);
                matrix_trafo_and_PV(i,2) = PV_all(i,simulation_number);
      end %i = nr_houses

 %%make list of pv per transformer:
id_ont_list = MyMonteCarloInput.id_ont_list;

for i = 1:length(id_ont_list)
    sum_pv_trafo = 0;
for j=1:length(matrix_trafo_and_PV) %look for trafo number
    if matrix_trafo_and_PV(j,1) == id_ont_list(i)
        sum_pv_trafo = sum_pv_trafo + matrix_trafo_and_PV(j,2);
    end
end %for j=1:length(matrix_trafo_and_PV)
    list_of_trafos(i) = sum_pv_trafo;
end %for i = 1:length(id_ont_list)

    output = list_of_trafos;

end %PV_per_trafo_per_sim

function plot_pv_per_trafo(obj, MyMonteCarloInput)
    list_of_trafos = MyMonteCarloInput.list_of_trafos;
    number_of_trafos = length(list_of_trafos);
    nr_sim = MyMonteCarloInput.nr_sim;
    list_of_trafos = list_of_trafos(1:(number_of_trafos-4),:);
    figure(10)
    bar(list_of_trafos(:,1),list_of_trafos(:,3))
    xlabel('\fontsize{12} Transformer number');
    ylabel('\fontsize{12} Total PV [kW]');
    title(['\fontsize{12} Aggregated PV per transformer (average of ' ...
        num2str(nr_sim),' simulations)']);
    set(gca,'xtick',[1:15:202])

end %function

function statistics_plot(obj, MyMonteCarloInput)
     figure(11);
     hold on;
    error_res = MyMonteCarloInput.error_res;
    error_agr = MyMonteCarloInput.error_agr;
    error_ind = MyMonteCarloInput.error_ind;
    error_tot = MyMonteCarloInput.error_tot;
    nr_simulations_array = MyMonteCarloInput.nr_simulations_array;
    plot(nr_simulations_array,error_res,'r');
    plot(nr_simulations_array,error_agr,'m');
    plot(nr_simulations_array,error_ind,'g');
    plot(nr_simulations_array,error_tot,'c');
    legend('Residential','Agricultural','Industry','Total')
    title(['Percentage Error of Sample Mean vs. Number of Simulations ...
        per Sample'])
    ylabel('\fontsize{12} Percentage Error');
    xlabel('\fontsize{12} Number of Simulations');
    hold off

    figure(12);
    hold on;
    std_deviation_res = MyMonteCarloInput.std_deviation_res;
    std_deviation_agr = MyMonteCarloInput.std_deviation_agr;
    std_deviation_ind = MyMonteCarloInput.std_deviation_ind ;
    std_deviation_tot = MyMonteCarloInput.std_deviation_tot;
    plot(nr_simulations_array,std_deviation_res,'r');
    plot(nr_simulations_array,std_deviation_agr,'m');
    plot(nr_simulations_array,std_deviation_ind,'g');
    plot(nr_simulations_array,std_deviation_tot,'c');
```

```matlab
        legend('Residential','Agricultural','Industry','Total');
        title(['Standard Deviation of Sample vs. Number of Simulations per↵
            Sample']);
        ylabel('\fontsize{12} Standard deviation');
        xlabel('\fontsize{12} Number of Simulations');
        hold off

        figure(13);
        hold on;
        average_res = MyMonteCarloInput.average_res;
        average_agr = MyMonteCarloInput.average_agr;
        average_ind = MyMonteCarloInput.average_ind;
        average_tot = MyMonteCarloInput.average_tot;
        plot(nr_simulations_array,average_res,'r');
        plot(nr_simulations_array,average_agr,'m');
        plot(nr_simulations_array,average_ind,'g');
        plot(nr_simulations_array,average_tot,'c');
        legend('Residential','Agricultural','Industry','Total');
        title(['Mean Value of Sample vs. Number of Simulations per Sample'↵
            ]);
        ylabel('\fontsize{12} Mean of sample [MW]'); %$\bar{X}$','↵
            interpreter','latex');
        xlabel('\fontsize{12} Number of Simulations per Sample');
        hold off
        forecast_PV = MyMonteCarloInput.forecast_PV;
            figure(14);
         hold on;
        plot(nr_simulations_array,average_tot,'c');
        refline(0,forecast_PV(4));
        title(['Mean Value of Sample vs Number of Simulations per Sample. ↵
            Total']);
        ylabel('\fontsize{12} Mean of sample [MW]'); %$\bar{X}$','↵
            interpreter','latex');
        xlabel('\fontsize{12} Number of Simulations per Sample');
        hold off
                figure(15);
        hold on;
        plot(nr_simulations_array,average_res,'r');
        refline(0,forecast_PV(1));
        title(['Mean Value of Sample vs Number of Simulations per Sample. ↵
            Residential.']);
        ylabel('\fontsize{12} Mean of sample [MW]'); %$\bar{X}$','↵
            interpreter','latex');
        xlabel('\fontsize{12} Number of Simulations per Sample');
        hold off
                    figure(16);
        hold on;
        plot(nr_simulations_array,average_agr,'m');
        refline(0,forecast_PV(2))
        title(['Mean Value of Sample vs Number of Simulations per Sample. ↵
            Agriculture.']);
        ylabel('\fontsize{12} Mean of sample [MW]'); %$\bar{X}$','↵
            interpreter','latex');
        xlabel('\fontsize{12} Number of Simulations per Sample');
        hold off
                    figure(17);
        hold on;
        plot(nr_simulations_array,average_ind,'g');
        refline(0,forecast_PV(3))
        title(['Mean Value of Sample vs Number of Simulations per Sample. ↵
            Industrial.']);
        ylabel('\fontsize{12} Mean of sample [MW]'); %$\bar{X}$','↵
            interpreter','latex');
        xlabel('\fontsize{12} Number of Simulations per Sample');
        hold off
    end %function statistics_splot

    function trafo_histogram(obj, MyMonteCarloInput)
        nr_sim = MyMonteCarloInput.nr_sim;
        nr_LVov_sim = MyMonteCarloInput.nr_LVov_sim;
        nr_MVov_sim = MyMonteCarloInput.nr_MVov_sim;

            figure(18); hold on;
            ylabel('\fontsize{12} Occurence');
```

```matlab
            xlabel('\fontsize{12} Number of buses in 0,4 kV grid with V > ←
                1.0 pu');
            hist(nr_LVov_sim');
            %hist(array_of_sums_res);
            title(['\fontsize{12} Number of Simulations: ' num2str(nr_sim)←
                ]);

            figure(19); hold on;
            ylabel('\fontsize{12} Occurence');
            xlabel('\fontsize{12} Number of buses in 10 kV grid with V > ←
                1.0 pu');
            hist(nr_MVov_sim);
            %hist(array_of_sums_res);
            title(['\fontsize{12} Number of Simulations: ' num2str(nr_sim)←
                ]);
    end %trafo_histogram

    function trafo_histogram_ev(obj, MyMonteCarloInput)
            nr_sim = MyMonteCarloInput.nr_sim;
            nr_LVov_sim = MyMonteCarloInput.nr_LVov_sim;
            nr_MVov_sim = MyMonteCarloInput.nr_MVov_sim;

            figure(20); hold on;
            ylabel('\fontsize{12} Occurence');
            xlabel('\fontsize{12} Number of buses in 0,4 kV grid with V < ←
                0.9 pu');
            hist(nr_LVov_sim');
            %hist(array_of_sums_res);
            title(['\fontsize{12}  Number of Simulations: ' num2str(nr_sim)←
                ]);

            figure(21); hold on;
            ylabel('\fontsize{12} Occurence');
            xlabel('\fontsize{12} Number of buses in 10 kV grid with V < ←
                0.9 pu');
            hist(nr_MVov_sim);
            %hist(array_of_sums_res);
            title(['\fontsize{12} Number of Simulations: ' num2str(nr_sim)←
                ]);
    end %trafo_histogram_ev

    end %methods (Static)
    end %classdef
```

Appendix C

MatLab Script for Worst Case Scenario

```matlab
%% Code to perform power flow of Worst Case Scenario
%% Author: Cecilie S. Arnemo
%% Last update: 28.05.14

clc; clear all;  format compact;
RootPath = 'M:\GaP_simulations\'; %for remote desktop at NTNU
cd([RootPath]);
addpath([RootPath 'GaP_matlabscripts']);
addpath([RootPath 'GaP_dss\WorstCaseScenario']);

%% Network data initialisation
NetworkFilename = ([RootPath 'GaP_dss\WorstCaseScenario\↩
    GW_GAP_MS_20131128.dss']);
mySimulator = NetSim(NetworkFilename);
%% Set voltage bases
mySimulator.SetVoltBase('ALL',[0.4 2.1 10 20 110]);
mySimulator.SetParam('load','ALL','Vminpu',0);
mySimulator.SetParam('load','ALL','Vmaxpu',5);
mySimulator.SetParam('load','ALL','model',1);
mySimulator.SetParam('Generator','ALL','Vminpu',0);
mySimulator.SetParam('Generator','ALL','Vmaxpu',5);
mySimulator.SetParam('Generator','ALL','model',1);
mySimulator.SolveNetwork;

V_pu_original=mySimulator.GetBusPosSeqVolt('all','pu');

%% Load excel file with cable limits s%%%%
[cable_nr, cable_char] = xlsread('Excel_files\cablelimits');
cable_types = cable_char(:,1);
cable_limits = cable_char(:,8);
xlrange_HA_id = 'A1:A7391';
xlrange_ONT_id = 'B1:B7391';
xlrange_PV_all = 'C1:ALN7391';
HA_id = xlsread(filename,1,xlrange_HA_id);
ONT_id = xlsread(filename,1,xlrange_ONT_id);
PV_all = xlsread(filename,1,xlrange_PV_all);
ONT_id_SUM = xlsread(filename,2);
NrHouses = length(HA_id);
ONTs = size(ONT_id_SUM);
NrONTs = ONTs(1);
%% Number of network buses
NrBus = mySimulator.NrBuses;
AllLoads =  mySimulator.GetList('Load','all');
load_and_volt = mySimulator.GetParam('Load','all','kV');
NrLoads = length(AllLoads);
Pload = 0.555;
```

```matlab
%% To find the PV potential of the roofs
[house_numbers,house_char] = xlsread('/Excel_files/←
    AllHouses_ha_worstcase');%
%numbers stored in "numbers", textstrings stored in "char"
id_ont_list = xlsread('\Excel_files\id_ont_list');
id_HA_list = xlsread('\Excel_files\id_HA_list');
PV_pot_all = house_numbers(:,16);
number_of_HA = 1577;
houses_IDs = house_numbers(1:number_of_HA,1);
%% === ASSIGN VALUES TO LOADS === %%
pf = 0.95;
for i = 1:NrLoads
if strcmp(load_and_volt(i,2),'0.4') %%%%check if load is a LV load
    mySimulator.SetParam('Load',cell2mat(load_and_volt(i,1)),'kW', ←
        Pload);
    mySimulator.SetParam('Load',cell2mat(load_and_volt(i,1)),'pf', pf)←
        ;
    mySimulator.SetParam('Load',cell2mat(load_and_volt(i,1)),'kvar', ←
        Pload*tan(acos(pf)))
elseif strcmp(load_and_volt(i,2),'10') %%%%check if load is a ←
    transformer
    noOfThisTransformer = str2double(load_and_volt(i,1));
    ix_row_nr = find(ONT_id_SUM(:,1)==noOfThisTransformer);%%%%look up←
        the ONT in MCS excel file
    noOfHousesThisTransformer = ONT_id_SUM(ix_row_nr,2); %%%%%find ←
        number of houses per ONT
    mySimulator.SetParam('Load',cell2mat(load_and_volt(i,1)),'kW',←
        noOfHousesThisTransformer*Pload);
    mySimulator.SetParam('Load',cell2mat(load_and_volt(i,1)),'pf', pf)←
        ;
    mySimulator.SetParam('Load',cell2mat(load_and_volt(i,1)),'kvar', ←
        noOfHousesThisTransformer*Pload*tan(acos(pf)));
end
end
AllLoads =  mySimulator.GetList('Load','all');
CustomerID_array = [0,5909,5806,5804,1022,1123,5836,5811,5535];%←
    ,5813,822,2631,2632,2633,2615,2614,2612,820,821,815]
nr_customers = length(CustomerID_array);
for s = 1:nr_customers
%s= nr of PVs on feeder
 %% Add passive generators %%%%
        BusName = mySimulator.GetParam('Load',num2str(←
            CustomerID_array(s)),'Bus1');%find bus where house is ←
            connected
        mySimulator.AddElem('Generator',num2str(CustomerID_array(s))←
            ,'bus1',BusName{2,2},'kW',0,'kvar',0)
end
%% === Assign PV to loads === %%
for s = 1:nr_customers
    s  %s= nr of PVs on feeder
        for h = 1:s
            ix =  find(houses_IDs==CustomerID_array(h));
            PV_pot = 0;
             for i = 1:length(ix);
              PV_pot = PV_pot + PV_pot_all(ix(i));
             end
             PV_pot_array(h) = PV_pot;
            %%%% Assign PV to house%%%%
            mySimulator.SetParam('Generator',num2str(←
                CustomerID_array(h)),'kW',PV_pot_array(h))
            mySimulator.SetParam('Generator',num2str(←
                CustomerID_array(h)),'kvar',0)
            mySimulator.SetParam('Generator',num2str(←
                CustomerID_array(h)),'pf',1)
        end %h

%% === Solve the circuit === %%%%
    mySimulator.SetParam('Generator','ALL','Vminpu',0);
    mySimulator.SetParam('Generator','ALL','Vmaxpu',5);
    mySimulator.SetParam('Generator','ALL','model',1);
    mySimulator.DSSObj.Text.command = ('solve mode=snap');
    mySimulator.SolveNetwork;
%% === Save Results ===
    %Save Bus Voltages:
```

```matlab
        V_temp = mySimulator.GetBusPosSeqVolt('ALL','pu');
        VBusAllpu(:,1) = V_temp(2:end,1);
        VBusAllpu(:,s+1) = (V_temp(2:end,2));
        temp_kV = mySimulator.GetBusPosSeqVolt('all','kV');
        VBusAll_kV(:,1) = temp_kV(2:end,1);
        VBusAll_kV(:,s+1) = temp_kV(2:end,2);
        %%Save Powers
        temp_S=mySimulator.GetPosSeqPower('Transformer','TR2-1076931842',←
            'kw');
        P(s) = cell2mat(temp_S(2,5));
        Q(s) = cell2mat(temp_S(2,6));
    end %for s=1:nr_customer

    %% ===== Postprocessing results of power flow ===== %%
        %% Voltages
        V_pu = cell2mat(VBusAllpu(:,2:end));
        V_kV = cell2mat(VBusAll_kV(:,2:end));
        VBusAll_kV(:,1) = temp_kV(2:end,1);
        VBusAll_kV(:,s+1) = temp_kV(2:end,2);

        %% Find PV-buses
        V_customer_buses=zeros(nr_customers,nr_customers);
    for h=1:nr_customers
        BusName = mySimulator.GetParam('Load',num2str(CustomerID_array(h)←
            ),'Bus1');%find bus where house is connected
        for x = 1:length(VBusAll_kV)
            if strcmp(VBusAll_kV(x,1),BusName{2,2})
                V_customer_buses(h,:) = cell2mat(VBusAll_kV(x,2:end));
            end
        end
    end

    PV_pot_array
    sum(PV_pot_array)
    V_customer_buses

    %% ===== Check overloading of line segments ===== %%
    all_currents = mySimulator.GetPosSeqCurrent('line','all','a');
    thermal_limit_violation = 0;
    for i = 2:length(all_currents)
        if strncmpi(cell2mat(all_currents(i,2)),'circb',5)
        elseif strncmpi(cell2mat(all_currents(i,2)),'disc',4)
        else
        this_line = cell2mat(all_currents(i,2));
        this_current_limit = mySimulator.GetParam('line',num2str(←
            this_line),'NormAmps');
            if cell2mat(all_currents(i,3))>str2num(cell2mat(←
                this_current_limit(2,2)))
                this_linesegment = all_currents(i,:)
                    this_current_limit
                thermal_limit_violation = thermal_limit_violation + 1;
            end
        end
    end
    thermal_limit_violation


    %% ===== Plot Results ===== %%
    color = {'b' 'g' 'r' 'c' 'm' 'k' 'y' '-*b' '-*r' '-*c'  '-*m' '-*k' '←
        -*y' '-.b' '-.g' '-.r' '-.c' '-.m' '-.k' '-.y' };

    for h=2:nr_customers
        figure(1)
        title('Voltages at Connection Point for Buildings')
        xlabel('\fontsize{12}Number of Buildings with roof-top PV');
        ylabel('\fontsize{12}Voltage [V]');
        set(gca,'xtick',[0:1:(nr_customers-1)])
        x=[0:1:(nr_customers-1)];
        plot(x, V_customer_buses(h,:)*1000, color{h}, 'LineWidth',1); hold←
            on;
        legend('Building A','Building B', 'Building C', 'Building D', '←
            Building E', 'Building F', 'Building G', 'Building H', '←
            Building I', 'Building J', 'Building K','Building L', '←
            Building M', 'Building N', 'Building O', 'Building P','←
```

```matlab
            Building Q', 'Building R','Building S','Location', '↵
            NorthEastOutside');
  end

figure (2)
plot(x,Q,'-c*'); hold on;
plot(x,P, '-b*'); hold on;
plot(x,sqrt((P.^2)+(Q.^2)),'-.r+'); hold on;
%refline(0,630);
title('Power at Transformer 23.')
xlabel('\fontsize{12}Number of Buildings with roof-top PV');
ylabel('\fontsize{12}Power [kVA,kW,kVAr]');
set(gca,'xtick',[0:1:nr_customers])
x=[0:1:(nr_customers)];
legend('Q','P','S', 'Location', 'NorthEastOutside')
```

Appendix D

MatLab Script for Power Flow Simulations

```matlab
%% Code to perform power flow of Monte Carlo Simulation Results
%% Author: Cecilie S. Arnemo
%% Last update: 01.06.14

clc; clear all;  format compact;
RootPath = 'M:\GaP_simulations\'; %for remote desktop at NTNU
cd([RootPath]);
addpath([RootPath 'GaP_matlabscripts']);
addpath([RootPath 'GaP_dss']);

% Network data initialisation
NetworkFilename = ([RootPath 'GaP_dss\GW_GAP_MS_20131128.dss']);
mySimulator = NetSim(NetworkFilename);
% Set voltage bases
mySimulator.SetVoltBase('ALL',[0.4 2.1 10 20 110]);
mySimulator.SetParam('load','ALL','Vminpu',0);
mySimulator.SetParam('load','ALL','Vmaxpu',5);
mySimulator.SetParam('load','ALL','model',1);
mySimulator.SetParam('Generator','ALL','Vminpu',0);
mySimulator.SetParam('Generator','ALL','Vmaxpu',5);
mySimulator.SetParam('Generator','ALL','model',1);
mySimulator.SolveNetwork;

V_pu_original=mySimulator.GetBusPosSeqVolt('all','pu');

%% Load excel file with thermal limit%%
[cable_nr, cable_char] = xlsread('Excel_files\cablelimits');
cable_types = cable_char(:,1);
cable_limits = cable_char(:,8);

%% Load excel file with Monte Carlo Simulation Results%%%
filename = ['Excel_files\Sim_results\PV_MC_1000sim_all_ONTs_HA'];
xlrange_HA_id = 'A1:A7391';
xlrange_ONT_id = 'B1:B7391';
xlrange_PV_all = 'C1:ALN7391';
HA_id = xlsread(filename,1,xlrange_HA_id);
ONT_id = xlsread(filename,1,xlrange_ONT_id);
PV_all = xlsread(filename,1,xlrange_PV_all);
ONT_id_SUM = xlsread(filename,2);
NrHouses = length(HA_id);
ONTs = size(ONT_id_SUM);
NrONTs = ONTs(1);
%% Number of network buses
NrBus = mySimulator.NrBuses;
AllLoads =  mySimulator.GetList('Load','all');
load_and_volt = mySimulator.GetParam('Load','all','kV');
```

```matlab
NrLoads = length(AllLoads);
Pload = 0.555;
%% === ASSIGN VALUES TO LOADS === %%
pf = 0.95;
for i = 1:NrLoads
if strcmp(load_and_volt(i,2),'0.4') %%%check if load is a LV load
    mySimulator.SetParam('Load',cell2mat(load_and_volt(i,1)),'kW', ←
        Pload);
    mySimulator.SetParam('Load',cell2mat(load_and_volt(i,1)),'pf', pf)←
        ;
    mySimulator.SetParam('Load',cell2mat(load_and_volt(i,1)),'kvar', ←
        Pload*tan(acos(pf)))
elseif strcmp(load_and_volt(i,2),'10') %%%check if load is a ONT
    noOfThisTransformer = str2double(load_and_volt(i,1));
    ix_row_nr = find(ONT_id_SUM(:,1)==noOfThisTransformer);%%%look up←
        the ONT in MCS excel file
    noOfHousesThisTransformer = ONT_id_SUM(ix_row_nr,2); %%%%find ←
        number of houses per ONT
    mySimulator.SetParam('Load',cell2mat(load_and_volt(i,1)),'kW',←
        noOfHousesThisTransformer*Pload);
    mySimulator.SetParam('Load',cell2mat(load_and_volt(i,1)),'pf', pf)←
        ;
    mySimulator.SetParam('Load',cell2mat(load_and_volt(i,1)),'kvar', ←
        noOfHousesThisTransformer*Pload*tan(acos(pf)));
end
end

AllLoads =  mySimulator.GetList('Load','all');
%% === Add generators to buses === %%
s = 1;
a = 0;
for i = 1:NrLoads
thisLoad = AllLoads(i);
    for j = 1:NrHouses
     if strcmpi(AllLoads(i),mat2str(HA_id(j))) %%check if load is a ←
        house
            a=a+1;
%% === Assign generator to houses === %%
            BusName = mySimulator.GetParam('Load',mat2str(HA_id(j)),'←
                Bus1');%find bus where house is connected
            mySimulator.AddElem('Generator',mat2str(HA_id(j)),'bus1',←
                BusName{2,2},'kW',PV_all(j,s),'kvar',0);
        loads_with_HAid(a)=thisLoad;
     end %if strcmp
    end %for j = 1:NrHouses
         for k = 1:NrONTs
            if strcmpi(AllLoads(i),mat2str(ONT_id_SUM(k,1))) %%←
                check if load is a transformer
%% === Assign total generators to transformer === %%
                a=a+1;
                BusName = mySimulator.GetParam('Load',mat2str(←
                    ONT_id_SUM(k,1)),'Bus1'); %find the bus where ←
                    transformer is connected
                mySimulator.AddElem('Generator',mat2str(ONT_id_SUM(k←
                    ,1)),'bus1',BusName{2,2},'kW',ONT_id_SUM(k,s+3),'←
                    kvar',0);
                ONTS_with_gen(a) = thisLoad;
            end
         end %for k
 end %for i =1:NrLoads

%% === Assign PV from Monte Carlo to generators === %%
nr_sim = 1000;
for s = 1:nr_sim
    s
 for i = 1:length(loads_with_HAid)%NrLoads
    for j = 1:NrHouses
     if strcmpi(loads_with_HAid(i),mat2str(HA_id(j))) %%check if load←
        is a house
%% === Assign PV to house === %%
            BusName = mySimulator.GetParam('Load',mat2str(HA_id(j)),'←
                Bus1');%find bus where house is connected
            mySimulator.SetParam('Generator',mat2str(HA_id(j)),'kW',←
                PV_all(j,s));
```

```matlab
                mySimulator.SetParam('Generator',mat2str(HA_id(j)),'pf',1);
                mySimulator.SetParam('Generator',mat2str(HA_id(j)),'kvar',0)
                    ;
        end %if strcmp
      end %for j = 1:NrHouses
  end
  for i = 1:length(ONTS_with_gen)%NrLoads

            for k = 1:NrONTs
                if strcmpi(ONTS_with_gen(i),mat2str(ONT_id_SUM(k,1))) %
                    %check if load is an ONT transformer
%% === Assign total PV of ONT === %%
                        BusName = mySimulator.GetParam('Load',mat2str(
                            ONT_id_SUM(k,1)),'Bus1'); %find the bus where
                            ONT is connected
                        mySimulator.SetParam('Generator',mat2str(
                            ONT_id_SUM(k,1)),'kW',ONT_id_SUM(k,s+3));
                        mySimulator.SetParam('Generator',mat2str(
                            ONT_id_SUM(k,1)),'pf',1);
                        mySimulator.SetParam('Generator',mat2str(
                            ONT_id_SUM(k,1)),'kvar',0);
                end
            end %for k
  end %for i =1:NrLoads
%% === Solve the circuit with new settings === %%
mySimulator.SetParam('Generator','ALL','Vminpu',0);
mySimulator.SetParam('Generator','ALL','Vmaxpu',5);
mySimulator.SetParam('Generator','ALL','model',1);
mySimulator.DSSObj.Text.command = ('solve mode=snap');
mySimulator.SolveNetwork;
%% === Save Results === %%
    %Save Bus Voltages
      V_temp = mySimulator.GetBusPosSeqVolt('ALL','pu');
      VBusAllpu(:,1) = V_temp(2:end,1);
      VBusAllpu(:,s+1) = (V_temp(2:end,2));
      temp_kV = mySimulator.GetBusPosSeqVolt('all','kV');
      VBusAll_kV(:,1) = temp_kV(2:end,1);
      VBusAll_kV(:,s+1) = temp_kV(2:end,2);

%% === Check transformer overloading === %%
      for i = 1:NrLoads
            for k = 1:NrONTs
                if strcmpi(AllLoads(i),mat2str(ONT_id_SUM(k,1))) %%
                    check if load is an ONT transformer
                    S_temp = mySimulator.GetPosSeqPower('Load',mat2str(
                        ONT_id_SUM(k,1)),'kW');
                    S_trafo(k,1)=ONT_id_SUM(k,1);
                    P_temp = cell2mat(S_temp(2,3));
                    Q_temp = cell2mat(S_temp(2,4));
                    S_trafo(k,s+1)=sqrt((P_temp^2)+(Q_temp^2));
                end
            end %for k
      end
%% === Check thermal limit violation === %%
      all_currents = mySimulator.GetPosSeqCurrent('line','all','a');
      thermal_limit_violation = 0;
  for i = 2:length(all_currents)
      if strncmpi(cell2mat(all_currents(i,2)),'circb',5)
      elseif strncmpi(cell2mat(all_currents(i,2)),'disc',4)
      else
      this_line = cell2mat(all_currents(i,2));
      this_current_limit = mySimulator.GetParam('line',num2str(
          this_line),'NormAmps');
          if cell2mat(all_currents(i,3))>str2num(cell2mat(
              this_current_limit(2,2)))
              this_linesegment = all_currents(i,:)
                this_current_limit
              thermal_limit_violation = thermal_limit_violation + 1;
          end
      end
  end
number_of_TLV(s) = thermal_limit_violation;
end %for s=1:nr_sim
```

```matlab
%% === Postprocessing results of power flow === %%
 filename = ['Excel_files\Sim_results\LF\LLMG\VBusAll_kV_',num2str(↩
     nr_sim),'pf',num2str(pf),'.xlsx'];
 xlswrite(filename,VBusAll_kV,1);

%% === Transformator Overloading Considerations === %%
transformator_rating = xlsread('Excel_files\TransformatorRatings');
for s= 1:nr_sim
overload = 0;
for i = 1:length(S_trafo)
    for j = 1:length(transformator_rating)
        if S_trafo(i,1) == transformator_rating(j,1)
            if S_trafo(i,s+1)>transformator_rating(j,3)
                overload = overload + 1;
                S_trafo(i,:)
                transformator_rating(j,:)
            end
        end
    end
end
overloaded_trafos(s) = overload;
end

 figure(4)
 bar(overloaded_trafos,'m');
 title(['Overloaded Transformers for Semi-Fast Charging Scenario. ↩
     Total number of transformers: ',num2str(length(↩
     transformator_rating))])
 xlabel('\fontsize{12}Simulation number');
 ylabel('\fontsize{12}Number of Overloaded Transformers');
 set(gca,'xtick',[1:9:nr_sim])

%% ========= Current Considerations ========= %%

 nr_lines = length(all_currents);
 figure(3)
 bar(number_of_TLV,'g');
 title(['Overloaded Lines for Summer Day Scenario. Total number of ↩
     lines: ',num2str(nr_lines)])
 xlabel('\fontsize{12}Simulation Number');
 ylabel('\fontsize{12}Number of Lines');
 set(gca,'xtick',[1:9:nr_sim])

%% ========= Voltage Considerations ========= %%
 V_pu = cell2mat(VBusAllpu(:,2:end));
 V_kV = cell2mat(VBusAll_kV(:,2:end));

%% Count the number of overvoltages in LV grid and MV grid
size_V_kV = size(V_kV);
nr_buses = size_V_kV(1);
nr_simulations_performed = size_V_kV(2);
 for s=1:nr_simulations_performed
     nr_lv = 0;
     nr_mv = 0;
     for h=1:nr_buses
         if V_kV(h,s)< 0.6 &&  V_kV(h,s)> 0.4
             nr_lv = nr_lv+1;
         end
         if V_kV(h,s)< 15 &&  V_kV(h,s)> 10
             nr_mv = nr_mv+1;
         end
     end
     nr_LVov_sim(s) = nr_lv;
     nr_MVov_sim(s) = nr_mv;
 end
 nr_LVov_sim;
 nr_MVov_sim;
%% plot histogram of the number of overvoltages per simulation

MyMonteCarlo = MonteCarlo();
MyMonteCarloInput.nr_LVov_sim = nr_LVov_sim;
MyMonteCarloInput.nr_MVov_sim = nr_MVov_sim;
MyMonteCarloInput.nr_sim = nr_sim;
MyMonteCarlo.trafo_histogram(MyMonteCarloInput)
```

```matlab
%% find voltage values that are above 1.0 pu %%
 [ix,iy] = find(V_pu(:,:)>1.0);
 ix_last = ix(length(ix));
 ix_nr_rows = find(ix(:) == ix_last);
 ix_rows = ix_nr_rows(1);
 overvoltages_pu = zeros(ix_rows,nr_sim);
 overvoltages_kV = zeros(ix_rows,nr_sim);
 for i=1:ix_rows
     thisBus = ix(i);
         overvoltages_names(i,1) = VBusAllpu(thisBus,1);
         overvoltages_pu(i,:) = V_pu(thisBus,:);
         overvoltages_kV(i,:) = V_kV(thisBus,:);
 end

 [ix2] = find(overvoltages_kV(:,1)<1.0); %find LV buses
 LV_buses = zeros(length(ix2),nr_sim);
 for i=1:length(ix2)
     thisBus = ix2(i);
      LV_buses(i,:) = overvoltages_kV(thisBus,:);
 end
  [ix3] = find(overvoltages_kV(:,1)>8.0 & overvoltages_kV(:,1)<15); %↩
      find MV buses
 MV_buses = zeros(length(ix3),nr_sim);
 for i=1:length(ix3)
     thisBus = ix3(i);
      MV_buses(i,:) = overvoltages_kV(thisBus,:);
 end

for i=1:length(LV_buses(:,1))
    figure (1)
    title('LV buses with V > 0.4 kV')
    xlabel('\fontsize{12}Simulation number');
    ylabel('\fontsize{12}Voltage [kV]');
    plot(LV_buses(i,:), 'r', 'LineWidth',1); hold on;

end
figure(1)
refline([0 0.412]);

for i=1:length(MV_buses(:,1))
    figure (2)
    title('MV buses with V > 10 kV')
    xlabel('\fontsize{12}Simulation number');
    ylabel('\fontsize{12}Voltage [kV]');
    plot(MV_buses(i,:), 'r', 'LineWidth',1); hold on;
end
```