Optimal trading strategy for storage systems

Snel, A.C.

Universiteit Leiden
Faculteit der Wiskunde en Natuurwetenschappen
Mathematisch Instituut Leiden

Optimal trading strategy for storage systems

A thesis submitted to the
Leiden Institute for Mathematics
in partial fulfillment of the requirements

for the degree

MASTER OF SCIENCE
in
MATHEMATICS and EDUCATION

by

Arjan Coenraad Snel
Thesis advisor: Dr. F.M. Spieksma

Leiden, the Netherlands
August 24, 2010

Mathematisch Instituut, Universiteit Leiden

MSc THESIS MATHEMATICS

“Optimal trading strategy for storage systems”

Arjan Coenraad Snel

University of Leiden

Chairman

Dr. F.M. Spieksma
(University of Leiden)

Responsible professor

Prof.dr.ir. J.K.I. Aardal
(TU Delft)

Daily supervisor

Ir. G.A. Bloemhof
(KEMA)

Committee member

Dr. D.C. Gijswijt
(University of Leiden)

August 2010

Arnhem, the Netherlands

Preface

This thesis describes the developments and the outcome of the research I did during my internship at KEMA Consulting Arnhem for my study Mathematics at the University of Leiden. The assignment was to develop a trading strategy that obtains a maximum profit for an electricity storage system used for buying and selling energy in the electrical grid.

During my internship of nine months, I have been working on a problem that is related to a project KEMA is involved in, the GROW-DERS project. At KEMA I got the opportunity to join in the meetings for this project, to learn from these meetings and to talk about the progress I made. These meetings were very valuable for me. During my internship, I attended an international meeting about the opportunities and challenges arising from developments in the power grid. It was very interesting to see how people with different interests gave a different picture of the future power grid. While some see opportunities, others see problems. Together with my supervisor at KEMA, Gabriël Bloemhof, I visited the Study Group Mathematics with Industry 2010 held at the Centrum Wiskunde & Informatica in Amsterdam. It was very instructive to see how the participants tried to tackle the tasks. Besides these special activities I learned very much from all the people I met at KEMA.

The helpfulness of all colleagues at KEMA creates a very pleasant working atmosphere. I want to thank all colleagues for the warm welcome I received, for their help and enthusiasm, and of course for the nice lunches and the coffee meetings. I want to thank Dieter, Roger, Sonja and Gabriël for the warm welcome that I received in the GROW-DERS project and for consultation during the meetings. I would like to thank my daily supervisor Gabriël Bloemhof for giving me the opportunity to do my research project at KEMA, for the help, the support and the suggestions for my research and for the comments to improve my thesis. I want to thank my supervisor Karen Aardal for the help, the support and the suggestions during our meetings in Delft, Utrecht and Arnhem, and for the comments to improve my thesis. I want to thank Floske Spieksma for the support and the suggestions during our meetings in Leiden, and for compiling the reading committee. I would like to thank Karen Aardal, Floske Spieksma, Dion Gijswijt and Gabriël Bloemhof for participating in the reading committee and for the suggestions and the comments to improve my thesis.

I would like to thank the ”Wisko’s” for making my study unforgettable. Especially, I want to thank Marco for test-reading my thesis.

In the last 28 years there have been many people that have inspired me and kept me motivated. While not everyone will have been aware of this, I want to thank you all.

Arjan Snel
Arnhem August 2010

Abstract

This thesis describes the result and the process of a research to determine an optimal trading strategy for storage systems in the low voltage grid. To give a clear insight in the problem and in the algorithms to solve the problem, a phased approach is used. First a simple model of a storage system is described, that is extended in three steps to the final model that is a realistic model of a storage system. All four models are described and for each of these models an algorithm is developed to determine an optimal trading strategy. In these models the energy prices per quarter of an hour are given in advance for 24 hours, the model is discrete in time. We use n intervals of a quarter of an hour in which the storage system can charge energy, discharge energy or do nothing. We assume that there is no residual value of energy. Though the problem solved is a normal LP problem, the phased approach and the description of the problem and the algorithm give insight in the solution that is required.

In the first model, Model A, the state of charge of the storage system of interval i, $SOC(i)$, is either full or empty, this can be naturally modeled as a binary integer problem. Algorithm 1 is developed to determine an optimal trading strategy as described above. Algorithm 1 has complexity $O(n)$.

In Model B the charge capacity, the discharge capacity and the capacity of the storage system can have different values. With three different values for these physical constrains, the SOC cannot be modeled as a discrete model and thus the SOC is modeled as a continuous model. Algorithm 2 is developed to determine an optimal trading strategy for Model B as described above, Algorithm 2 has complexity $O(n^2)$.

As an extension to Model B, in Model C energy losses from using the storage system are taken into account. There is energy required for charging and for discharging the storage system. This is energy that cannot be used for trading. Also, in time the energy in the storage system decreases, this is energy that cannot be sold. The energy that cannot be sold constitutes a loss from using the storage system. To take the losses into account, there are two virtual energy prices developed, the virtual charge price and the virtual discharge price. Similar to the previous model, the maximum amount of energy to trade can be determined, using the new determined $SOC(i)$. Algorithm 3 is developed to determine an optimal trading strategy for Model C, as described above. This Algorithm has complexity $O(n^2)$.

In the final model, Model D, there are bounds included in the model. With these bounds it is possible to use the storage system for trading as well as for solving problems in the low voltage grid. To solve problems in the low voltage grid, space to store too much energy that is in the low voltage grid is required. It is also possible that not all the energy demanded can be transported, for instance because of the capacity of the network. If there is a storage system nearby the problem it is possible that the energy available in the storage system can help to overcome the

problem. For such a problem, the storage system is used to supply energy. To be able to help overcome both types of problems, there is a lower and an upper bound required. With these bounds, there is less storage space available for trading. To be able to solve problems in the low voltage grid, every interval must have a SOC within the bounds. Algorithm 4 is developed to determine an optimal trading strategy for Model D. While the complexity of Algorithm 2 and 3 is $O(n^2)$, the complexity of Algorithm 4 is $O(n^3)$.

To reduce the complexity, a greedy algorithm is developed. For every iteration i, interval i is first used to discharge the maximum amount of energy that is possible with respect to the discharge capacity. After this, the minimum amount of energy must be charged to get the $SOC(i)$ equal to the LB. The absolute local minimum before interval i is used to charge energy for minimum cost. This is done for all intervals, and gives an optimal trading strategy. Algorithm 5 determines an optimal trading strategy for Model D with complexity $O(n^2)$.

For KEMA the program ATMP[1] is developed. The code of the algorithms that are used in ATMP are written in Visual Basic Application of Excel. Therefore these algorithms can be used by KEMA for the overal program PLATOS[2]. ATMP is used to give clear insight in the algorithms developed and the output of the algorithms is processed graphically. The user can even try to develop a trading strategy that is better than the trading strategy developed by ATMP. This helps the user to get a good insight in the problem and trust in the solution. In the Appendices the algorithms used are described.

[1] Algorithm for Trading with Maximum Profit

[2] PLAnning Tool for Optimal Storage

Extended summary

This thesis describes the result and the process of a research to determine an optimal trading strategy for storage systems in the low voltage grid. To give a clear insight in the problem and in the algorithms to solve the problem, a phased approach is used. First a simple model of a storage system is described, that is extended in three steps to the final model that is a realistic model of a storage system. All four models are described and for each of these models an algorithm is developed to determine an optimal trading strategy. For these models, the energy prices per quarter of an hour are assumed to be given in advance for 24 hours, the model is discrete in time. We assume that there is no residual value of energy. In these models the storage system can charge energy, discharge energy or do nothing during each of the n intervals. For charging energy the energy price of the relevant interval is the cost and for discharging energy the energy price of the relevant interval is the profit. The algorithm is developed to determine a strategy to obtain the maximum profit.Though the problem solved is a normal LP problem, the phased approach and the description of the problem and the algorithm give insight in the solution that is required.

In the first model, Model A, the state of charge of the storage system of interval i, $SOC(i)$, is either full or empty. Therefore we can make the following decisions during one interval: we can buy energy, sell energy, or do nothing. We cannot sell energy if the SOC is empty, and we cannot buy energy when the SOC is full. This is a simple description of a storage system, that is used to show that it is optimal to charge energy in interval i, if the $SOC(i)$ is empty and the energy price of interval $i+1$, $p(i+1)$ is larger than $p(i)$. For discharging, it is optimal to discharge in interval j, if the $SOC(j)$ is full and the energy price of interval $j+1$, $p(j+1)$ is smaller than $p(j)$. Algorithm 1 is developed to determine an optimal trading strategy as described above. Algorithm 1 has complexity $O(n)$.

In Model B the storage system can have three different values for the charge capacity, the discharge capacity and the capacity of the storage system. With the capacity of the storage system, C, larger than the charge capacity of the storage system, ChC, it is possible that an optimal solution will not make fully use of the capacity of the storage system for every interval. Therefore the state of charge of the storage system for every interval i, $SOC(i)$, can take any value between zero and C. Also the quantity of energy that is charged during interval i, $ChQ(i)$, can take any value between zero and ChC. Likewise for the discharge capacity of the storage system, $DChC$. When the capacity of the storage system is larger than the absolute value of the discharge capacity of the storage system, the quantity of energy that is discharged during an interval i, $DChQ(i)$, can take any value between $DChC$ and zero. With some small adjustments, the description of the optimal trading strategy for Model A can be applied to Model B. The last local minimum is the last interval, that can be used to charge energy, of a non-increasing period for the energy price, that has a subsequent local maximum. A subsequent local maximum is the last interval, for which the discharge capacity is not fully used, of a non-decreasing period for the energy price after a local minimum. Now it is required

to determine the maximum amount of energy to trade, since this is no longer given by the capacity of the storage system. The capacity available to charge and to discharge is determined, and for all intervals k, between the profitable combination of intervals i and j, to charge and to discharge it is determined if the $SOC(k)$ is not larger than the capacity of the storage system after charging. If for one interval r, the $SOC(r)$ becomes larger than the capacity, the amount of energy to charge is decreased to the maximum amount of energy that can be charged to get the $SOC(r)$ equal to the capacity of the storage system. Algorithm 2 is developed to determine an optimal trading strategy for Model B as described above, Algorithm 2 has complexity $O(n^2)$.

As an extension to Model B, in Model C energy losses from using the storage system are taken into account. There is energy required for charging and for discharging the storage system. This is energy that cannot be used for trading. Also, in time the energy in the storage system decreases, this is energy that cannot be sold. The energy that cannot be sold are losses from using the storage system. With these losses taken into account, the model becomes more realistic. To be able to find the last local minimum and the subsequent local maximum, the costs for the losses suffered must be taken into account. The energy price of every interval can be recalculated by taking the losses caused by storage into account. If there are two intervals, interval r and interval $r + k$, with energy price of interval r, $p(r) = p(r + k)$, because of the losses caused by storage, energy charged in interval r would decrease in the time k, and thus this energy is more expensive since we have in interval $r + k$ less energy left. To use this to determine the last local minimum and the subsequent local maximum, a virtual energy price is determined for every interval. The virtual energy price of interval i, $vp(i)$, is the original energy price, $p(i)$, multiplied with the residual after the losses caused by storage, $RLBS$, to the power i, as given in expression (1).

$$vp(i) := p(i) \cdot RLBS^i \tag{1}$$

Also the losses caused by (dis)charging must be taken into account. These losses are taken into account in the virtual price, which let to two virtual prices for every interval. For charging energy, the virtual charging price is determined, and for discharging energy, the virtual discharge price is determined. These prices are determined to be able to calculate if it is profitable to use a combination of intervals for trading. Since energy is lost by charging, the virtual charge price of energy must be larger than the virtual price already determined. The virtual charging price of interval i, $cp(i)$, is therefore determined as the virtual price of interval i, $vp(i)$, devided by the residual after losses caused by charging, $RLBC$, as in expression (2).

$$cp(i) := \frac{p(i) \cdot RLBS^i}{RLBC} \tag{2}$$

The losses caused by discharging also have effect on the profit. Energy that is lost because of discharging is energy that cannot be sold. Therefore the virtual discharging price is smaller than the already determined virtual price. The virtual discharging price of interval j, $dcp(j)$ is therefore determined as the virtual price of interval j, $vp(j)$, multiplied with the residual after losses caused by discharging, $RLBDC$, as in expression (3).

$$dcp(j) := p(j) \cdot RLBS^j \cdot RLBDC \tag{3}$$

The charging and discharging prices that are determined for all intervals, can be used to determine the last local minimum and the subsequent local maximum. First the last local minimum is determined using the charging price. If a last local minimum is determined, the subsequent local maximum is determined by the discharging price. While in the previous models it was clear that once a combination was determined, this combination was profitable, now it is possible that there is a combination determined that is not profitable, since there is a charge and a discharge price used. Now a combination is profitable if the charge price of the last local minimum is smaller than the discharge price of the subsequent local maximum. If the combination determined is not profitable, a new subsequent local maximum is determined for the already determined last local minimum. If there is an interval i, for which the charging price is less than the charging price of the last local minimum found, interval i is the new last local minimum for which a subsequent local maximum is determined. Once a profitable combination of a last local minimum and a subsequent local maximum is determined, the amount of energy to trade is to be determined. Therefore it is required to know the $SOC(k)$ for all intervals k between the last local minimum and the subsequent local maximum. The SOC is affected by the losses caused by storage and by the losses caused by charging. The losses caused by discharging is calculated over energy that would be discharged. The losses caused by discharging only affect the profit and not the SOC. Every interval energy is stored in the system, energy is lost. To determine the current SOC, the residual after losses caused by storage is multiplied with the SOC of the previous interval. To take losses by charging into account, the charge quantity of an interval is multiplied with the residual after losses caused by charging. The $SOC(i)$ can be determined by expression (4).

$$SOC(i) := SOC(i-1) \cdot RLBS + ChQ(i) \cdot RLBC + DChQ(i) \tag{4}$$

Similar to the previous model, the maximum amount of energy to trade can be determined, using the new determined $SOC(i)$. Algorithm 3 is developed to determine an optimal trading strategy for Model C, as described above. This Algorithm has complexity $O(n^2)$.

In the final model, Model D, there are bounds included in the model. With these bounds it is possible to use the storage system for trading as well as for solving problems in the low voltage grid. To solve problems in the low voltage grid, space to store too much energy that is in the low voltage grid is required. It is also possible that it is not possible to transport all the energy demanded, for instance because of the capacity of the network. If there is a storage system nearby the problem it is possible that the energy available in the storage system can help to overcome the problem. For such a problem, the storage system is used to supply energy. To be able to help overcome both types of problems, there is a lower and an upper bound required. With these bounds, there is less storage space available for trading. To be able to solve problems in the low voltage grid, every interval must have a SOC within the bounds. It is expected that there is energy in the storage system at the beginning of the period over which an optimal trading strategy is to be determined. With this energy, the SOC of all intervals is determined, using expression (4). It is required to keep the SOC for all intervals smaller or equal to the upper bound, UB. Therefore, for the first interval this is determined. If the SOC is larger than the UB, the amount of energy that is required to be discharged is determined. For the next interval it is determined as well if the SOC is smaller or equal to the UB. Once this is true, all subsequent intervals have a SOC that is smaller or equal to the UB. If for the last interval the $SOC(n)$ is larger than the lower bound, LB, the amount of energy that is in the storage system, that is not required for solving problems, can be discharged in the first local maximum. The first local maximum is the last interval i, that can be used for discharging, of a non-decreasing period for the energy price, with $SOC(i) > LB$. If the $SOC(i)$ is smaller than the lower bound, it is required to charge energy. To charge energy for minimum costs, interval m with the absolute minimum energy price available before interval i, is determined to charge the required amount of energy. Interval m is the absolute local minimum before interval i. For all the intervals the SOC is within the bounds. An optimal trading strategy can be determined that keeps the SOC within the bounds. Since there is energy in the storage system, it is possible to discharge energy, before we charge energy to keep the SOC within the bounds. To be able to determine the best combinations for trading, first the interval to discharge energy is determined and than the interval to charge energy is determined. The interval to discharge energy, is the first local maximum. The first local maximum has a domain in which the interval to charge is determined. The domain starts with the first interval and it ends with the first interval, k, since the first local maximum, that has a SOC that is less or equal to the lower bound. The interval to charge energy, is the absolute local minimum before interval k. The maximum amount of energy to trade can be determined similar to the previous model. Algorithm 4 is developed to determine an optimal trading strategy for Model D. While the complexity of Algorithm 2 and 3 is $O(n^2)$, the complexity of Algorithm 4 is $O(n^3)$.

To reduce the complexity, a greedy algorithm is developed. For every iteration i, interval i is first used to discharge the maximum amount of energy that is possible with respect to the discharge capacity. After this, the minimum amount of energy must be charged to get the $SOC(i)$ equal to the LB. The absolute local minimum before interval i is used to charge energy for minimum cost. This is done for all intervals, and gives an optimal trading strategy. Algorithm 5 determines an optimal trading strategy for Model D with complexity $O(n^2)$.

For KEMA the program ATMP[3], is developed. The code of the algorithms that are used in ATMP are written in Visual Basic Application of Excel. Therefore these algorithms can be used by KEMA for the overal program PLATOS[4]. ATMP is used to give clear insight in the algorithms developed and the output of the algorithms is processed graphically. The user can even try to develop a trading strategy that is better than the trading strategy developed by ATMP. This helps the user to get a good insight in the problem and trust in the solution. In the Appendices the algorithms used are described.

[3] Algorithm for Trading with Maximum Profit

[4] PLAnnig Tool for Optimal Storage

Contents

Introduction

In this thesis the developments and the outcome of the research done for the study Mathematics at the University of Leiden are described. The research is done during an internship at KEMA. The problem to solve is an optimization problem. The assignment was to develop a trading strategy that obtains a maximum profit for an electricity storage system used for buying and selling energy in the electrical grid. The storage system and the electrical grid are subject to physical constraints. KEMA is involved in a project closely connected to this problem, the GROW-DERS project. To get a better insight in the assignment, first an introduction to KEMA, power grids and GROW-DERS is presented.

0.1 Grid developments and storage

KEMA is a multinational company specializing in strategic and technical energy consultancy, operational support, measurements and inspection, and testing and certification. The internship was at KEMA Consulting at the office in Arnhem. For more details we refer the reader to [8].

The electrical grid can be divided in three main levels, the so called high, medium and low voltage grid. The high voltage grid is used for transmission, the medium voltage grid for sub-transmission and the low voltage grid is used for distribution. The research topic is an application related to the low voltage grid. In the current grid, the centralized generators supply energy to the high voltage grid. The high voltage grid will transport energy to the medium voltage grid. The medium voltage grid will transport energy to the low voltage grid and to large industry. The large industry both demands and supplies energy. The low voltage grid will distribute energy to the customers, like households that demand energy. Figure 1 gives an overview of an electrical grid. For more details we refer the reader to [1]. In the last decade some households became suppliers of energy, using small generators as photovoltaics, PV, windmills and combined heat and power, CHP. At the moment the generation by households is on small scale, but in the future growth is expected. With the upcoming decentralized generation of renewable energy by households, the low voltage grid is subject to change. The generation of renewable energy is currently not fully controlled or regulated, and it is expected that demand and supply will not be coordinated, at least not in the same way as the centralized generation. Generation of renewable energy by households can fluctuate, and the centralized generators are not able to react as fast as the change in generation of renewable energy. Therefore the generation of renewable energy by households is not very reliable at the moment because of the fluctuations. Leveling the supply of this energy would make it more useful. For leveling the supply, some sort of storage system could be helpful. This storage system could be charged with energy when there is more energy generated than the average supply, and could be discharged when the energy generated is less than the average. In addition, because of the growing demand of energy by households, the low voltage grid is subject to changes. These changes bring problems for the daily-activities and for the long-term planning of the low voltage grid.

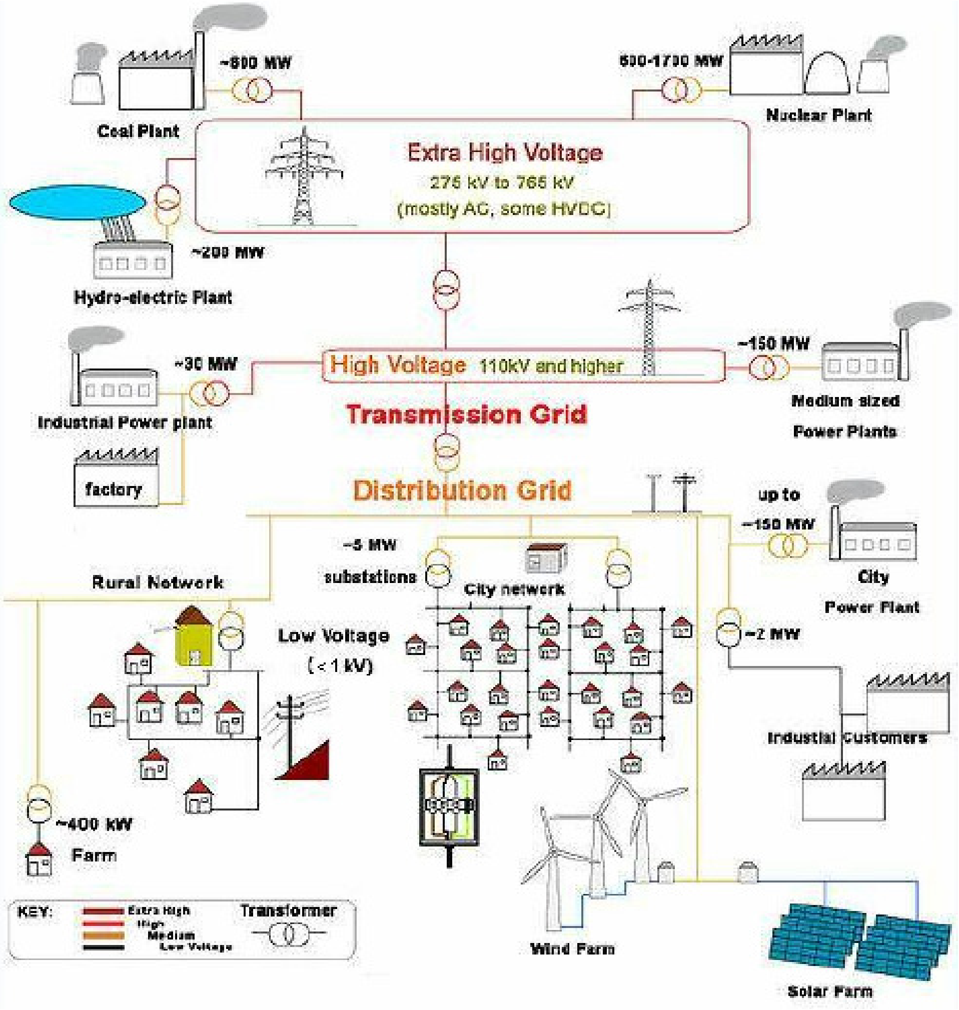


Figure 1: The electrical grid

At the high voltage level, renewable energy generated by great (offshore) wind farms is very fluctuating, which causes problems for processing the energy. If it would be possible to level the supply or to control this, a better result of processing the energy generated by wind farms can be achieved. At the moment, controlling the supply is done by not fully using the capacity of the wind farms. In the future, perhaps it is possible to level the supply by using storage systems. Though the high voltage level is not part of the scope of the assignment, the outcome of this thesis might be useful if storage systems are used for leveling the supply by wind farms.

A storage system can be used in an electrical grid to store energy. A storage system can be charged and discharged and thereby subtract energy from, or add energy to the electrical grid. The amount of energy that can be charged or discharged from the storage system depends on the capacity of the electrical grid, on the charging respectively discharging power of the storage system and on the state of charge of the storage system. There is a difference between the capacity of the storage system and the charge and discharge power of the storage system concerning the units it is expressed in. The capacity of a storage system can be expressed in kWh, which is a unit for energy. The charge and discharge power of a storage system can be expressed in kW, which is a unit for power.

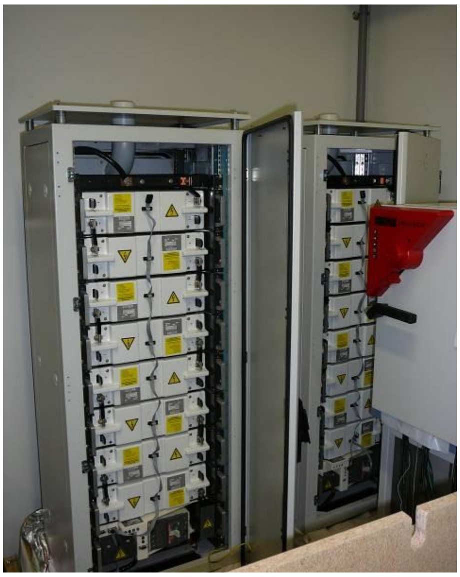


Figure 2: Example of a storage system (Li-ion, from SAFT)

Another application of a storage system is trading energy, with the aim of making a profit. Amsterdam hosts the headquarter of the APX, short for the **A**msterdam **P**ower **Ex**change. There are two markets for trading electricity: the day-ahead market and the intraday market. For the day-ahead market the prices are calculated one day ahead. The market members can submit their orders until a day before it is needed, after which supply and demand are compared, and the prices for each hour of the following day are calculated. For the intraday market, the APX offers market members the opportunity to trade energy in 15 minute intervals, 1 hour blocks, and 2 hour blocks up to two hours prior to delivery. For more details we refer the reader to [7]. Because of the fluctuating prices there is a potential profit by trading energy. For trading energy profitable, energy must be bought at a low energy price and be sold later in time at higher price. In between a storage system can be used to store the energy. Therefore a storage system is set to charge energy at low price and is set to discharge at high price.

The increased use of electric cars is a new challenge and opportunity for the low voltage grid. The energy demand will increase enormously. But when these cars are not used, and are plugged into the electrical grid, the energy stored in the cars, or any free storage space in these cars, could be of great use as well.

0.2 The GROW-DERS project

One of the projects in which KEMA is involved, is the GROW-DERS project. GROW-DERS stands for **g**rid **r**eliability and **o**perability **w**ith **d**istributed g**e**ne**r**ation using flexible **s**torage. Storage systems are the focus for the GROW-DERS project. Next to being used for energy trade, one can use storage systems to solve physical problems in the low voltage grid.

GROW-DERS is an innovative demonstration project that offers a better insight into the possibilities of the use of storage systems in the low voltage grid. GROW-DERS offers operational experience and examines the technical and economical feasibility of some storage systems. To determine the benefit of a storage system, all the possible applications must be looked at. Of course, these storage systems can only help solving non-permanent problems in networks, since a storage system can only act as a buffer; it cannot generate energy. Perhaps storage systems also bring new problems for the network, as for instance when the storage system has a trading strategy that is in conflict with the constraints of the network. By contributing to solving problems in the network, a storage system can save costs by postponing, or even preventing investments. With the possibility to store energy it might also be possible to make a profit by trading energy using the storage system as a depot. For more details we refer the reader to [6].

KEMA developed the PLATOS model for the GROW-DERS project, PLATOS stands for **pla**nning **t**ool for **o**ptimising **s**torage. The PLATOS model is a simulation program that models the network. With the PLATOS model KEMA is able to give a clear view of the benefits of the storage system used in the network. The PLATOS model is developed to determine the best locations for storage systems in the low voltage grid. Besides these benefits, a good strategy needs to be developed to determine the optimal profit obtained by adding storage systems into the network. One of the partners of the GROW-DERS project developed a simple program to use a storage system for trading. This simple program was developed to give an insight in what is needed to make a more realistic model of a storage system. This program was processed by KEMA in the PLATOS model of the network with storage systems included.

To get a good insight into the possibilities for storage systems in the low voltage grid, a program that can determine the optimal trading strategy is needed. This is the topic of the assignment. This problem applies to the daily activities in the low voltage grid.

0.3 Problem description and approach

The problem considered in this thesis is an optimization problem, namely to develop an optimal trading algorithm for an electricity storage system. The solution to the problem must meet the physical constraints of the low voltage grid. It was desirable to start with a simple model and to make it more realistic using several steps. This phased approach gives KEMA the insight they want into all the intermediate results. The phased approach leads in this thesis to several models, called A, B, C and D. For these models there are algorithms developed to determine an optimal trading strategy to obtain a maximum profit, called Algorithm 1, 2, 3, 4 and 5 with som heuristics and sub algorithms. For the models, we assume that energy prices per quarter of an hour are given in advance for 24 hours by the APX. The model is discrete in time, for which we use a time-step of fifteen minutes. We assume that there is no residual value for energy. For all four models these assumptions are similair. Since there are four models, there must be differences between the models as well. Differences between the models are wether the state of charge SOC is a boolean or not. If the SOC is either empty or full, the SOC is like a boolean, while if the SOC can take any value between empty and full, the SOC is not a boolean. Energy losses for using the storage system are not taken into account in the first two models. To be able to use the storage system for other purposes besides trading, for the last model there are bounds for the SOC taken into account. The similarities and the differences between the models are shown in Table 1.

Similairities	Differences
Prices are given	SOC as boolean or not
Discrete model	Energy losses taken into account or not
No residual value for energy	Bounds for the SOC taken into account or not

Table 1: Similarities and differences between the models

The assignment is formulated as follows;

The assignment is to develop a practical and mathematically correct algorithm that gives an optimal trading strategy for an electricity storage system. The solution to this problem must meet the physical constraints of the low voltage grid. It is desired to start with a simple model and make it more realistic using several modeling steps.

In the final model the trading strategy must take into account that a storage system has a power to charge, a power to discharge and a capacity to store energy. There will be energy losses from using the storage system. The storage system can be used for trading as well as for solving problems in the low voltage grid. The phased approach gives KEMA the insight they want in all the intermediate results.

For the phased approach, the models used are described in Table 2.

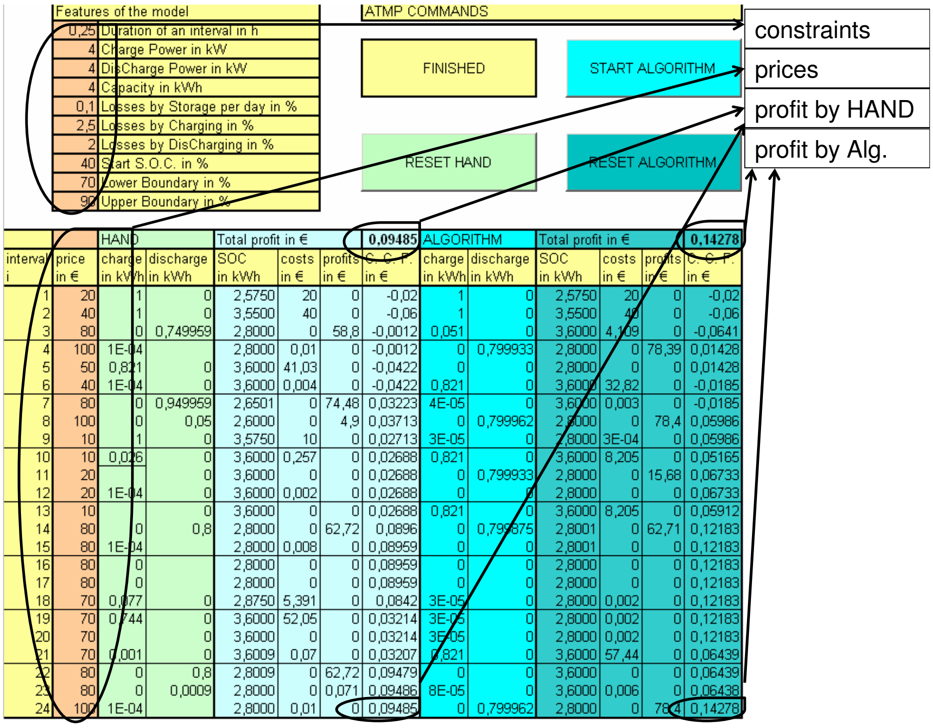
Characterization	Model A	Model B	Model C	Model D
SOC as boolean	yes	no	no	no
Energy losses taken into account	no	no	yes	yes
Bounds for the SOC taken into account	no	no	no	yes

Table 2: Description of the models

With the diverse audience, mathematicians at the university and engineers at KEMA and grid companies, the report is a mix of theory with models, algorithms and proofs, and practical examples with graphs and explanations.

0.4 Outline of the thesis

During the research, an algorithm is developed that, for a given period of time, with the energy prices given, will determine an optimal trading strategy for a given storage system to obtain a maximum profit. This algorithm is used in a program called ATMP, **A**lgorithm for **T**rading with **M**aximum **P**rofit. ATMP is developed to use the algorithm and to clarify how the algorithm works. The output of the algorithm is processed into graphs that give a clear overview of the strategy. It is desirable that the storage system can be used for other purposes besides trading. Therefore some constraints are added to the problem to be able to use the storage system for solving problems in the electrical grid, as well as for trading. To show how the user can chose what constraints must be taken into account, the frontpage of ATMP is given in Figure 3.

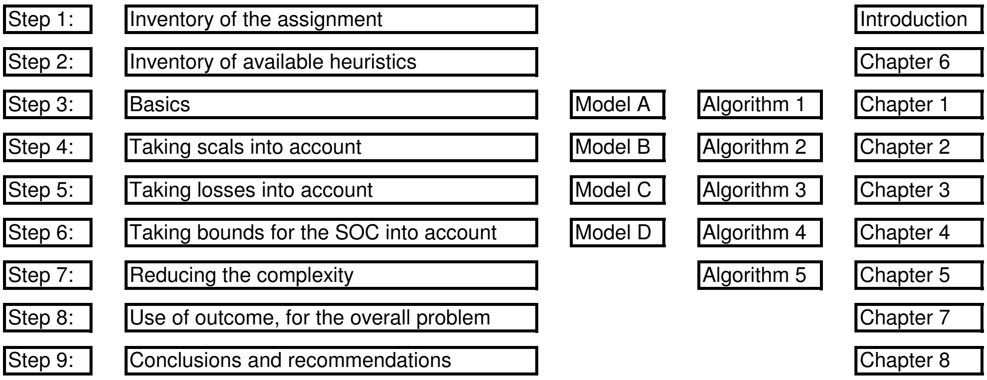


Features of the model	
0,25	Duration of an interval in h
4	Charge Power in kW
4	DisCharge Power in kW
4	Capacity in kWh
0,1	Losses by Storage per day in %
2,5	Losses by Charging in %
2	Losses by DisCharging in %
40	Start S.O.C. in %
70	Lower Boundary in %
90	Upper Boundary in %

		HAND		Total profit in €			**0,09485**	ALGORITHM		Total profit in €			**0,14278**
interval i	price in €	charge in kWh	discharge in kWh	SOC in kWh	costs in €	profits in €	C. C. F. in €	charge in kWh	discharge in kWh	SOC in kWh	costs in €	profits in €	C. C. F. in €
1	20	1	0	2,5750	20	0	-0,02	1	0	2,5750	20	0	-0,02
2	40	1	0	3,5500	40	0	-0,06	1	0	3,5500	40	0	-0,06
3	80	0	0,749959	2,8000	0	58,8	-0,0012	0,051	0	3,6000	4,109	0	-0,0641
4	100	1E-04		2,8000	0,01	0	-0,0012	0	0,799933	2,8000	0	78,39	0,01428
5	50	0,821	0	3,6000	41,03	0	-0,0422	0	0	2,8000	0	0	0,01428
6	40	1E-04	0	3,6000	0,004	0	-0,0422	0,821	0	3,6000	32,82	0	-0,0185
7	80	0	0,949959	2,6501	0	74,48	0,03223	4E-05	0	3,6000	0,003	0	-0,0185
8	100	0	0,05	2,6000	0	4,9	0,03713	0	0,799962	2,8000	0	78,4	0,05986
9	10	1	0	3,5750	10	0	0,02713	3E-05	0	2,8000	3E-04	0	0,05986
10	10	0,026	0	3,6000	0,257	0	0,02688	0,821	0	3,6000	8,205	0	0,05165
11	20		0	3,6000	0	0	0,02688	0	0,799933	2,8000	0	15,68	0,06733
12	20	1E-04	0	3,6000	0,002	0	0,02688	0	0	2,8000	0	0	0,06733
13	10		0	3,6000	0	0	0,02688	0,821	0	3,6000	8,205	0	0,05912
14	80	0	0,8	2,8000	0	62,72	0,0896	0	0,799875	2,8001	0	62,71	0,12183
15	80	1E-04		2,8000	0,008	0	0,08959	0	0	2,8001	0	0	0,12183
16	80	0		2,8000	0	0	0,08959	0	0	2,8000	0	0	0,12183
17	80	0		2,8000	0	0	0,08959	0	0	2,8000	0	0	0,12183
18	70	0,077	0	2,8750	5,391	0	0,0842	3E-05	0	2,8000	0,002	0	0,12183
19	70	0,744	0	3,6000	52,05	0	0,03214	3E-05	0	2,8000	0,002	0	0,12183
20	70		0	3,6000	0	0	0,03214	3E-05	0	2,8000	0,002	0	0,12183
21	70	0,001	0	3,6009	0,07	0	0,03207	0,821	0	3,6000	57,44	0	0,06439
22	80	0	0,8	2,8009	0	62,72	0,09479	0	0	3,6000	0	0	0,06439
23	80	0	0,0009	2,8000	0	0,071	0,09486	8E-05	0	3,6000	0,006	0	0,06438
24	100	1E-04		2,8000	0,01	0	0,09485	0	0,799962	2,8000	0	78,4	0,14278

Figure 3: The front page of ATMP

In Chapters 1-3 Model A, Model B and Model C will be described that are used to find a new algorithm to give an optimal trading strategy. In Chapter 4 the final program to find an optimal trading strategy will be described. Chapter 5 describes the reduction of the complexity for the final algorithm. In Chapter 6 some other heuristics to give an optimal trading strategy for a storage system are described. Chapter 7 describes the overall problem and how the outcome of the research can be of use for this problem. The conclusions and recommendations are given in Chapter 8. An overview of this thesis is given in Figure 4.



Step 1:	Inventory of the assignment			Introduction
Step 2:	Inventory of available heuristics			Chapter 6
Step 3:	Basics	Model A	Algorithm 1	Chapter 1
Step 4:	Taking scals into account	Model B	Algorithm 2	Chapter 2
Step 5:	Taking losses into account	Model C	Algorithm 3	Chapter 3
Step 6:	Taking bounds for the SOC into account	Model D	Algorithm 4	Chapter 4
Step 7:	Reducing the complexity		Algorithm 5	Chapter 5
Step 8:	Use of outcome, for the overall problem			Chapter 7
Step 9:	Conclusions and recommendations			Chapter 8

Figure 4: Overview

Chapter 1

Algorithm 1: The basics

In this chapter a simple model of a storage system will be described. Here the basics of an optimal trading strategy will be explained and proven. In Section 1.3 the overall approach to determine an optimal trading strategy to obtain a maximum profit for Model A is described.

1.1 Model of the storage system

In the first model, Model A, the energy prices per quarter of an hour are supposed to be given in advance for 24 hours. The model is discrete, for which we use a time-step of fifteen minutes. We assume that there is no residual value of energy. These are assumptions made for the models in general. For Model A, the state of charge of the storage system, SOC, is either full or empty. Therefore we can make the following decisions during one interval: we can buy energy, sell energy, or do nothing. We cannot sell energy if the SOC is empty, and we cannot buy energy when the SOC is full. The decision in interval i determines the situation for the next interval $i+1$. Selling energy in interval i gives a profit of $p(i)$, and buying energy in interval i gives costs of $p(i)$ which can be looked at as a profit of $-p(i)$ as is shown in Figure 1.1.

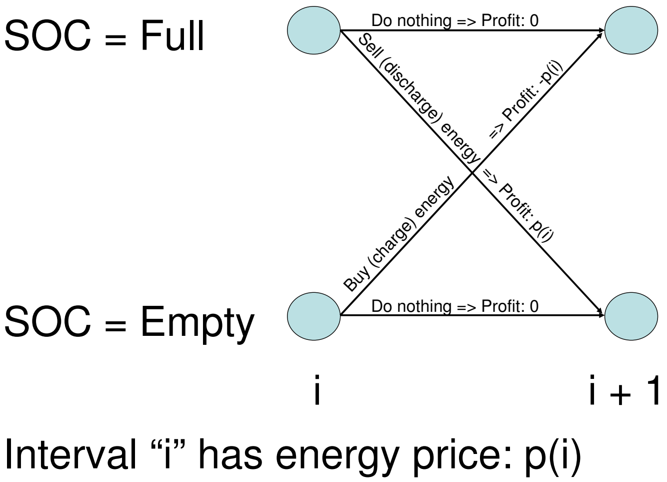


Figure 1.1: Costs and profit by charging and discharging

To develop the optimal trading strategy, using the storage system for trading, we can describe the problem as a single source shortest path problem as shown in Figure 1.2. Chapter 6 shows that it will become undesirable to describe the problem as shortest path problem. Therefore in the next sections, a new algorithm to determine an optimal trading strategy, called Algorithm 1, is described.

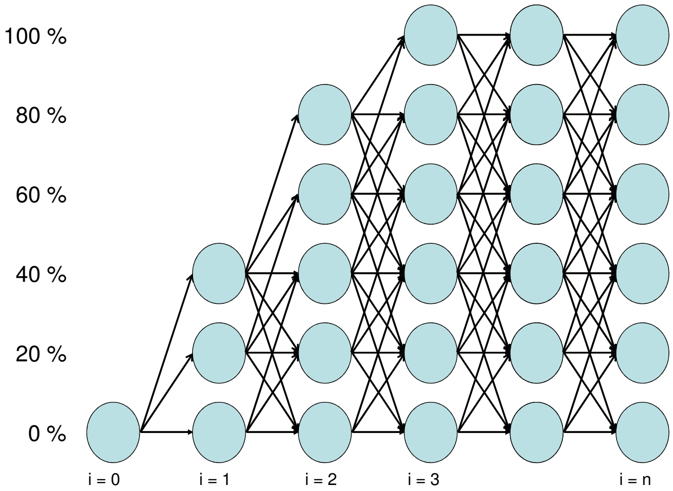


Figure 1.2: Optimal trading strategy determined by dynamic programming

In the Introduction short descriptions of a storage system, the power market and the grid were given. This section gives a model of the storage system. The capacity of a storage system, C, can be expressed in kWh, which is a unit for energy. The (dis)charge power of a storage system can be expressed in kW, which is a unit for power. The APX uses intervals of 15 minutes for the energy prices in €/MWh, thus the energy prices that are calculated by the APX are constant for that time. Therefore models that are used in this context can logically be discretized in time using intervals of 15 minutes. In this model of the storage system it is only required to know the amount of energy that can be (dis)charged during an interval. Therefore we will define the maximum amount of energy that can be charged during one interval as the *charge capacity*, ChC, in kWh, and the maximum amount of energy that can be discharged during one interval as the *discharge capacity*, $DChC$, in kWh.

Definition 1.1. ChC in kWh:=(the charge power in kW)·(the time of an interval in h.)

Definition 1.2. $DChC$ in kWh:=(the discharge power in kW)·(the time of an interval in h.)

For the modeled storage system the charge capacity is one, the discharge capacity is minus one and the capacity of the storage system is one. Therefore in each interval the state of charge of the storage system for every interval i, $SOC(i)$, can either be fully charged or discharged. In this model it is assumed that there are no energy losses by using the storage system, and the power grid gives no constraints. The storage system is empty at the start and there is no residual value for energy. The energy prices for interval i, $p(i)$, are given by the APX, these energy prices are independent of the $SOC(i)$ since such a storage system in the low voltage grid is too small to influence the national market.

Table 1.1 gives a summary of the properties of the modeled storage system. Also the decision variables are included in this table. Such a table will be used in every chapter to give an overview of what is changed in the model as compared to the previous chapter. (Note that $DChQ(i) \leq 0$.)

Name	Abbreviation	Value	Unit
Energy price for interval i	$p(i)$	input	€/kWh
Charge Capacity	ChC	1	kWh
Discharge Capacity	$DChC$	-1	kWh
Capacity of the Storage System	C	1	kWh
Quantity of energy charged in interval i	$ChQ(i)$	$\{0,1\}$	kWh
Quantity of energy discharged in interval i	$DChQ(i)$	{-1,0}	kWh
State of Charge for interval i	$SOC(i)$	$\{0,1\}$	kWh

Table 1.1: Parameters and decision variables for Model A

1.2 Mathematical model

To develop an optimal trading strategy we can describe the optimization problem with the following mathematical model:

$$\max \sum_{i=1}^{n} (-(DChQ(i) + ChQ(i)) \cdot p(i)) \tag{1.1}$$

$$s.t.\ SOC(i) = \sum_{j=1}^{i} (ChQ(j) + DChQ(j)) \qquad ; 1 \leq i \leq n \tag{1.2}$$

$$SOC(i) \in \{0,1\} \qquad ; 1 \leq i \leq n \tag{1.3}$$

$$ChQ(i) \in \{0,1\} \qquad ; 1 \leq i \leq n \tag{1.4}$$

$$DChQ(i) \in \{-1,0\} \qquad ; 1 \leq i \leq n \tag{1.5}$$

In expression (1.1) the profit that can be made by using the storage system for trading is maximized. Although this seems to be a very short mathematical model of the problem, there are many different summations made in expression (1.2) since $1 \leq i \leq n$. In expression (1.3) it is given that the $SOC(i)$ is either equal to one, which is the capacity of the storage system or equal to zero for interval i, for i between one and n. In expression (1.4) it is given that the quantity of energy charged in interval i is always zero or one, while in expression (1.5) the quantity of energy discharged is always zero or minus one for i between one and n.

With this mathematical model the optimization problem is described as a binary integer programming problem. Though these problems are classified as nondeterministic polynomial time hard, for more details we refer the reader to [4]. In Section 1.3 it is proven that this specific problem can be solved and in Section 1.5 it is shown that this problem can be solved in linear time.

1.3 Approach

With the capacity of the storage system equal to the charge capacity, it is possible to fully charge the storage system during interval i if the $SOC(i)$ is empty. With the capacity of the storage system equal to minus the discharge capacity, it is also possible to fully discharge the storage system during interval j if the $SOC(j)$ is full.

To obtain a profit by trading energy, the selling energy price must be larger than the purchase energy price. To obtain the maximum profit, the difference between the selling and purchase price must be as large as possible. Since it is not possible to sell energy that is not stored, first energy must be bought to charge the storage system. In Model A the storage system can charge and discharge infinitely many times. Hence the storage system can charge energy in interval i, if the $SOC(i)$ is not full, and if $p(i)$ is less than $p(i+1)$. The storage system can discharge energy in interval j if the $SOC(j)$ is not empty, and if $p(j)$ is higher than $p(j+1)$. Once the storage system is charged, it cannot be charged again before it is discharged since the state of charge is full. The storage system can only be discharged after it is charged. This can be summarized as in Table 1.2.

SOC \ Action	Buying/Charging	Selling/Discharging	Do nothing
Full	not possible	possible	possible
Empty	possible	not possible	possible

Table 1.2: Possible actions for trading in Model A

By Table 1.2, it is known when it is possible to buy, to sell or to do nothing. To be able to determine an optimal trading strategy, it is required to know when it is most profitable to buy and sell. To show when it is best to trade, as an example a price list is given and an optimal trading strategy is determined for this example. In Figure 1.3, in the first graph the energy price for all intervals, are shown (externally given by the APX). In this graph it can be seen that it is best to charge in interval 3 and to discharge in interval 6 to obtain the maximum profit. In the second graph the charging and the discharging is shown. In the third graph the resulting SOC can be seen. The cumulative cash flow is given in the fourth graph, in this last graph the total profit is shown. When we annalyse why it is optimal to charge in interval 3 and to discharge in interval 6, we can define when we want to charge and when we want to discharge. We want to charge in a *last local minimum* as in Definition 1.3 and we want to discharge in a *subsequent local maximum* as in Definition 1.4.

Definition 1.3. A *last local minimum* is the first interval i, that can be used for charging for which the energy price is less than the energy price of the next interval in line.
$SOC(i) = 0 \;\&\; p(i) < p(i+1)$

Definition 1.4. A *subsequent local maximum* is the first interval j, that can be used for discharging for which the energy price is larger than the energy price of the next interval in line.
$SOC(j) = 1 \;\&\; p(j) > p(j+1)$

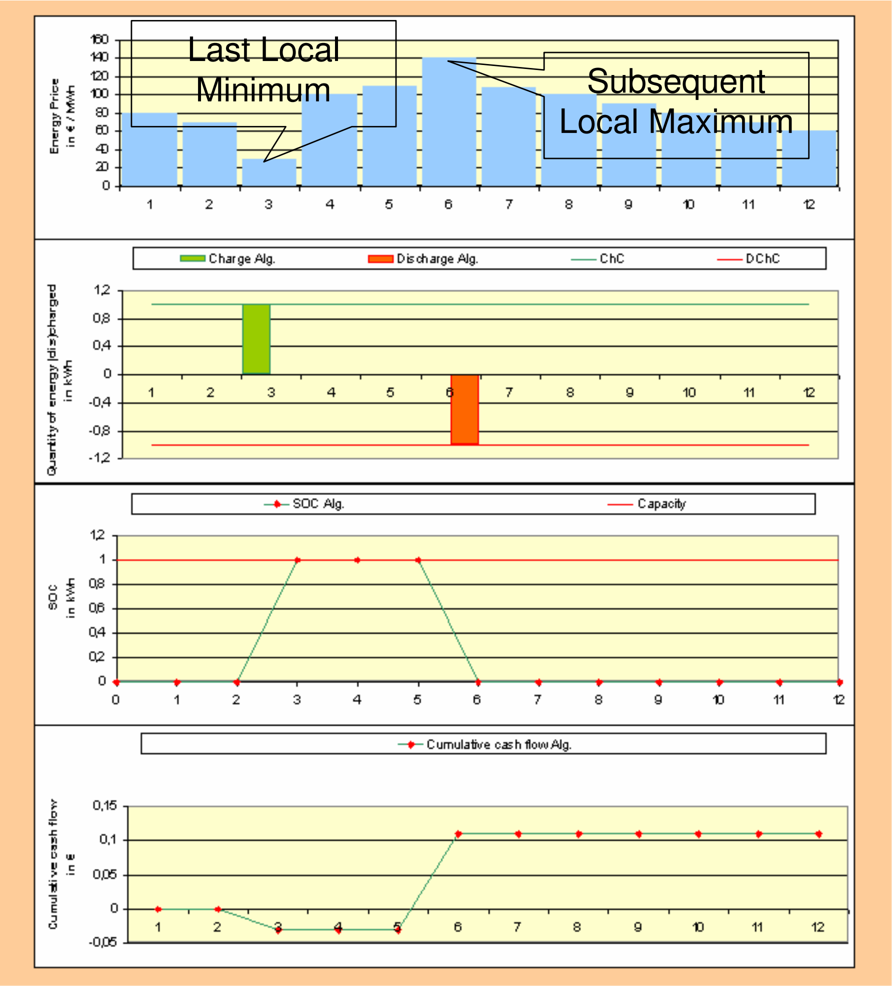


Figure 1.3: The last local minimum and it's subsequent local maximum

1.4 Optimal trading strategy

To obtain a maximum profit, the storage system will be charged in every last local minimum and discharged in every subsequent local maximum. To make sure the last interval can be used as subsequent local maximum we must be able to compare the energy price of the last interval, interval n, with the energy price of interval $n + 1$. Since there is no residual value for energy, the energy price for interval $n + 1$ can be set to be 0.

The trading strategy to obtain a maximum profit can be formulated as an algorithm, see Algorithm 1. In the next chapters this algorithm will be extended. Since this algorithm is used as basis for the other algorithms, Algorithm 1 is shown next. The other algorithms are listed in the appendixes. The text in green are comments.

Algorithm 1

```
Declarations
$i = 1..n$                 the set of intervals
$p(i)$                     the energy price of interval $i$
$ChQ(i)$                   is the quantity charged in interval $i$
$DChQ(i)$                  is the quantity discharged in interval $i$
$SOC(i) := 0$              state of charge of interval $i$ is empty
$SOC(i) := 1$              state of charge of interval $i$ is full

Data & Initialization
$p(1)..p(n)$
$p(n+1) := 0$
$SOC(1) := 0$

Program
 $i := 1$
 : This WHILE LOOP goes chronologically through the intervals starting with interval 1 :
 while $i \leq n$ do
   : IF the current interval is a last local minimum, set ChQ(i) as charging and set $SOC(i)$ as FULL :
   if $p(i) < p(i+1)$ and $SOC(i) < 1$ then
     $SOC(i) := 1$
     $ChQ(i) := 1$
   end if
   : IF the current interval is a subsequent local maximum, set DChQ(i) as discharging and set $SOC(i)$ as EMPTY :
   if $p(i) > p(i+1)$ and $SOC > 0$ then
     $SOC(i) := 0$
     $DChQ(i) := -1$
   end if
   $i = i + 1$
   $SOC(i) = SOC(i-1)$
 end while
```

The outcome of Algorithm 1, an optimal trading strategy, is a list of intervals to charge $ChQ(i)$, and a list of intervals to discharge $DChQ(i)$. The output of Algorithm 1 is processed in a program to show the $ChQ(i)$, the $DChQ(i)$, the resulting $SOC(i)$, and the cumulative cash flow.

Proposition 1.5. *Algorithm 1 produces an optimal trading strategy to obtain a maximum profit.*

In order to prove that this trading strategy is optimal for Model A, we need to prove that it is not possible to obtain a higher profit with another trading strategy.

Proof. If interval i is determined as the last local minimum, we know from Definition 1.3 that for all intervals k that can be used for charging up to interval $i+1$, $p(k)$ is not less than $p(i)$. Therefore it is not possible to charge energy for a price that is less than the energy price of interval i, in the intervals up to interval $i+1$, interval $i+1$ included. Since for all intervals k up to interval i, interval i included, $p(k)$ are non-increasing, it is not possible to make a profit with first buying and subsequently selling energy during these intervals that are available for charging energy, before interval i. Since interval i is the last interval of a list of intervals with non-increasing energy prices it is given that $p(i+1)$ is larger than $p(i)$. Therefore, we can sell, in interval $i+1$, the energy charged in interval i with a profit.

With the definition of the subsequent local maximum given in Definition 1.4 we know that if interval j is determined as the subsequent local maximum, then $p(j)$ is larger than $p(l)$, with interval l between interval i and $j+1$, interval $j+1$ included. The energy prices of the intervals between interval i and j, interval j included, are non-decreasing. Therefore it is not possible to make a higher profit by first selling and subsequently buying energy between the intervals i and j. Since it is possible to buy energy in interval $j+1$ and $p(j)$ is higher than $p(j+1)$, it is best to sell the energy in interval j, since it is profitable to sell energy in interval j, buying energy in interval $j+1$. Once the last local minimum and subsequent local maximum are determined the process is repeated which gives an optimal trading strategy for a storage system as modeled in Model A. □

In Figure 1.4 an example is given of how the energy price can fluctuate. In this figure it is shown how the process can be repeated and that charging in the last local minimum and discharging in the subsequent local maximum gives an optimal trading strategy.

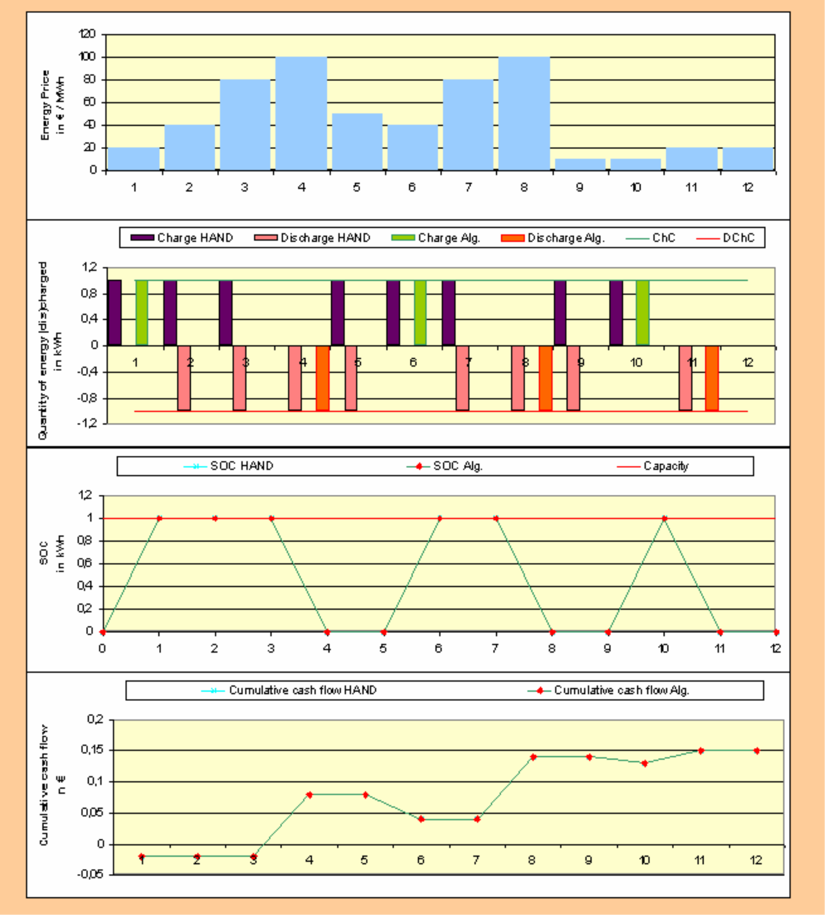


Figure 1.4: Example of optimum trading

1.5 Structure and complexity

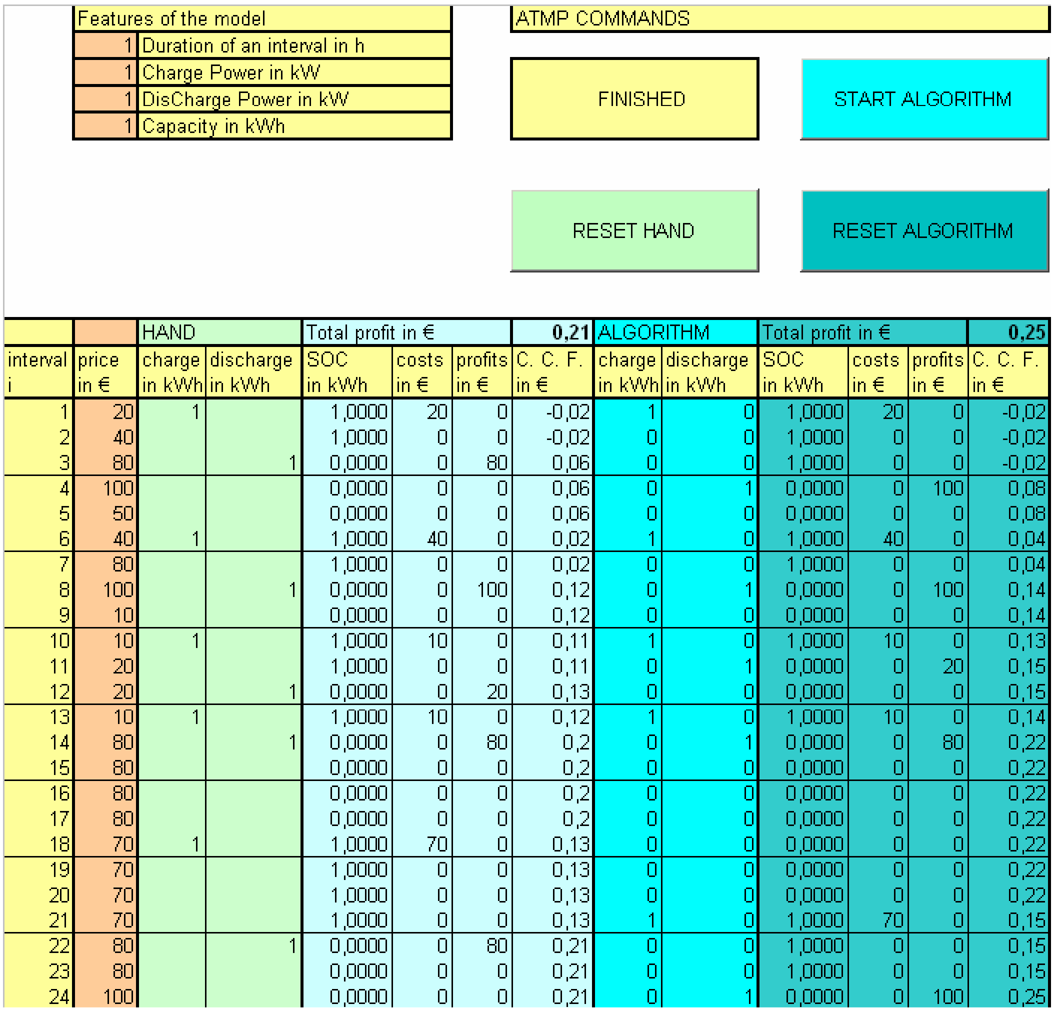
The structure of Algorithm 1 is straightforward. For each interval we determine if it is a last local minimum, a subsequent local maximum, or if it is just an interval that will not be used to charge or discharge energy. Every interval is checked once to see if it can be used to trade. The complexity to check every interval once is $O(n)$. To determine if interval i will be used for charging or for discharging energy, the energy price of interval i and the energy price of interval $i + 1$ are compared, and the $SOC(i)$ is determined. This has complexity $O(1)$. Thus, the complexity of Algorithm 1 is $O(n) \cdot O(1) = O(n)$.

1.6 Reflection and result

To give a clear overview of the trading strategy, the program ATMP, **A**lgorithm for **T**rading with **M**aximum **P**rofit, is written in Visual Basic Application Excel during the research. In ATMP the user is able to compare Algorithm 1 with any other trading strategy one can come up with. In Figure 1.5 the front page of ATMP is shown. The user can enter the energy price for every interval. Also the charge power, the discharge power and the capacity of the storage system can be entered in this front page. The user can make an attempt to determine an optimal trading strategy to obtain a maximum profit. With a simple click on the button "START PROGRAM" an optimal trading strategy that gives the maximum profit will be given.

In Figure 1.6 the graphs drawn by ATMP are shown. In the first graph, the energy price is given. The second graph shows which intervals Algorithm 1 determined to charge and discharge the storage system and during which intervals the user wants to charge and discharge the storage system. The third graph shows the effect on the SOC by charging and discharging the storage system, for Algorithm 1 as well as for the user. In the last graph the cumulative cash flow is shown for Algorithm 1 and the user. This last graph shows which strategy has a better result.

An optimal trading strategy for a storage system as given in Model A is to charge in every last local minimum as defined in Definition 1.3, and to discharge in every subsequent local maximum as defined in Definition 1.4. Since it is wanted to charge the maximum amount of energy in the last local minimum and to discharge the maximum amount of energy in the subsequent local maximum, the output of the Algorithm would not be different when this problem was modeled as a linear programming problem. With Algorithm 1 the optimal trading strategy to obtain a maximum profit is determined in linear time. Algorithm 1 will be used as a reference in the following extended models.



Features of the model	
1	Duration of an interval in h
1	Charge Power in kW
1	DisCharge Power in kW
1	Capacity in kWh

		HAND		Total profit in €			0,21	ALGORITHM		Total profit in €			0,25
interval i	price in €	charge in kWh	discharge in kWh	SOC in kWh	costs in €	profits in €	C. C. F. in €	charge in kWh	discharge in kWh	SOC in kWh	costs in €	profits in €	C. C. F. in €
1	20	1		1,0000	20	0	-0,02	1	0	1,0000	20	0	-0,02
2	40			1,0000	0	0	-0,02	0	0	1,0000	0	0	-0,02
3	80		1	0,0000	0	80	0,06	0	0	1,0000	0	0	-0,02
4	100			0,0000	0	0	0,06	0	1	0,0000	0	100	0,08
5	50			0,0000	0	0	0,06	0	0	0,0000	0	0	0,08
6	40	1		1,0000	40	0	0,02	1	0	1,0000	40	0	0,04
7	80			1,0000	0	0	0,02	0	0	1,0000	0	0	0,04
8	100		1	0,0000	0	100	0,12	0	1	0,0000	0	100	0,14
9	10			0,0000	0	0	0,12	0	0	0,0000	0	0	0,14
10	10	1		1,0000	10	0	0,11	1	0	1,0000	10	0	0,13
11	20			1,0000	0	0	0,11	0	1	0,0000	0	20	0,15
12	20		1	0,0000	0	20	0,13	0	0	0,0000	0	0	0,15
13	10	1		1,0000	10	0	0,12	1	0	1,0000	10	0	0,14
14	80		1	0,0000	0	80	0,2	0	1	0,0000	0	80	0,22
15	80			0,0000	0	0	0,2	0	0	0,0000	0	0	0,22
16	80			0,0000	0	0	0,2	0	0	0,0000	0	0	0,22
17	80			0,0000	0	0	0,2	0	0	0,0000	0	0	0,22
18	70	1		1,0000	70	0	0,13	0	0	0,0000	0	0	0,22
19	70			1,0000	0	0	0,13	0	0	0,0000	0	0	0,22
20	70			1,0000	0	0	0,13	0	0	0,0000	0	0	0,22
21	70			1,0000	0	0	0,13	1	0	1,0000	70	0	0,15
22	80		1	0,0000	0	80	0,21	0	0	1,0000	0	0	0,15
23	80			0,0000	0	0	0,21	0	0	1,0000	0	0	0,15
24	100			0,0000	0	0	0,21	0	1	0,0000	0	100	0,25

Figure 1.5: The front page of ATMP(1)

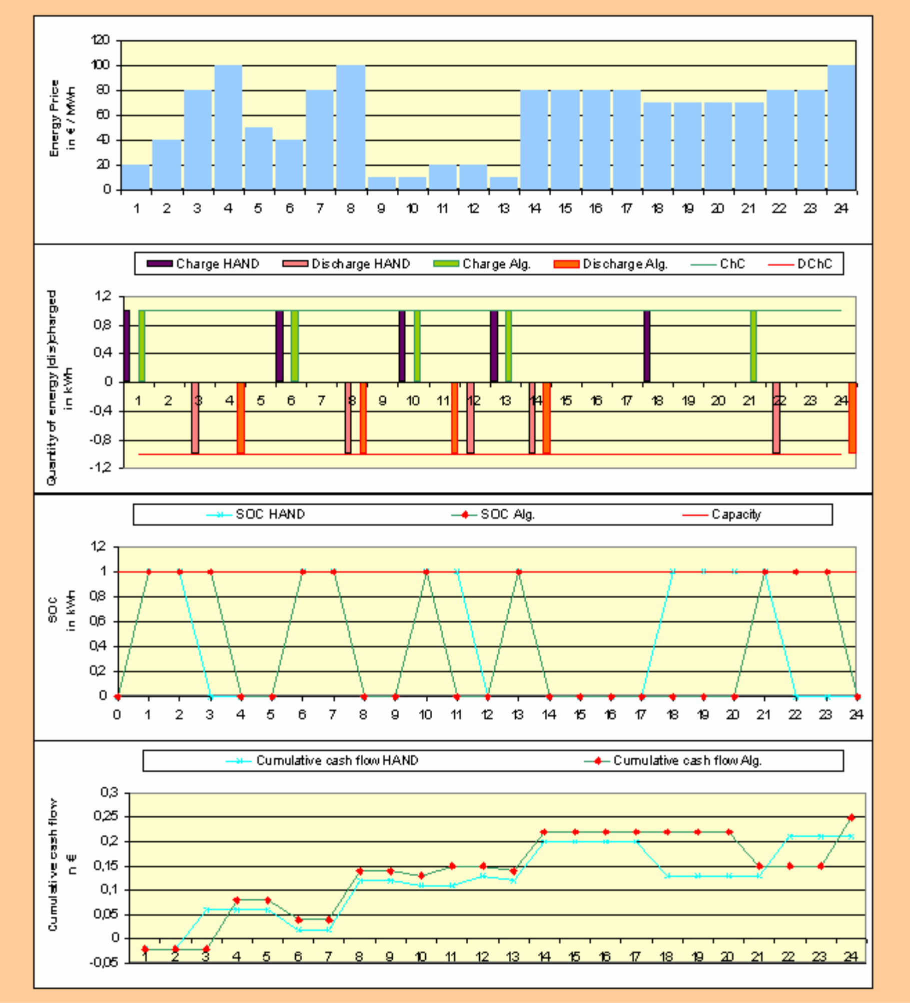


Figure 1.6: The graphs produced by ATMP(1)

Chapter 2

Algorithm 2: Including scaling

In this chapter, Model A as described in Chapter 1 will be extended. In reality, storage systems can have different values for the capacity of the storage system, the charge capacity and the discharge capacity. Therefore, as an extension to Model A, in Model B it is possible to enter different values for these constraints. With this extension, the model of the storage system becomes more realistic. Even though this is only a small extension, it makes a big difference in how an optimal trading strategy that obtains a maximum profit for the modeled storage system can be determined.

2.1 Model of the storage system

Model B is an extension to Model A, in which more states of charge are used, for instance 0%, 20%, 40%, 60%, 80% and 100% of the full capacity of the storage system. Also, the power to charge can be limited as well as the power to discharge. Therefore the charge and discharge capacity are limited. As an example, we assume that during fifteen minutes the storage system can charge up to 40% of the full capacity of the storage system because of the limited power to charge. And because of the limited power to discharge, during fifteen minutes the storage system can discharge up to 60% of the full capacity of the storage system. Still, this problem can be solved as a single source shortest path problem, but the graph we obtain becomes larger. An example of such a graph is given in Figure 2.1.

In Chapter 1, the capacity, the charge and the discharge capacity of the storage system modeled in Model A all have the same absolute value. The storage system modeled in Model B can have three different values for these physical constraints. Model B will only be different from Model A when the capacity of the storage system is larger than the charge and/or the absolute value of the discharge capacity. Otherwise the charge capacity and the discharge capacity will be bound by the capacity of the storage system and therefore the capacity of the storage system would always be fully used like in Model A.

With the capacity of the storage system, C, larger than the charge capacity of the storage system, ChC, it is possible that an optimal solution will not make fully use of the capacity of the storage system for every interval. Therefore the state of charge of the storage system for every interval i, $SOC(i)$, can take any value between zero and C. Also the quantity of energy that is charged during interval i, $ChQ(i)$, can take any value between zero and ChC. Likewise for the discharge capacity of the storage system, $DChC$. When the capacity of the storage system is larger than the absolute value of the discharge capacity of the storage system,

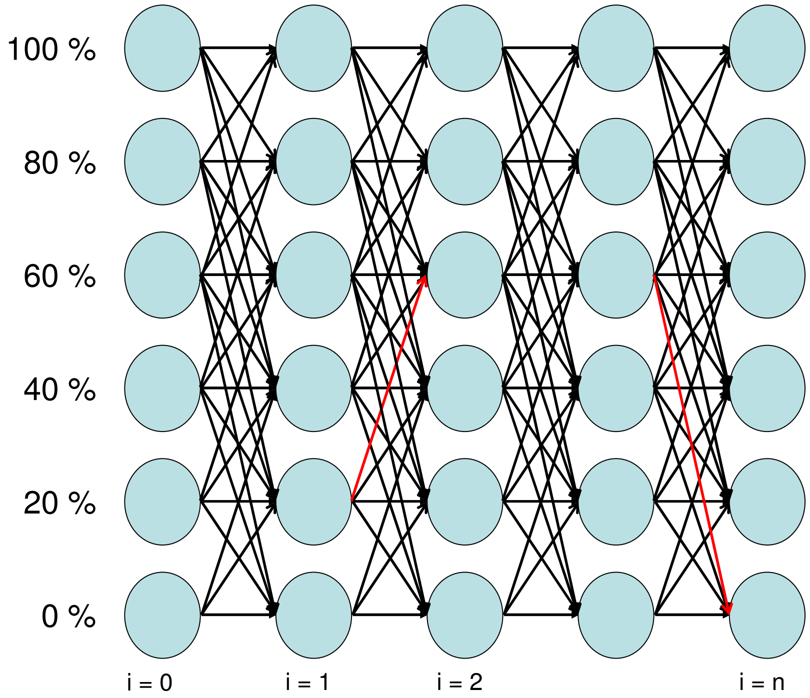


Figure 2.1: The problem as a graph

the quantity of energy that is discharged during an interval i, $DChQ(i)$, can take any value between $DChC$ and zero.

With this extension the model becomes more realistic, and even though this change might not seem to make that big a difference, the optimization problem to determine an optimal trading strategy for the modeled storage system to obtain a maximum profit, is solved in polynomial time like the problem in Chapter 1, but notice that the running time of Algorithm 1 in Chapter 1 is even linear. With the absolute values of the three physical constraints not equal to each other, it is possible to charge more than once before discharging and to discharge several times after charging. The problems that occur because of the extension of Model A will be discussed by examples. These examples will show which improvements are needed to Algorithm 1, to determine an optimal trading strategy that obtains a maximum profit for Model B. There are several extensions to Algorithm 1 required to develop a new algorithm. Algorithm 1 is extended step by step, so that all extensions can be described.

Like in Chapter 1 the parameters and the decision variables of Model B are summarized in Table 2.1. In this table the difference between Model A and B is clear. While in Model A the ChC, the $DChC$ and the C are given as ± 1, in Model B these are arbitrary input values. Also for Model A, the $SOC(i)$, the $ChQ(i)$, the $DChQ(i)$ can be seen as booleans, while in Model B they have values between zero and C, zero and ChC, and $DChC$ and zero, respectively. This makes Model A a discrete model for the SOC, the ChQ and the $DChQ$ while Model B is a continuous model.

Name	Abbreviation	Value	Unit
Energy price for interval i	$p(i)$	input	€/kWh
Charge Capacity	ChC	input	kWh
Discharge Capacity	$DChC$	input	kWh
Capacity of the Storage System	C	input	kWh
Quantity of energy charged in interval i	$ChQ(i)$	$0 \leq ChQ(i) \leq ChC$	kWh
Quantity of energy discharged in interval i	$DChQ(i)$	$DChC \leq DChQ(i) \leq 0$	kWh
State of Charge for interval i	$SOC(i)$	$0 \leq SOC(i) \leq C$	kWh

Table 2.1: Parameters and decision variables for Model B

2.2 Mathematical model

The optimization problem to determine an optimal trading strategy for the modeled storage system can be described with the following mathematical model:

$$\max \sum_{i=1}^{n} (-(DChQ(i) + ChQ(i)) \cdot p(i)) \tag{2.1}$$

$$s.t.\ SOC(i) = \sum_{j=1}^{i} (ChQ(j) + DChQ(j)) \qquad ; 1 \leq i \leq n \tag{2.2}$$

$$0 \leq SOC(i) \leq C \qquad ; 1 \leq i \leq n \tag{2.3}$$

$$0 \leq ChQ(i) \leq ChC \qquad ; 1 \leq i \leq n \tag{2.4}$$

$$DChC \leq DChQ(i) \leq 0 \qquad ; 1 \leq i \leq n \tag{2.5}$$

The difference between Model A and B is shown in the differences between expression (1.3) till (1.5) and (2.3) till (2.5). While the mathematical model for Model A is a binary integer programming problem, the mathematical model for Model B is a linear programming problem. In 1979, it was proven by L.G. Khachiyan, that it is possible to solve such a problem in polynomial time. For more details we refer the reader to [3]. To get a good insight in this specific problem, an algorithm to solve this problem will be developed in this chapter. As said above, $C > ChC$ and/or $C > |DChC|$ for Model B to be an extension of Model A. (Note that for Model B, just as for Model A: $DChQ(i) \leq 0$.)

2.3 Approach

To give clear examples that show the shortcomings of Algorithm 1 for this new model, the input ChC, $DChC$ and C are set. For all examples in this chapter we use $ChC = 1$, $DChC = -1$ and $C = 3$.

2.3.1 Heuristic 2.1: State of charge

With the capacity of the storage system being three times larger than the charge and the absolute value of the discharge capacity of the storage system, it is clear that the state of charge as used in Chapter 1 can no longer be used as a boolean. Therefore we extend Algorithm 1 such that the state of charge can take any value between zero and the capacity of the storage system. There is also an extension of the charge and discharge capacity since it is possible that the capacity of the storage system does not allow the storage system to charge or discharge the total charge or discharge capacity. These extensions to Algorithm 1 are not enough to give an optimal trading strategy. We need to extend the algorithm further. Therefore the extended version of Algorithm 1 is called Heuristic 2.1, this heuristic is given in Appendix A. The trading strategy as determined by Heuristic 2.1 could be summarized as follows:
If the energy price for interval i, $p(i)$, is less than $p(i+1)$ and the $SOC(i)$ is not full, the storage system must be charged as much as possible in interval i. If the energy price for interval j, $p(j)$, is larger than $p(j+1)$ and the $SOC(j)$ is not empty, the storage system must be discharged as much as possible in interval j. This heuristic goes chronologically once through all intervals.

2.3.2 Counterexample Heuristic 2.1

The following counterexample can be used to show that it cannot be guaranteed that Heuristic 2.1 determines an optimal trading strategy. The energy prices are as shown in the first graph of Figure 2.2. The second graph shows when to charge and when to discharge, according to the trading strategy determined by Heuristic 2.1 and according to a better trading strategy. The effect on the SOC is shown in the third graph and the last graph shows the cumulative profit for both trading strategies. For this example we used $ChC = 1$, $DChC = 1$ and $C = 3$. The strategy as given by Heuristic 2.1 gives an outcome with a loss.
$-20 - 40 - 80 + 100 = -40$.
In the graph, it is taken into account that the energy prices are in MWh, therefore the losses are $-40/1000 = -0.04$.

As given in Chapter 1 there are no residual values, thus it is not desirable to charge more energy than can be discharged. Since the charge capacity is equal to the absolute value of the discharge capacity, it is not optimal to charge in interval 1, 2 and 3 while only in interval 4 energy is discharged. Instead a better outcome would be to charge in interval 1 and 2 and to discharge in interval 3 and 4. Since all profitable combinations of intervals to charge and to discharge are used, this is an optimal solution. This trading strategy would give a profit of
$100 + 80 - 40 - 20 = 120$.
In the grapg this gives 0.12.

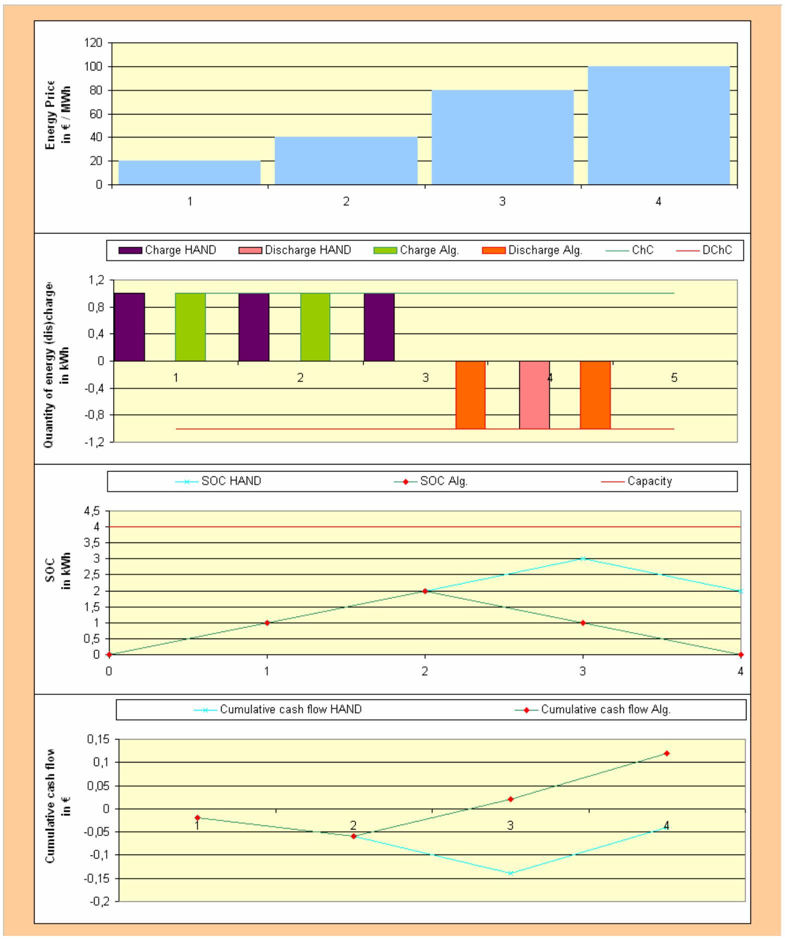


Figure 2.2: Counterexample 1

2.3.3 Heuristic 2.2: The subsequent local maximum

The previous example shows that it is important to charge energy in an interval only when there is a subsequent interval in which we can discharge this energy with a maximum profit. Since it was optimal in Chapter 1 to discharge energy in the so called subsequent local maximum we need to find a subsequent local maximum for every last local minimum. Once an interval is fully used to discharge energy it can no longer be used to discharge more energy. Therefore it is desirable to not define it as a subsequent local maximum again. In Algorithm 1 an interval would not be visited twice, thus an interval could not be defined as subsequent local maximum if the discharge capacity of this interval was already fully used. But now it could occur that an interval would be defined as subsequent local maximum while the discharge capacity of this interval is fully used, and therefore we redefine the subsequent local maximum.

Definition 2.1. A *subsequent local maximum* is the last interval, for which the discharge capacity is not fully used, of a non-decreasing period for the energy price after a local minimum.

Using this new definition of the subsequent local maximum in a new heuristic gives a new trading strategy. This extension of Heuristic 2.1 is called Heuristic 2.2, this heuristic is given in Appendix B. The trading strategy as determined by Heuristic 2.2 could be summarized as follows:
If the energy price for interval i, $p(i)$ is less than $p(i+1)$, and the $SOC(i)$ is not full, there must be a subsequent local maximum determined, interval j. The maximum amount of energy that can be charged in interval i and discharged in interval j, with the $SOC(k)$ taken into account for $i \leq k \leq j$ will be charged and discharged in the intervals i and j.

2.3.4 Counterexample Heuristic 2.2

If Heuristic 2.2 would be used, an optimal solution is still not guaranteed. The next counterexample can be used to show this. The energy prices are as shown in the first graph of Figure 2.3. The second graph shows when to charge and when to discharge, according to the trading strategy determined by Heuristic 2.2 and according to a better trading strategy. The effect on the SOC is shown in the third graph and the last graph shows the cumulative profit for both trading strategies.

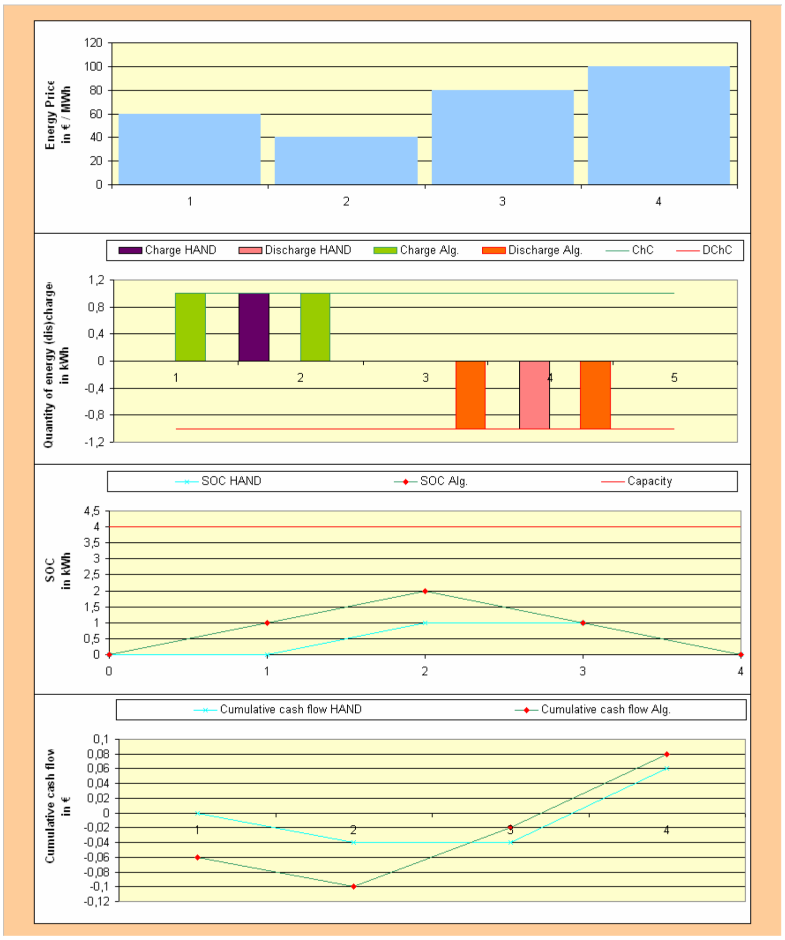


Figure 2.3: Counterexample 2

The strategy as given by Heuristic 2.2 gives an outcome with a profit of
$-40 + 100 = 60$.
Also for this example, in the graph this is 0.06.

With Heuristic 2.2, interval 1 was not identified as an interval to charge, while the energy that can be charged in interval 1 can be discharged in interval 3. This would give a profit of
$-60 - 40 + 80 + 100 = 80$.
Which is in the graph 0.08.

This profit is better than the outcome of Heuristic 2.2, therefore an improvement to Heuristic 2.2 is required.

The extension described next will give an algorithm that determines an optimal trading strategy.

2.4 Optimal trading strategy

With only a little extension of Heuristic 2.2 an optimal trading strategy is developed. Interval 1 was not found as a possibility to charge energy since Heuristic 2.2 only goes through the price list once chronologically to find the last local minimum. To find every last local minimum that can be used to charge energy, to be discharged in the subsequent local maximum, the heuristic cannot continue chronologically. After using the last found last local minimum it would be better to start to search again for the new last local minimum in the first interval for which the state of charge is not equal to the capacity of the storage system, after the last interval for which he state of charge is equal to the capacity of the storage system. Now we need to redefine the last local minimum.

Definition 2.2. The *last local minimum* is the last interval, that can be used to charge energy, of a non-increasing period for the energy price, that has a subsequent local maximum.

Combining the new definition of the last local minimum with Heuristic 2.2 gives Algorithm 2, that guarantees an optimal strategy. Algorithm 2 is given in Appendix C. After a short explanation of how this algorithm works it will be proven that it guarantees an optimal trading strategy.

Algorithm 2 can be described with the flow stream as given in Figure 2.4. To determine the maximum quantity energy to trade, the minimum of the energy that can be charged in the last local minimum i, and the energy that can be discharged in the subsequent local maximum j, is determined. Than it is determined for every interval k, between the last local minimum and the subsequent local maximum if the quantity energy added to the $SOC(k)$ is not larger than the capacity of the storage system. If the quantity energy is to larg, the new quantity is determined using Heuristic 2.3.

Heuristic 2.3

```
$CHQ := min(ChC + DChQ(i) - ChQ(i), C - SOC(i))$
$DCHQ := min(DChC - DChQ(j), C - SOC(j))$
$Q := min(CHQ, DCHQ)$
$k := i + 1$
:In this WHILE LOOP for all intervals from the LLM to the SLM the maximum amount of energy to be stored is determined. :
while $k < j$ do
  $Q := min(Q, C - SOC(k))$
  $k := k + 1$
end while
```

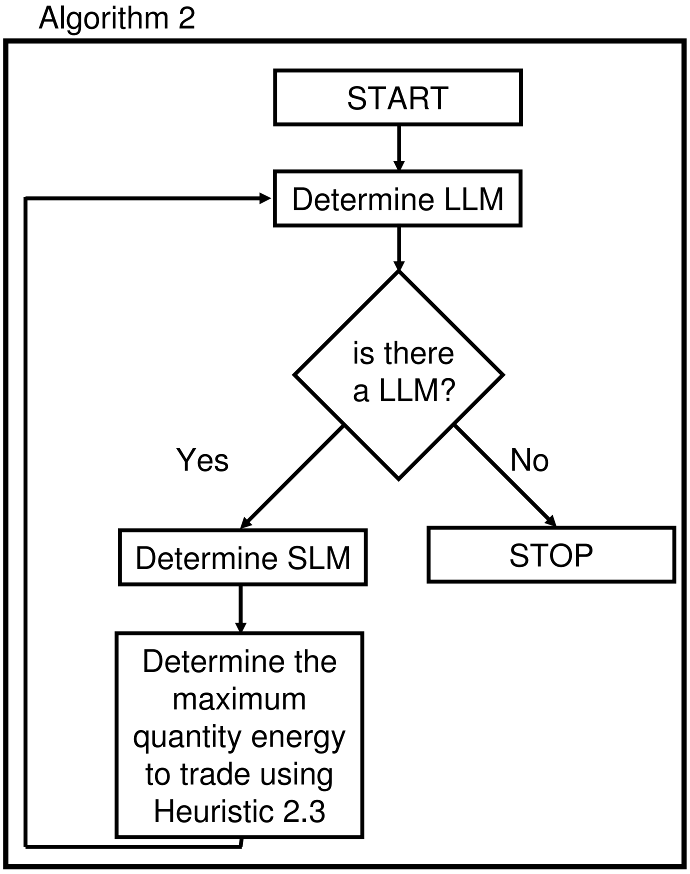

Algorithm 2
START
Determine LLM
is there a LLM?
Yes
No
Determine SLM
STOP
Determine the maximum quantity energy to trade using Heuristic 2.3

Figure 2.4: Flow stream of Algorithm 2

Algorithm 2 gives like Algorithm 1, a list of intervals in which to (dis)charge energy, together with the amount of energy to be (dis)charged in these intervals. The last local minimum as defined in Definition 2.2 and the subsequent local maximum as defined in Definition 2.1 are determined in Algorithm 2. It is never possible that we find a subsequent local maximum before an interval that is already used for charging energy. Therefore we can only use intervals for charging after the last interval for which the $SOC = C$. This is used in Algorithm 2. The subsequent local maximum is always after the last local minimum, and also not before the last interval that can be used for discharging energy, before the last found subsequent local maximum. This is also used in Algorithm 2. To give a feasible solution, the maximum amount of energy to be charged, discharged and stored is determined in Algorithm 2.

Proposition 2.3. *Algorithm 2 produces an optimal trading strategy that gives a maximum profit for the modeled storage system by Model B.*

In order to prove that the new trading strategy is optimal for Model B, we need to prove that it is not possible to obtain a larger profit with another trading strategy. It is important to understand that the new trading strategy determined by Algorithm 2 for Model B, is in fact not that different from the trading strategy determined by Algorithm 1 for Model A. In the strategy determined by Algorithm 1, it was known that it was not possible to charge more than once before discharging. Therefore searching for a new local minimum was only needed for intervals after the interval that was used for discharging, since the state of charge would be equal to the capacity of the storage system until then. In Model B it is possible to charge more often before discharging. Therefore the search for a last local minimum is started in the first interval after the last interval with the $SOC = C$ for every iteration. It is proven in Chapter 1 that it is optimal to charge in a last local minimum with a subsequent local maximum for the energy price and to discharge in this subsequent local maximum. Since we redefined the last local minimum and the subsequent local maximum we still need to prove that using these new defined last local minimum and subsequent local maximum still provides an optimal trading strategy.

Proof. We must prove that, when an interval i is determined as a last local minimum, to be used to charge energy, there is no interval available before the matching subsequent local maximum (the interval which will be used to discharge energy) with a lower energy price than the last local minimum. This means that we need to prove that when interval i cannot charge energy, since during previous iterations energy is charged in other intervals, that it can never be better to use interval i instead of previous found intervals. When using interval i as a last local minimum for charging energy interferes with charging energy during interval k that is determined as a last local minimum during a previous iteration, the energy price of interval k has to be less than or equal to the price of the new found interval i. When charging in a new found interval i is not possible because of an interval k that was determined as a last local minimum in a previous iteration, the so called subsequent local maximum, interval j, that was found as subsequent local maximum of the last local minimum k, has to be later in time then the new found last local minimum i. To give a clear overview, the first graph of Figure 2.5 shows the intervals i, j and k.

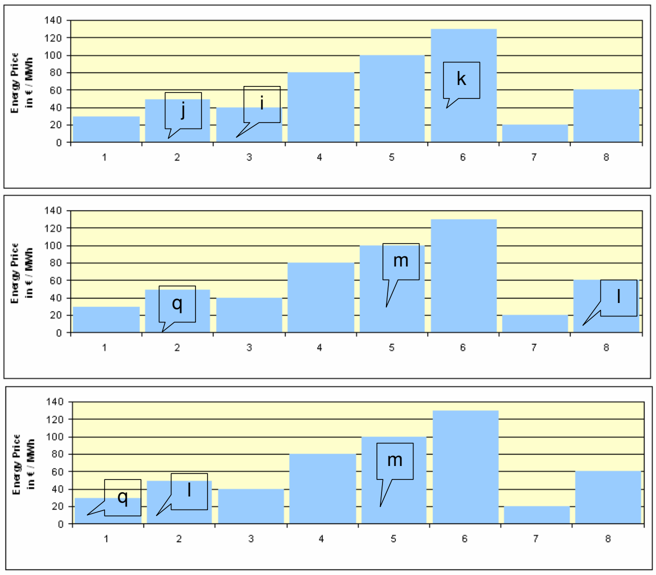


Figure 2.5: The algorithm improves the outcome every iteration

The first found last local minimum k is the interval with the absolute minimum price of all intervals available for charging before the subsequent local maximum j. Therefore the new found interval i has to have a price larger than the energy price of interval k used for charging.

We need to prove that when an interval l is used to discharge energy, that there is not an interval m, not fully used to discharge energy, after the found last local minimum q, with an energy price that is larger than the energy price of the interval l that is used. If interval m would be before interval l, than interval m would be the subsequent local maximum of the found last local minimum q and thus fully used for discharging energy if possible. This is shown in the second graph of Figure 2.5.

Let interval l be used to discharge energy, with an energy price that is less than the energy price for a subsequent interval m, as shown in the third graph of Figure 2.5. If there is not another interval before interval m where energy can be charged, to be discharged in interval m, and it is possible that interval l will be determined as the last local minimum and interval m the subsequent local maximum, the energy planned to be discharged in interval l will not be discharged in interval l but this energy will be discharged in interval m. This means that the action of discharging in interval l is undone and this amount of energy is discharged in interval m. Therefore it is not possible that an interval l in which energy is discharged has an energy price that is less than the energy price of a subsequent interval m that is not used for the maximum discharge capacity when it is possible to determine interval l as the last local minimum. This is shown in Figure 2.5

The search for a last local minimum is started in the first interval after the last interval with the $SOC = C$ for every iteration, therefore every usable moment to charge and to discharge energy will be found.
This makes the trading strategy optimal. □

2.5 Structure and complexity

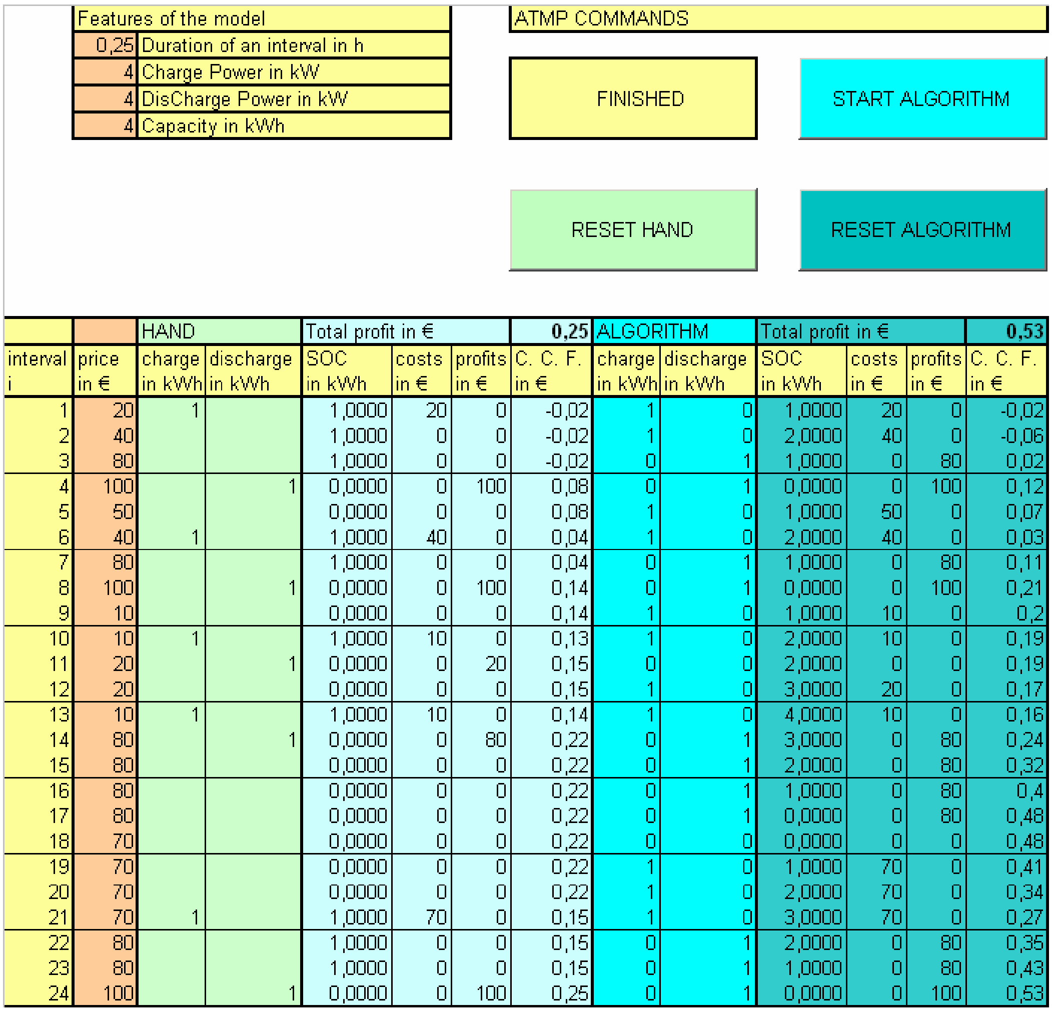
The structure of Algorithm 2 is more complex. There are three inner loops, those will be described first. During the first inner loop there is an interval determined as the last local minimum. Since there are n intervals, this can be done in $O(n)$. If there is a last local minimum determined, the second inner loop determines an interval as the subsequent local maximum. This is also done in $O(n)$. Once there is a last local minimum and a subsequent local maximum determined, the amount of energy to be charged, discharged and stored is determined in the third inner loop. This is also done in $O(n)$. These are all actions that are done one after another. This gives $O(n)$ for the three inner loops together.

The inner loops of Algorithm 2 will be repeated until there cannot be a last local minimum determined in the first inner loop. It is possible that there are intervals first determined as subsequent local maximum and during a subsequent iteration determined as last local minimum. Once an interval is determined as last local minimum, it cannot be determined as subsequent local maximum during subsequent iterations. Therefore the three inner loops can be repeated at most $2 \cdot 1/2 \cdot n$ times. While it is possible that a last local minimum is repeatedly used, for every time a last local minimum is repeatedly used there is a subsequent local maximum used that cannot be used again as a subsequent local maximum. Thus this process has $O(n)$, and since this includes the inner loops with complexity $O(n)$, the complexity of Algorithm 2 is $O(n) \cdot O(n) = O(n^2)$. This is clearly worse than the complexity of Algorithm 1, therefore the small extension of Model A has great impact for the complexity.

2.6 Reflection and result

In this section there will be a reflection on the results given. The differences between Algorithm 1 and Algorithm 2 is shown in Figure 2.7. In this figure, the solution of Algorithm 1 is called HAND and Algorithm 1 is called Algorithm. It is shown that Algorithm 1 uses only the last local minimum and the subsequent local maximum as defined in Definitions 1.3 and 1.4, while Algorithm 2 uses the new defined last local minimum and subsequent local maximum as defined in this chapter. Therefore the profit of Algorithm 1 is at least as large as the profit of Algorithm 2.

To obtain an optimal profit for Model B, the storage system is charged in every local minimum as defined in Definition 2.2 with a subsequent local maximum as defined in Definition 2.1. The amount of energy that must be charged in this last local minimum depends on the SOC of all intervals between the last local minimum and the subsequent local maximum including the last local minimum, and it depends on the charge and discharge capacity that is still available. This trading strategy gives an optimal outcome as proven above. With Model B, the modeled storage system is more realistic than with Model A. Algorithm 2 is more complex than Algorithm 1.



Features of the model	
0,25	Duration of an interval in h
4	Charge Power in kW
4	DisCharge Power in kW
4	Capacity in kWh

ATMP COMMANDS

		HAND		Total profit in €			0,25	ALGORITHM		Total profit in €			0,53
interval i	price in €	charge in kWh	discharge in kWh	SOC in kWh	costs in €	profits in €	C. C. F. in €	charge in kWh	discharge in kWh	SOC in kWh	costs in €	profits in €	C. C. F. in €
1	20	1		1,0000	20	0	-0,02	1	0	1,0000	20	0	-0,02
2	40			1,0000	0	0	-0,02	1	0	2,0000	40	0	-0,06
3	80			1,0000	0	0	-0,02	0	1	1,0000	0	80	0,02
4	100		1	0,0000	0	100	0,08	0	1	0,0000	0	100	0,12
5	50			0,0000	0	0	0,08	1	0	1,0000	50	0	0,07
6	40	1		1,0000	40	0	0,04	1	0	2,0000	40	0	0,03
7	80			1,0000	0	0	0,04	0	1	1,0000	0	80	0,11
8	100		1	0,0000	0	100	0,14	0	1	0,0000	0	100	0,21
9	10			0,0000	0	0	0,14	1	0	1,0000	10	0	0,2
10	10	1		1,0000	10	0	0,13	1	0	2,0000	10	0	0,19
11	20		1	0,0000	0	20	0,15	0	0	2,0000	0	0	0,19
12	20			0,0000	0	0	0,15	1	0	3,0000	20	0	0,17
13	10	1		1,0000	10	0	0,14	1	0	4,0000	10	0	0,16
14	80		1	0,0000	0	80	0,22	0	1	3,0000	0	80	0,24
15	80			0,0000	0	0	0,22	0	1	2,0000	0	80	0,32
16	80			0,0000	0	0	0,22	0	1	1,0000	0	80	0,4
17	80			0,0000	0	0	0,22	0	1	0,0000	0	80	0,48
18	70			0,0000	0	0	0,22	0	0	0,0000	0	0	0,48
19	70			0,0000	0	0	0,22	1	0	1,0000	70	0	0,41
20	70			0,0000	0	0	0,22	1	0	2,0000	70	0	0,34
21	70	1		1,0000	70	0	0,15	1	0	3,0000	70	0	0,27
22	80			1,0000	0	0	0,15	0	1	2,0000	0	80	0,35
23	80			1,0000	0	0	0,15	0	1	1,0000	0	80	0,43
24	100		1	0,0000	0	100	0,25	0	1	0,0000	0	100	0,53

Figure 2.6: The front page of ATMP(2)

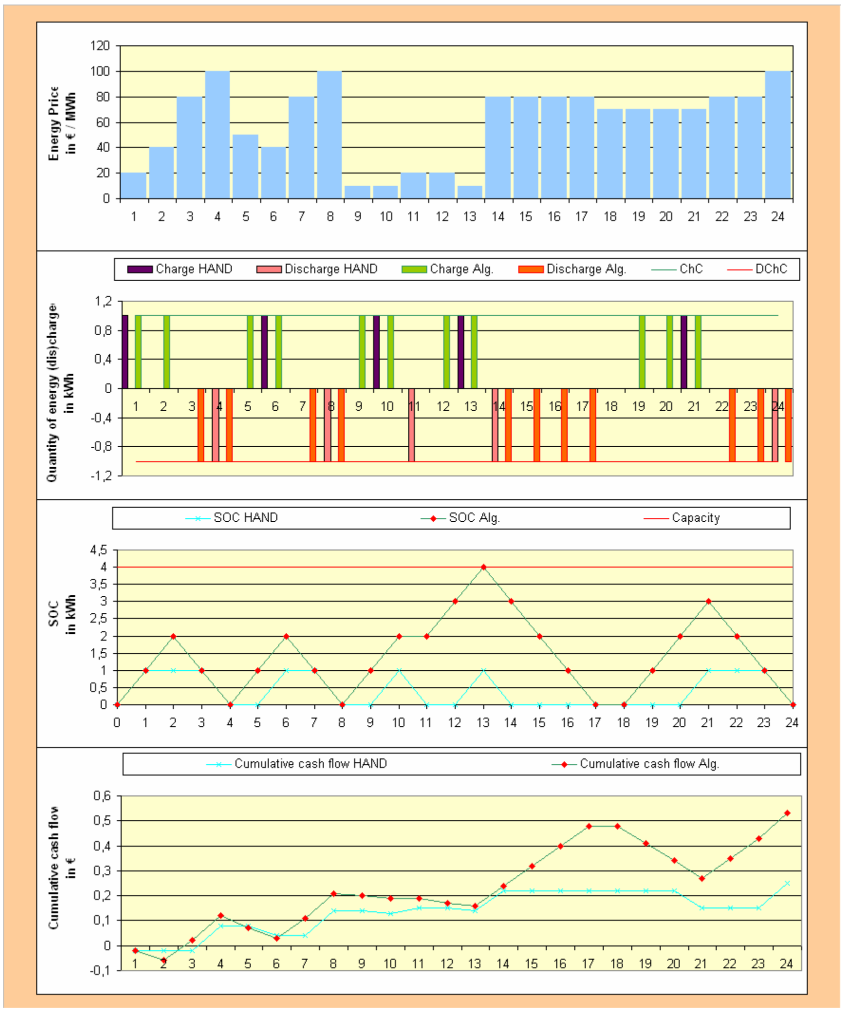


Figure 2.7: The graphs produced by ATMP(2)

Chapter 3

Algorithm 3: Including losses

In Model C, as extension to Model B, losses will be taken into account. By the law of conservation of energy, using a storage system cannot be without losses. There is energy required to (dis)charge the storage system, electricity energy will be transformed into thermal energy. There is also electricity energy transformed for instance into irretrievable chemical energy caused by storage. Therefore with taking losses into account, the modeled storage system becomes more realistic. This chapter describes the new calculations, required to determine an optimal trading strategy taking losses into account. Algorithm 3 is described as the extension of Algorithm 2. Algorithm 3 determines an optimal trading strategy for a modeled storage system by Model C. Though there are some new calculations required, the complexity of Algorithm 3 is $O(n^2)$ like the complexity of Algorithm 2.

3.1 Model of the storage system

As an extension to Model B, in Model C energy losses from using the storage system can be taken into account. There is energy required for charging and for discharging the storage system. This is energy that cannot be used for trading. Also, in time the energy in the storage system decreases, this is energy that cannot be sold. The energy that cannot be sold are losses from using the storage system. With these losses taken into account, there will be many more states of charge. In order to construct an associated graph in which we can model the problem as a single source shortest path problem, the states of charge must be discrete. With the losses taken into account, the state of charge can take any value within the capacity of the storage system. Therefore it is no longer possible to construct a discrete model of the problem, when losses are taken into account. There are three types of losses as described above:

- losses caused by storage (LBS),
- losses caused by charging (LBC),
- losses caused by discharging ($LBDC$).

To take these losses into account in the mathematical model, it is important to clarify how these losses are modeled. Also it is important to know how they affect the state of charge of the storage system, SOC, and/or the profit. In this section the losses caused by storage are described first. The effect of these losses on the $SOC(i)$ and the profit will be described. After this, the losses caused by charging are described. For these losses, the effect on the $SOC(i)$ and the profit is described, in this description the other losses are not taken into account. At last the losses caused by discharging are described. These losses are modeled such that they have no effect on the $SOC(i)$, but they affect the profit. Finally the new calculations to determine the $SOC(i)$ and the profit are expressed in expression (3.11) and (3.12).

3.1.1 Losses caused by storage

Model D is, like the other models, discrete in time. Therefore the result of the losses by storage during an interval is required. To be able to determine the $SOC(i)$, the $SOC(i-1)$ must be reduced with the losses by storage. The energy that is lost in time during interval i, is modeled as a percentage of the $SOC(i-1)$. We determine the $SOC(i)$ by calculating the residual energy after the losses by storage, $RLBS$ as calculated in expression (3.1).

$$RLBS := \frac{100 - LBS}{100} \tag{3.1}$$

For an interval i, in which there is no energy charged or discharged, the $SOC(i)$ is calculated, as in expression (3.2), using the residual after the losses by storage.

$$SOC(i) := SOC(i-1) \cdot RLBS \tag{3.2}$$

The effect on the SOC by the losses caused by storage are shown in Figure 3.1, to be able to show the effect extreem values are chosen.

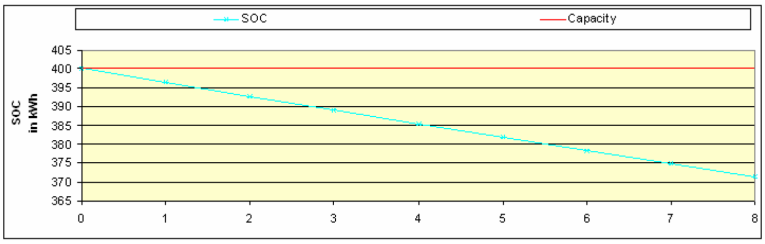


Figure 3.1: The losses by storage

The losses caused by storage must be taken into account for every interval. If there is a combination of a last local minimum i and a subsequent local maximum j found for trading, the contribution to the profit of this combination (i, j), can be expressed as in expression (3.3). In this expression the discharge quantity of the subsequent local maximum is expressed as the charge quantity of the last local minimum, decreased with the losses caused by storage.

$$Profit(i, j) := p(j) \cdot ChQ(i) \cdot RLBS^{j-i} - p(i) \cdot ChQ(i) \tag{3.3}$$

In the mathematical model, to determine the $SOC(i)$, the losses by storage are adapted as in expression (3.4).

$$SOC(i) := \sum_{i=1}^{m}((ChQ(i) + DChQ(i)) \cdot RLBS^{m-i}); 1 \leq i \leq n \tag{3.4}$$

In the mathematical model, to determine the profit, the losses by storage are adapted. This is not as clear as for the $SOC(i)$, but since the $SOC(i)$ is decreased taking the losses caused by storage into account, the $DChQ(i)$ is bound, and thus the losses are taken into account for the profit as well. To show how the losses by storage affect the SOC and the profit, an extreme example is shown in Figure 3.2.

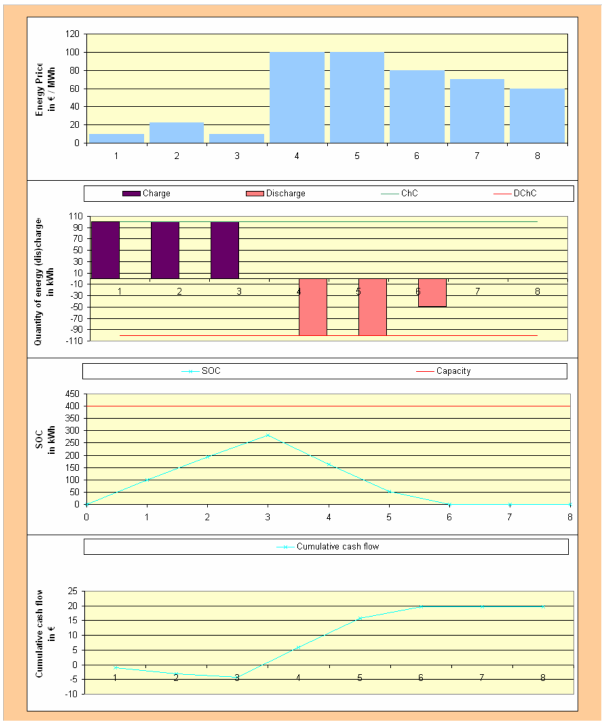


Figure 3.2: The losses by storage affect the SOC and the profit

3.1.2 Losses caused by charging

When energy is charged, the resulting losses must be taken into account as well. Again it is needed to determine the energy that really is in the storage system, after charging the storage system. The energy that is lost by charging, is modeled as a percentage of the energy that is charged. We are interested in the $SOC(i)$ after charging in interval i, thus we calculate the residual after the losses suffered by charging, $RLBC$, as in expression (3.5).

$$RLBC := \frac{100 - LBC}{100} \tag{3.5}$$

The $SOC(i)$ for interval i, in which there is energy charged, is calculated as in expression (3.6).

$$SOC(i) := SOC(i-1) \cdot RLBS + ChQ(i) \cdot RLBC \tag{3.6}$$

If there is a combination of a last local minimum i and a subsequent local maximum j found for trading, the contribution to the profit, of this combination (i, j), can be expressed as in expression (3.7). In this expression the discharge quantity of the subsequent local maximum is expressed as the charge quantity of the last local minimum, decreased with the losses caused by charging.

$$Profit(i, j) := p(j) \cdot ChQ(i) \cdot RLBC - p(i) \cdot ChQ(i) \tag{3.7}$$

In the mathematical model, to determine the $SOC(i)$, the losses by charging are adapted as in expression (3.8).

$$SOC(i) := \sum_{i=1}^{m} ((ChQ(i) + DChQ(i)) \cdot RLBC); 1 \leq i \leq n \tag{3.8}$$

In the mathematical model, to determine the profit, the losses by charging are adapted similar as the losses caused by storage. Since the $SOC(i)$ is decreased taking the losses caused by charging into account, the $DChQ(i)$ is bound, and thus the losses are taken into account for the profit as well. To show how the losses by charging affect the SOC and the profit, an extreme example is shown in Figure 3.3.

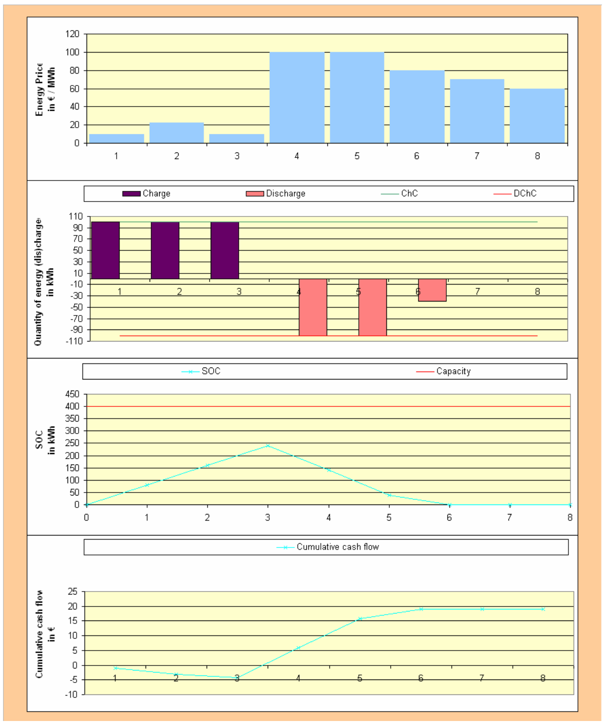


Figure 3.3: The losses by charging affect the SOC and the profit

3.1.3 Losses caused by discharging

When discharging energy, the resulting losses must be taken into account as well. The losses caused by discharging are expressed as a percentage of the energy that is discharged. Therefore, a percentage of the energy that is discharged cannot be sold since electricity energy is transformed into thermal energy. Similar as the other losses, also for the losses suffered by discharging, we need to determine the residual of the discharge energy after losses by discharging, $RLBDC$ as calculated in expression (3.9).

$$RLBDC := \frac{100 - LBDC}{100} \tag{3.9}$$

While the other losses affected the $SOC(i)$ the losses caused by discharging are taken over energy that is discharged and thus this has no effect on the $SOC(i)$. The electricity energy that is transformed in heat, cannot be sold. Therefore these losses affect the profit. If there is a combination of a last local minimum i and a subsequent local maximum j found for trading, the contribution to the profit, of this combination (i, j), can be expressed as in expression (3.10). In this expression the discharge quantity of the subsequent local maximum is expressed as the charge quantity of the last local minimum, decreased with the losses caused by discharging.

$$Profit(i,j) := p(j) \cdot ChQ(i) \cdot RLBDC - p(i) \cdot ChQ(i) \tag{3.10}$$

With the losses suffered by discharging having no effect on the $SOC(i)$, in the mathematical model, the $SOC(i)$ is calculated as in expression (3.11).

$$SOC(i) := \sum_{i=1}^{m}((ChQ(i) \cdot RLBC + DChQ(i)) \cdot RLBS^{m-i}); 1 \leq i \leq n \tag{3.11}$$

In the mathematical model, to determine the profit, the losses by discharging are adapted as in expression (3.12).

$$profit := \sum_{i=1}^{n}(-(DChQ(i) \cdot RLBDC + ChQ(i)) \cdot p(i)) \tag{3.12}$$

To show how the losses by charging affect the profit, an extreme example is shown in Figure 3.3.

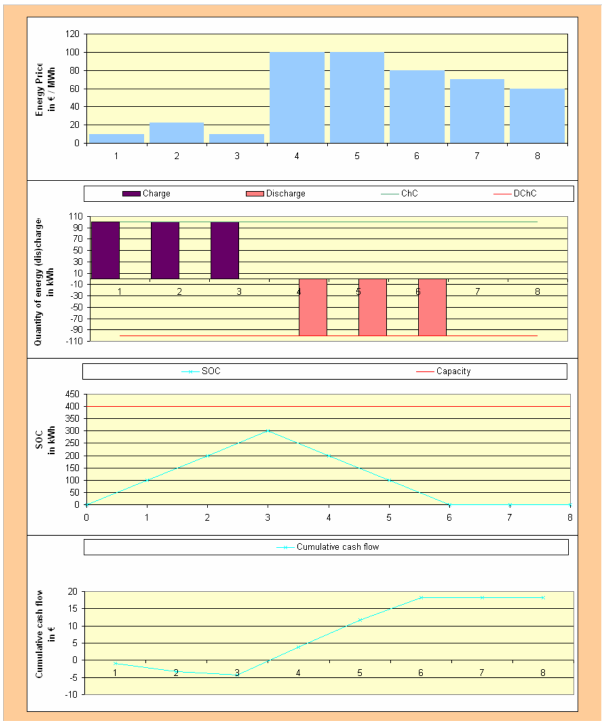


Figure 3.4: The losses by discharging affect the profit

The effect of the losses is that there is less efficiency by charging and by discharging. Therefore the profit decreases. How the losses are included in Algorithm 3 is described in the next sections, but first the variables used in this chapter will be summarized in Table 3.1.

Name	Abbreviation	Value	Unit
Energy price for interval i	$p(i)$	input	€/kWh
Charge Capacity	ChC	input	kWh
Discharge Capacity	$DChC$	input	kWh
Capacity of the Storage System	C	input	kWh
Quantity of energy charged in interval i	$ChQ(i)$	$0 \leq ChQ(i) \leq ChC$	kWh
Quantity of energy discharged in interval i	$DChQ(i)$	$DChC \leq DChQ(i) \leq 0$	kWh
State of Charge for interval i	$SOC(i)$	$0 \leq SOC(i) \leq C$	kWh
Charging price for interval i	$cp(i)$	calculated	€/kWh
Discharging price for interval i	$dcp(i)$	calculated	€/kWh
Losses by Charging	LBC	$0 - 100$ (% of ChQ)	%
Losses by Discharging	$LBDC$	$0 - 100$ (% of $DChQ$)	%
Losses by Storage	LBS	$0 - 100$ (% of SOC)	%
Residual after Losses by Charging	$RLBC$	$\frac{100-LBC}{100}$	
Residual after Losses by Discharging	$RLBDC$	$\frac{100-LBDC}{100}$	
Residual after Losses by Storage	$RLBS$	$\frac{100-LBS}{100}$	

Table 3.1: Parameters and decision variables for Model C

3.2 Mathematical model

The optimization problem to determine an optimal trading strategy for the modeled storage system can be described with the following mathematical model:

$$\max \sum_{i=1}^{n} (-(DChQ(i) \cdot RLBDC + ChQ(i)) \cdot p(i)) \tag{3.13}$$

$$s.t.\ SOC(i) := \sum_{i=1}^{m} ((ChQ(i) \cdot RLBC + DChQ(i)) \cdot RLBS^{m-i}) \qquad ; 1 \leq i \leq n \tag{3.14}$$

$$0 \leq SOC(i) \leq C \qquad ; 1 \leq i \leq n \tag{3.15}$$

$$0 \leq ChQ(i) \leq ChC \qquad ; 1 \leq i \leq n \tag{3.16}$$

$$DChC \leq DChQ(i) \leq 0 \qquad ; 1 \leq i \leq n \tag{3.17}$$

This mathematical model shows that the losses by discharging only affect the profit in expression (3.13), as described above. The state of charge that is calculated in expression (3.14), is influenced by the losses by charging, this is because not all energy charged can be stored. Some of this energy is used for charging. Also the losses by storage affect the *SOC* as shown in this model. The *SOC* is not influenced by the losses by discharging, since this is energy that would not be in the storage system already.

3.3 Approach

With the losses taken into account, some calculations become more complex, but the basics of the trading strategy as described in Chapter 2 are not changed; still we search for the last local minimum, LLM, and the subsequent local maximum, SLM, and we determine the maximum amount of energy to trade. In this section we will describe what calculations are required to find the last local minimum, the subsequent local maximum and the maximum amount of energy that can be charged, discharged and stored. The losses that are taken into account, can be used to determine a virtual energy price to charge energy in interval i, $cp(i)$, and a virtual energy price to discharge energy in interval j, $dcp(j)$. These prices are used to determine the trading strategy.

3.3.1 Charging price and discharging price

To determine the last local minimum, the losses must be taken into account. The amount of energy that is in the storage system after charging, the SOC, will decrease in time due to the losses by storage. To reflect the costs of these losses, a virtual price can be calculated for every interval. Energy stored decreases in time. If for interval i and $i+1$ the energy prices are equal, because of the losses by storage it would be better to charge energy in interval $i+1$. The virtual price of interval i, $vp(i)$ should thus be larger than the virtual price of interval $i+1$. The original energy price of every interval can be recalculated by expression (3.18).

$$vp(i) := p(i) \cdot RLBS^i \tag{3.18}$$

The virtual price can be used for charging as well as for discharging since they both are affected by the losses suffered by storage. The losses by charging will change the energy price for charging energy, and the losses by discharging will change the energy price for discharging energy. The losses for charging and for discharging are taken into account separately, to be able to make a difference between charging energy in an interval, and not discharging the stored energy in an interval to be able to discharge this energy in another interval. The losses caused by charging increase the virtual price. Therefore the charging price is divided by the residual after losses by charging. The losses caused by discharging decrease the virtual price. The discharging price is multiplied by the residual after losses by discharging as in Algorithm 3.1.

Algorithm 3.1

Data & Initialization

$p(1)..p(n)$	original energy price
$cp(1)..cp(n)$	virtual charging price
$dcp(1)..dcp(n)$	virtual discharging price
$RLBC$, $RLBDC$ and $RLBS$	

Program

```
$i := 1$
while $i \leq n+1$ do
    $cp(i) := \frac{p(i) \cdot RLBS^i}{RLBC}$
    $dcp(i) := p(i) \cdot RLBS^i \cdot RLBDC$
    $i := i + 1$
end while
```

Once the energy prices are recalculated by Algorithm 3.1, the new prices can be used to determine the last local minimum and the subsequent local maximum. With the new prices to charge and to discharge energy, an optimal trading strategy can be determined. The extension of Algorithm 2 is Algorithm 3. Section 3.4 describes Algorithm 3, but first in this section an overview of the algorithms used in Algorithm 3 is given. At the end, it will be proven that Algorithm 3 guarantees an optimal trading strategy for a storage system modeled in Model C.

3.3.2 The last local minimum and the subsequent local maximum

To find the subsequent local maximum as defined in Definition 2.1, for a given last local minimum as defined in Definition 2.2, some calculations are required. In Model C, it is possible that there is a subsequent local maximum j found, with a $dcp(j)$, that is not profitable in combination with the last local minimum i, since $cp(i) \geq dcp(j)$. Therefore a new algorithm is developed to determine a last local minimum and it's subsequent local maximum. First a last local minimum i is determined. For this last local minimum i, the subsequent local maximum is searched. If there is a subsequent local maximum found that is not profitable, the search continues. If an interval k is found, with $cp(k) \leq cp(i)$, interval k is the new last local minimum and for this last local minimum k, the search for the subsequent local maximum continues. If there is a profitable subsequent local maximum found, the maximum amount of energy to trade is to be determined as in Section 3.3.3. In Algorithm 3.2 as described in Appendix D, it is described how the last local minimum and it's subsequent local maximum can be determined.

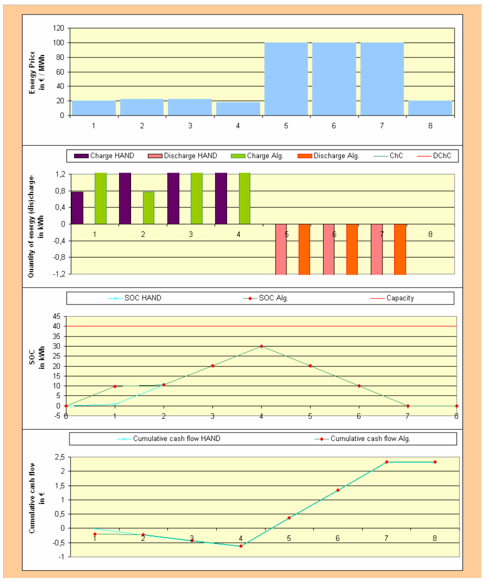


Figure 3.5: The search for the last local minimum

While in Algorithm 2 the search for a last local minimum started in the first interval available for charging, not fully used for discharging, before the last found last local minimum, the search for a last local minimum will now start in the first interval that can be used for charging in Algorithm 3. For Algorithm 2 it was not required to start the search in that interval but as shown in Figure 3.5, for Algorithm 3 it is required, to be able to determine the optimal trading strategy. In the figure, it is not real clear that the profit is better with this new strategy. This is because the losses are only 2.5% thus this is not very clear in a graph, but it is shown that the new algorithm uses another trading strategy.

3.3.3 Amount of energy

When the last local minimum and the subsequent local maximum are known, the energy to trade must be calculated. Taking the losses in Model C into account, calculating the maximum amount of energy to charge, discharge and store is more complex. The basics of these calculations are already given in Algorithm 2, new calculations are needed, which are described in Algorithm 3.3 as given in Appendix E. In Algorithm 3.3 the calculation as described in expression (3.11) is used to determine the SOC. The SOC must be kept larger or equal to zero and smaller or equal to the capacity of the storage system. The charge and the discharge capacity must be respected as well, for the solution to be feasible.

3.3.4 New charging price

When energy is charged, the losses caused by charging are taken into account by the charging price. If the discharge quantity of interval j, $DChQ(j)$, is smaller than zero, this energy can be used to be discharged in a subsequent interval $j + k$, as shown in Figure 3.6. The first graph is the start situation. The second graph shows the result after 1 iteration. This is what is described here. The third graph is described later. To transport the energy that is available because of the discharge quantity of interval j, the losses caused by charging should not be taken into account again. Therefore it is needed to recalculate the charging price of interval j, if by an iteration the $DChQ(j)$ becomes negative, as in expression (3.19)

$$cp(j) := cp(j) \cdot RLBC \tag{3.19}$$

Similar when we are used to just transport energy from interval m to subsequent intervals, the $DChQ(m)$ can become zero again, as shown in the third graph of Figure 3.6. Than we cannot just transport energy from interval m to subsequent intervals but energy must be charged again and thus the losses caused by charging must be taken into account again for the charging price, as in expression (3.20). The final result of this strategy is shown in Figure 3.7.

$$cp(j) := \frac{cp(j)}{RLBC} \tag{3.20}$$

It is not required to do these calculations for the discharging price, this is because it is not possible that an interval set to charge energy is determined as a subsequent local maximum. Algorithm 3.3 as given in Appendix E uses the expressions (3.19) and (3.20), to determine the new charging prices if required after the charging and discharging quantities are determined.

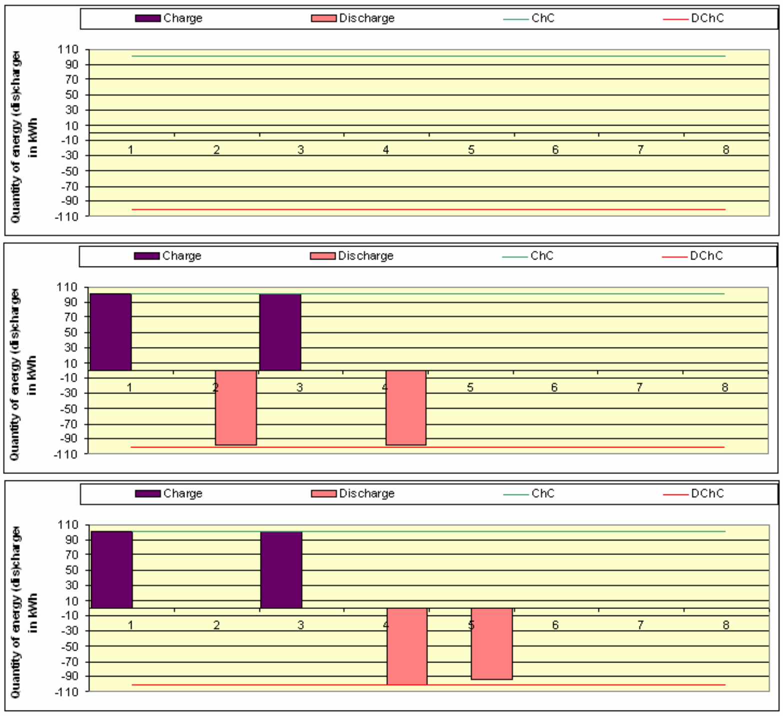


Figure 3.6: The changing charging price

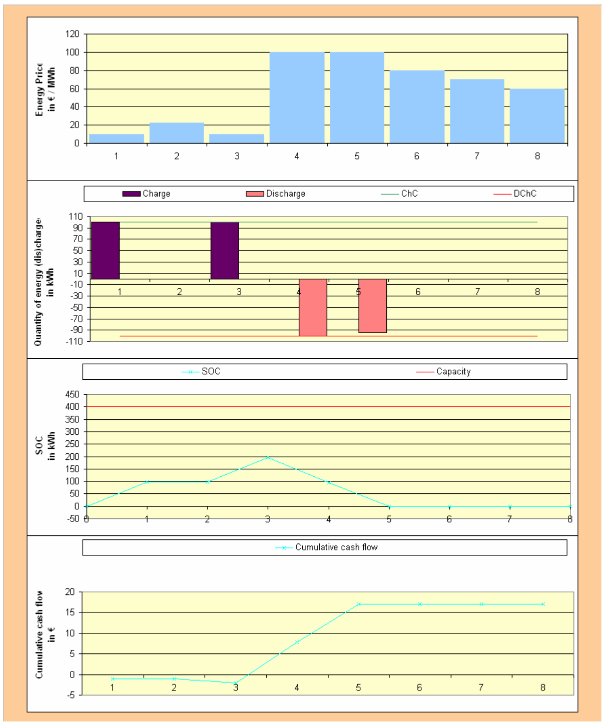


Figure 3.7: Recalculating the charging price when interval i is set to discharge

3.4 Optimal trading strategy

Algorithm 3 is an extension of Algorithm 2. Like in Algorithm 2, in Algorithm 3 the maximum amount of energy to trade is used for every last local minimum combined with a profitable subsequent local maximum. To give a clear view of Algorithm 3 the flow stream of Algorithm 3 is given in Figure 3.8. The used algorithms 3.1 till 3.3 are described earlier, as well as expressions (3.19) and (3.20), that are used in Algorithm 3.3.

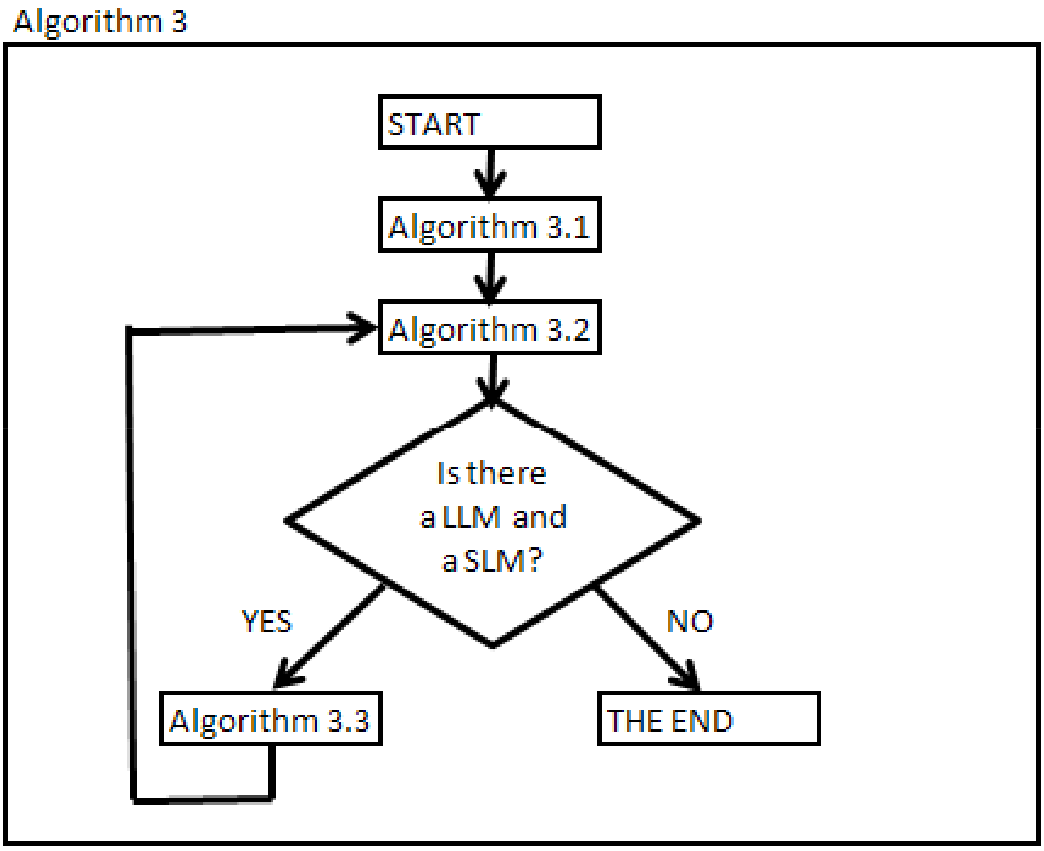


Figure 3.8: Flow stream of Algorithm 3

Algorithm 3 gives a list of intervals to charge energy, with the amount of energy to be charged during these intervals. There is also a list given of intervals to discharge energy, with the amount of energy to be discharged. The profit of this strategy is also determined.

Proposition 3.1. *Algorithm 3 determines an optimal trading strategy, for a storage system modeled in Model C, to obtain a maximum profit.*

Proof. With the new calculations, Algorithm 2, to determine the optimal trading strategy as described in Chapter 2, is extended. The extension of Model B did influence the calculations to determine the last local minimum and the subsequent local maximum. Like in Algorithm 2, the last local minimum, which is the interval used to charge energy has the absolute minimum price to charge energy, of all intervals available to charge energy, before the subsequent local maximum that is used to discharge this energy. The same applies to discharging. It is not possible to discharge the storage system for a discharge price higher than the discharge price for the interval that is finally used to discharge. This makes the trading strategy optimal, similarly as proven in Chapter 2 for Algorithm 2 for Model B. □

3.5 Structure and complexity

To give a clear overview in this section a flow-diagram of Algorithm 3 is given with the complexity.

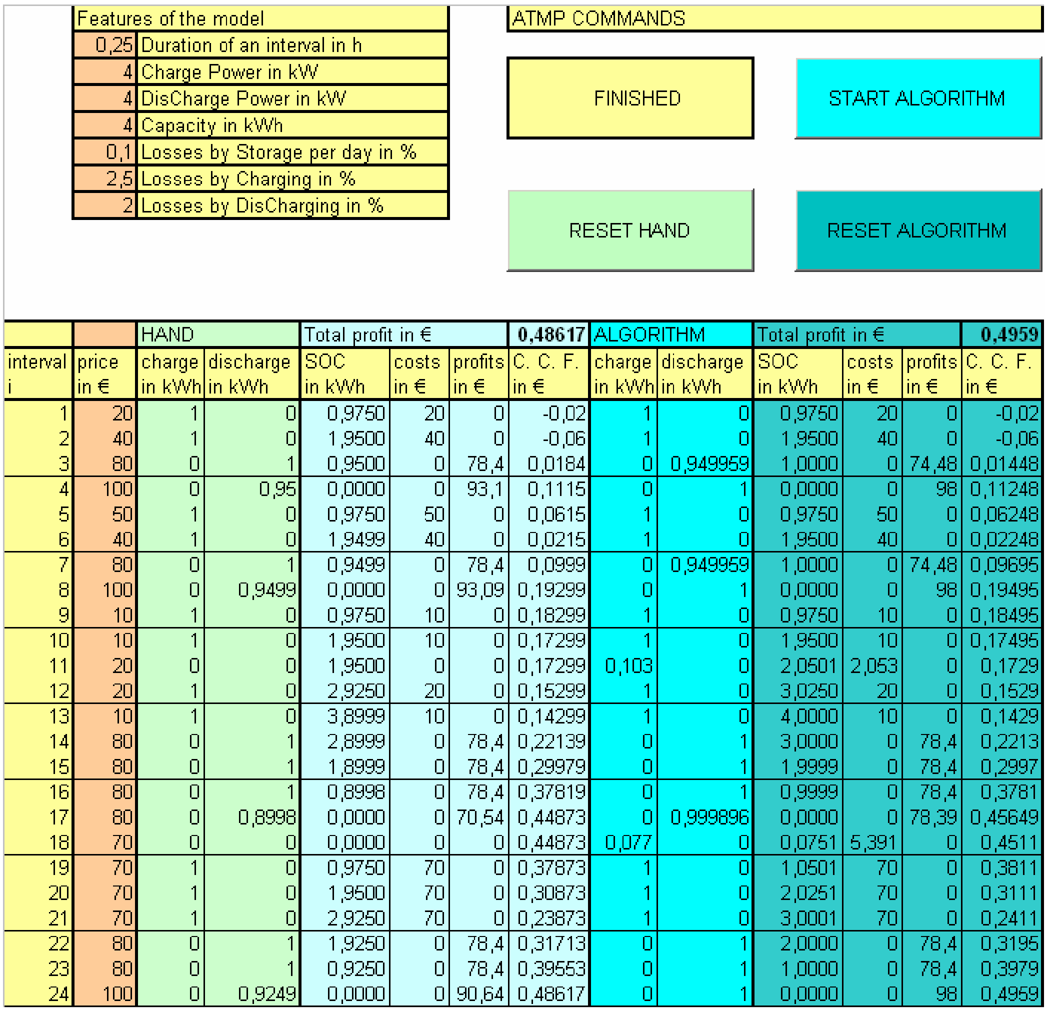
Nr.	Determine	Next Action	
1.	Algorithm 3.1	Algorithm 3.2	$O(n)$
2.	Algorithm 3.2	If there is an LLM; Algorithm 3.3 Else; THE END	$O(n)$
3.	Algorithm 3.3	Algorithm 3.2	$O(n)$

Table 3.2: Flow-diagram of Algorithm 3

As proven in Chapter 2 it is only needed to search for a last local minimum n times, therefore the complexity of Algorithm 3 is $O(n^2)$. This is the same as the complexity of Algorithm 2.

3.6 Reflection and result

With the losses taken into account in this model, it cannot be expected that the profit in this model is larger than the profit in the previous model, for the same price list. But Algorithm 2 does not give a feasible solution to the problem and hence it cannot guarantee optimality in general. If Algorithm 2 is used to determine an optimal profit for a modeled storage system as in Model C, and the storage system is discharged for as much as there is energy stored, to make the solution feasible. This feasible solution will never give a larger profit than can be obtained by the optimal trading strategy determined by Algorithm 3. In Figure 3.9 it is shown how the optimal trading strategy determined by Algorithm 2 is transformed to respect the physical constraints of the modeled storage system. This is done by HAND and therefore the modified solution of Algorithm 2 is called HAND in this example. Algorithm 3 is called Algorithm in this example. In this figure Algorithm 3 and Algorithm 2 can be compared. To give a clear view of the differences in the trading strategies, in Figure 3.10 the price list, the charge and discharge activities, the state of charge and the profit of both algorithms are shown. With the extensions to Algorithm 2, a new algorithm to determine the optimal trading strategy for Model C is found. Though the calculations for finding the last local minimum and the subsequent local maximum became more complex, still it is possible to find an exact solution for this problem as proven above in $O(n^2)$.



Features of the model	
0,25	Duration of an interval in h
4	Charge Power in kW
4	DisCharge Power in kW
4	Capacity in kWh
0,1	Losses by Storage per day in %
2,5	Losses by Charging in %
2	Losses by DisCharging in %

		HAND		Total profit in €			0,48617	ALGORITHM		Total profit in €			0,4959
interval i	price in €	charge in kWh	discharge in kWh	SOC in kWh	costs in €	profits in €	C. C. F. in €	charge in kWh	discharge in kWh	SOC in kWh	costs in €	profits in €	C. C. F. in €
1	20	1	0	0,9750	20	0	-0,02	1	0	0,9750	20	0	-0,02
2	40	1	0	1,9500	40	0	-0,06	1	0	1,9500	40	0	-0,06
3	80	0	1	0,9500	0	78,4	0,0184	0	0,949959	1,0000	0	74,48	0,01448
4	100	0	0,95	0,0000	0	93,1	0,1115	0	1	0,0000	0	98	0,11248
5	50	1	0	0,9750	50	0	0,0615	1	0	0,9750	50	0	0,06248
6	40	1	0	1,9499	40	0	0,0215	1	0	1,9500	40	0	0,02248
7	80	0	1	0,9499	0	78,4	0,0999	0	0,949959	1,0000	0	74,48	0,09695
8	100	0	0,9499	0,0000	0	93,09	0,19299	0	1	0,0000	0	98	0,19495
9	10	1	0	0,9750	10	0	0,18299	1	0	0,9750	10	0	0,18495
10	10	1	0	1,9500	10	0	0,17299	1	0	1,9500	10	0	0,17495
11	20	0	0	1,9500	0	0	0,17299	0,103	0	2,0501	2,053	0	0,1729
12	20	1	0	2,9250	20	0	0,15299	1	0	3,0250	20	0	0,1529
13	10	1	0	3,8999	10	0	0,14299	1	0	4,0000	10	0	0,1429
14	80	0	1	2,8999	0	78,4	0,22139	0	1	3,0000	0	78,4	0,2213
15	80	0	1	1,8999	0	78,4	0,29979	0	1	1,9999	0	78,4	0,2997
16	80	0	1	0,8998	0	78,4	0,37819	0	1	0,9999	0	78,4	0,3781
17	80	0	0,8998	0,0000	0	70,54	0,44873	0	0,999896	0,0000	0	78,39	0,45649
18	70	0	0	0,0000	0	0	0,44873	0,077	0	0,0751	5,391	0	0,4511
19	70	1	0	0,9750	70	0	0,37873	1	0	1,0501	70	0	0,3811
20	70	1	0	1,9500	70	0	0,30873	1	0	2,0251	70	0	0,3111
21	70	1	0	2,9250	70	0	0,23873	1	0	3,0001	70	0	0,2411
22	80	0	1	1,9250	0	78,4	0,31713	0	1	2,0000	0	78,4	0,3195
23	80	0	1	0,9250	0	78,4	0,39553	0	1	1,0000	0	78,4	0,3979
24	100	0	0,9249	0,0000	0	90,64	0,48617	0	1	0,0000	0	98	0,4959

Figure 3.9: The front page of ATMP(3)

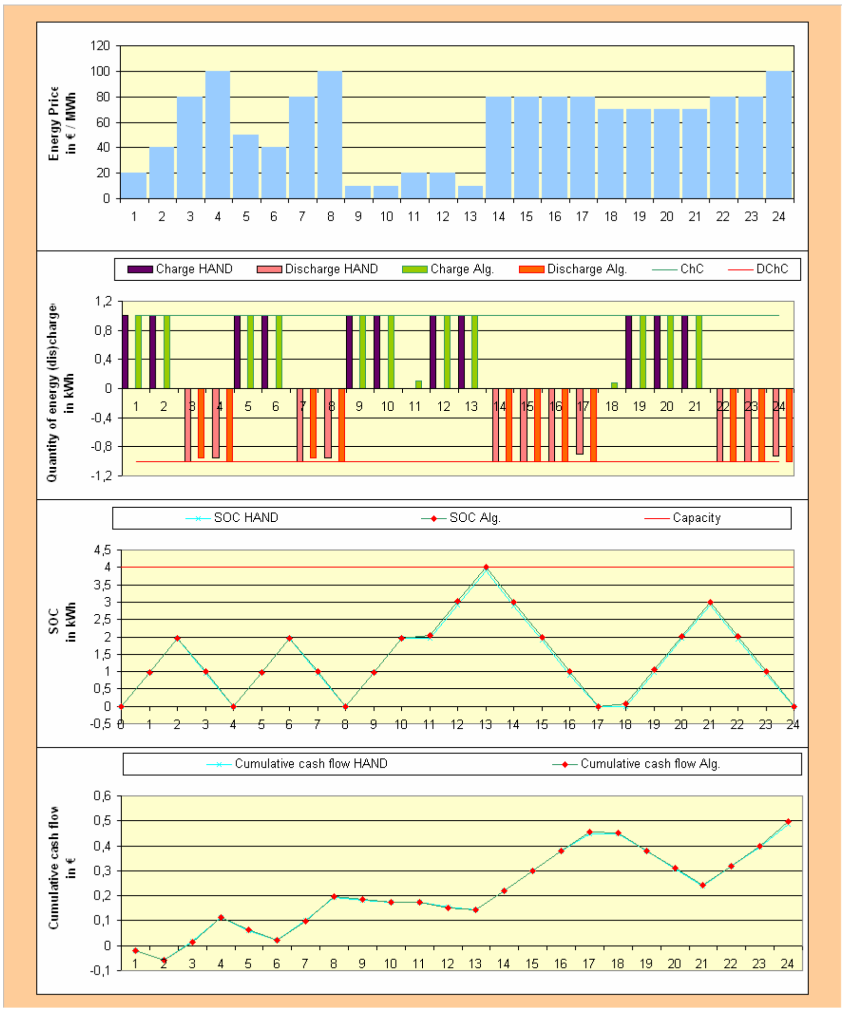


Figure 3.10: The graphs produced by ATMP(3)

Chapter 4

Algorithm 4: Including bounds

In this chapter Model C as described in Chapter 3 is extended by including bounds for the state of charge, yielding Model D to be solved by Algorithm 4. With bounds, the storage system can be used for trading energy as well as for solving problems in the low voltage grid. There are several problems that can be solved using a storage system, as described in Section 7.2. For instance, it is possible that the demand is larger than the supply for a certain period. In such a situation, the storage system can be discharged to help overcome this problem, for a period of time, provided the SOC is high enough. It is also possible that there is more (renewable) energy generated in the low voltage grid than can be used in the low voltage grid at the same time. If this energy cannot be transported to the medium voltage grid, the storage system can help by storing energy. To be able to use the storage system for both of these problems it is required to have a lower bound and an upper bound respectively, for the state of charge.

4.1 Model of the storage system

In Model D, the state of charge of the storage system, SOC, needs to be higher than a lower bound, LB, and less than an upper bound, UB. These bounds are used so the storage system can also be used for other purposes such as solving problems in the electrical grid, besides trading. With this extension to Model C, the final model of the storage system for this research is developed, Model D. In Table 4.1 the parameters and the decision variables used in this chapter are summarized.

4.2 Mathematical model

This optimization problem can be described just as before with a mathematical model. Besides expression (4.3), the following mathematical model is the same as the mathematical model as given in Chapter 3.

Name	Abbreviation	Value	Unit
Energy price for interval i	$p(i)$	input	€/kWh
Charge Capacity	ChC	input	kWh
Discharge Capacity	$DChC$	input	kWh
Capacity of the Storage System	C	input	kWh
Quantity of energy charged in interval i	$ChQ(i)$	$0 \leq ChQ(i) \leq ChC$	kWh
Quantity of energy discharged in interval i	$DChQ(i)$	$DChC \leq DChQ(i) \leq 0$	kWh
State of Charge for interval i	$SOC(i)$	$0 \leq SOC(i) \leq C$	kWh
Charging price for interval i	$cp(i)$	calculated	€/kWh
Discharging price for interval i	$dcp(i)$	calculated	€/kWh
Losses by Charging	LBC	$0 - 100$ % of ChQ	%
Losses by Discharging	$LBDC$	$0 - 100$ % of $DChQ$	%
Losses by Storage	LBS	$0 - 100$ % of SOC	%
Residual after Losses by Charging	$RLBC$	$\frac{100-LBC}{100}$	
Residual after Losses by Discharging	$RLBDC$	$\frac{100-LBDC}{100}$	
Residual after Losses by Storage	$RLBS$	$\frac{100-LBS}{100}$	
Lower Bound	LB	$0 - 100$ % of C	kWh
Upper Bound	UB	$LB - 100$ % of C	kWh

Table 4.1: Parameters and decision variables for Model D

$$\max \sum_{i=1}^{n} (-(DChQ(i) \cdot RLBDC + ChQ(i)) \cdot p(i)) \tag{4.1}$$

$$s.t.\ SOC(i) = \sum_{i=1}^{m} ((ChQ(i) \cdot RLBC + DChQ(i)) \cdot RLBS^{m-i}) \qquad ; 1 \leq i \leq n \tag{4.2}$$

$$LB \leq SOC(i) \leq UB \qquad ; 1 \leq i \leq n \tag{4.3}$$

$$0 \leq ChQ(i) \leq ChC \qquad ; 1 \leq i \leq n \tag{4.4}$$

$$DChC \leq DChQ(i) \leq 0 \qquad ; 1 \leq i \leq n \tag{4.5}$$

This mathematical model shows that the lower bound and the upper bound affect the state of charge as expressed in expression (4.3). The costs because of losses of storage will be larger with a larger lower bound, since the costs are calculated over the state of charge.

4.3 Approach

The bounds for the SOC can be chosen in such a way that the storage system can help to overcome problems, when supply and demand are not coordinated. The lower bound can also be used to not discharge the storage system beyond a given percentage, because of the physical constraints of the storage system. In the previous models a trading strategy was not feasible if not all constraints were met, for instance, it is simply not possible, to charge more energy than the charge capacity can handle. In this model the bounds are set to help overcome problems. It now is possible that these constraints are not met because of the starting condition of the storage system. If possible, it is required to meet the constraints of the bounds of the SOC, but when this is not possible still the best trading strategy can be determined. When it is not possible to get the SOC within the bounds, this is because of the starting condition. First the SOC must be within the bounds and then the trading can begin. In this research it is taken into account that the algorithm to develop an optimal trading strategy, is able to deal with a starting condition with the SOC not within the bounds. This means that when it is not possible to set the SOC within the bounds the algorithm first will (dis)charge energy to get the SOC within the bounds. In the rest of this thesis, the possibility that the SOC is not within the bounds for the first interval(s), is assumed to be taken care of, because once the SOC is within the bounds, this is maintained. For determining an optimal trading strategy, a phased approach is used. There are five algorithms used, one for every phase:

- Determine the SOC of all intervals with the SOC to start,
- Getting the SOC under the upper bound for all intervals,
- Getting the SOC of the last interval equal to the lower bound,
- Getting the SOC of all intervals within the bounds,
- Determine an optimal trading strategy that respects the bounds.

In this section these different phases will be described.

4.3.1 State of charge to start with

As said before, the first concern is to keep the SOC within the bounds. To make the model more realistic, the *start* SOC can have any value between zero and the capacity of the storage system. Now for every interval i, the $SOC(i)$ must be set between the bounds. To get the SOC within the bounds, first the $SOC(i)$ will be calculated for every interval i, by Algorithm 4.1, using the *start* SOC and the losses by storage.

```
Algorithm 4.1
 Declarations
 $SOC(0)$                          the SOC to start with
 $LBS$                             the losses by storage
 $SOC(1) := ... := SOC(n) := 0$

 Program
  : This WHILE LOOP goes chronological through the intervals starting with interval 1 to set
  the SOC for every interval :
  while $i < n + 1$ do
    $SOC(i) := startSOC \cdot LBS^i$
    $i := i + 1$
  end while
```

Algorithm 4.1 is only used to determine the SOC of all intervals, using the $SOC(0)$. To get the SOC within the bounds, Algorithm 4.2 till Algorithm 4.4 are used.

4.3.2 State of charge is less or equal to upper bound

With Algorithm 4.1 we know the $SOC(i)$ for every interval i. The $SOC(i)$ must be less or equal to the UB for every interval i. In Algorithm 4.2 as given in Appendix F, the SOC of the first interval is compared with the UB. If the SOC of this interval is larger than the UB, this interval is set to discharge energy. The amount of energy to be discharged is determined in Algorithm 4.2 using expression (4.6).

$$DChQ(i) := min(DChC - DChQ(i), SOC(i) - UB) \tag{4.6}$$

If for the next interval the SOC is still larger than the UB, again this interval is set to discharge energy and so on. If for an interval j the $SOC(j)$ is less or equal to the UB, all subsequent intervals will have a SOC less or equal to the UB. This is because there is not an interval set to charge energy yet in the algorithms used. This means that all intervals have a SOC smaller or equal to the UB, and thus Algorithm 4.2 is finished.

4.3.3 State of charge of the last interval is equal to lower bound

For interval n, the last interval, it is desirable to have the $SOC(n)$ equal to the LB, since there is no residual value. While for the other models this was always true, in this model it is possible that, because of the *start* SOC there is more energy in the storage system than required. Since there is no residual for energy, it is best to get the $SOC(n)$ equal to the lower bound. That way, there is not more energy in the storage system reserved for solving problems than required. When the $SOC(n)$ is larger than the LB by Algorithm 4.1, all intervals have a SOC that is larger. Therefore it is not desirable to charge energy for solving problems. In this situation, it is best to discharge this extra amount of energy in the first local maximum as defined in Definition 4.1.

Definition 4.1. *A first local maximum is the last interval i, not fully used to discharge energy, of a non-decreasing period for the energy price with $SOC(i) > LB$.*

In Chapter 3 it is shown that some calculations are necessary to find the subsequent local maximum, the same calculations must be done to determine the first local maximum. The new prices for every interval are calculated like in Chapter 3 in Algorithm 3.1. In Algorithm 4.3, given in Appendix G, it is described how the extra amount of energy in interval n is discharged in the first local maximum. In expression (4.7) the calculation required to determine the amount of energy to be discharged in the first local maximum i is given.

$$DChQ(i) := min(DChC - DChQ(i), (SOC(n) - LB) \cdot LBS^{i-n}) \tag{4.7}$$

4.3.4 State of charge within the bounds

If $SOC(n) \geq LB$, all intervals have a SOC that is within the bounds. Else, with the new prices, we can charge energy to get the SOC larger or equal to the lower bound, and keep it less or equal to the upper bound for minimum cost. This is done by checking if the SOC is bigger or equal to the LB for every interval. If the $SOC(j)$ for interval j is less than the LB, in an interval before interval j, including interval j, the storage system needs to charge energy. Now the charging prices will be used to charge for the minimum price. The storage system must be charged for minimum cost, therefore the absolute local minimum until interval j needs to be determined as defined in Definition 4.2.

Definition 4.2. *The absolute local minimum until interval j is the interval with the smallest charging price, that can be used for charging, before interval $j + 1$.*

In the absolute local minimum i, until interval j, the storage system must be charged to get the $SOC(j)$ equal to the LB. The amount of energy that is needed to charge is calculated in expression (4.8). In this expression the losses caused by charging and storage are taken into account. Since it is not possible to charge more energy than possible by the charge capacity, and it is only required to charge energy needed to get the $SOC(j)$ equal to the LB the minimum of these amounts of energy is determined.

$$ChQ(i) := min(ChC - ChQ(i), (LB - SOC(j)) \cdot LBS^{j-i}) \tag{4.8}$$

It is possible that by charging in interval i the SOC for a subsequent interval k, before interval j would become bigger than the UB, therefore this is checked in Algorithm 4.4 as in expression (4.9) for k from interval i till interval j.

$$ChQ(i) := min(ChQ(i), (UB - SOC(k)) \cdot LBS^{i-k}) \tag{4.9}$$

In Algorithm 4.4 as given in Appendix H, it is described how we can charge energy to get the SOC larger or equal to the LB, and to keep the SOC less or equal to the UB with minimum costs. Once all intervals are checked, the SOC is kept within the bounds for all intervals and Algorithm 4.4 is finished.

4.3.5 Optimal trading within the bounds

Now we finally got the SOC of all intervals within the bounds, if this was possible with the *start* SOC. Now a trading strategy can be determined that respects the bounds. To determine an optimal trading strategy, Algorithm 4.5 as given in Appendix I, is developed to be used after Algorithm 4.1 till 4.4. In the Chapters 1 till 3, the only possible strategy for trading was to charge before discharging. This was the only possible way for trading, since there was no energy in the storage system, before the trading strategy started. Now it is possible to discharge before charging since there are intervals with a SOC that is larger than the lower bound, since it can cost less to charge energy for solving problems in advance. The extra energy, that is available in interval k can be used to discharge in interval j, with $j \leq k$. Before the interval for which the SOC becomes less than the lower bound, energy must be charged again. It is possible that it is after interval j, that an amount of energy must be charged again, so that in subsequent intervals the SOC never is less than the lower bound. Therefore a new strategy is needed to trade optimal.

The previous algorithms first determined a last local minimum after which a subsequent local maximum was determined, since it was needed to charge before discharging. If the SOC, for every interval since interval j until interval k, is larger than the LB, with interval j the subsequent local maximum, and $j < k$. Using the strategy of Algorithm 3 makes it possible that the last local minimum i, that is used for charging energy for the subsequent local maximum j, is not the interval with the absolute minimum charging price, available to charge energy to be discharged in interval j, while the absolute local minimum until interval k, as defined in Definition 4.2, has the absolute minimum charging price. In Chapter 1 till 3 the last local minimum always was the interval with the absolute minimum charging price that was available to be used for charging. Now it is better to first determine the first local maximum j as defined in Definition 4.1, and then determine the absolute local minimum i.

The absolute local minimum has to be in a certain domain. The domain of interval j contains all previous intervals $\{1, ..., j-1\}$, all adjacent intervals $\{j, ..., j+h\}$ that have more energy reserved for solving problems in the grid than the lower bound requires, together with the first interval after all these intervals. The domain of interval j is $\{1,...,j+h+1\}$. Only the last interval of the domain is required to determine the absolute local minimum. Therefore the last interval k of the domain is determined as $k := j+h+1$. For this domain interval i, with the smallest new price is the best interval to charge energy to be discharged in interval j. Only if the charging price for the interval with the absolute minimum charging price is less than the discharging price for the first local maximum, the first local maximum will be used to discharge energy.

There are two possible situations now to determine the amount of energy that can be charged and discharged, since there is either energy charged before discharging or there is energy charged after discharging. If the energy is charged after discharging, the energy that can be discharged in j needs to be determined. To keep the SOC for interval m with $j \leq m \leq i$ above the lower bound, the maximum amount to discharge in interval j is determined as in expression (4.10), for every interval m between interval j and interval i, with $DChQ(j) := DChC - DChQ(j)$ to begin with.

$$DChQ(j) := min(DChQ(j), (SOC(h) \cdot LBS^{i-m} - LB) \cdot LBS^{j-i}) \tag{4.10}$$

Charging the storage system again to get the SOC within the bounds for all intervals, will be done as described earlier in Algorithm 4.3. It is only required to use Algorithm 4.3, for the intervals subsequent to interval j. If the energy is charged before discharging, the calculations that are done are the same as in Algorithm 3 in Chapter 3.

4.4 Optimal trading strategy

The five algorithms used to determine an optimal trading strategy are described above. To give a clear view of Algorithm 4, the algorithm to determine an optimal trading strategy for Model D, the flow stream of Algorithm 4 is given in Figure 4.1. The used algorithms 4.1 till 4.5 are described earlier.

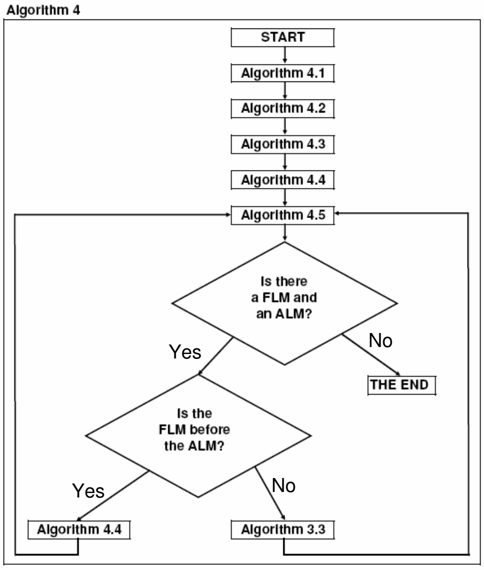


Figure 4.1: Flow stream of Algorithm 4

Algorithm 4 gives a list of intervals to charge energy, with the amount of energy to be charged during these intervals. There is also a list given of intervals to discharge energy, with the amount of energy to be discharged. The profit of this strategy is also determined.

Proposition 4.3. *Algorithm 4 determines an optimal trading strategy for a modeled storage system by Model D, that obtains a maximum profit.*

Proof. Model D, the extension of Model C, required a new approach. In Model D, the SOC must be kept within the bounds. To be able to make an optimal trading strategy, it is wanted to keep the SOC within the bounds for minimum cost. Therefore first the SOC is determined in Algorithm 4.1 for all intervals. To get the SOC below the upper bound, Algorithm 4.2 is used. This Algorithm is used to discharge energy to get the SOC smaller or equal to the UB, regardless of the energy price. In order not to have more energy in the storage system than is required to help overcome problems in the low voltage grid, Algorithm 4.3 is used. This algorithm discharges the amount of energy that is in the storage system, that is not needed for solving problems in the low voltage grid, in the first local maximum, to get the $SOC(n)$ equal to the LB. It is not possible to make a larger profit by discharging the extra energy in the storage system earlier. If it would be better to discharge this energy later, this will be determined by Algorithm 4.5. Algorithm 4.4 is the last algorithm to get the SOC within the bounds for all intervals. Algorithm 4.4 charges energy for minimum cost to get the SOC equal to the lower bound, without getting the SOC above the upper bound. If Algorithm 4.3 is used, and there is energy discharged to get the $SOC(n)$ equal to the LB, Algorithm 4.4 is not used. Since Algorithm 4.3 is used to get the SOC within the bounds with a maximum profit and Algorithm 4.4 is used to get the SOC within the bounds, for a minimum cost, these algorithms give a strategy to keep the SOC within the bounds with minimum cost / maximum profit.

After Algorithm 4.1 till 4.4 are used to get the SOC within the bounds for minimum cost / maximum profit, Algorithm 4.5 is used to determine an optimal trading strategy. Algorithm 4.5 uses a first local maximum as defined in Definition 4.1 and an absolute local minimum as defined in Definition 4.2 to obtain a maximum profit by trading energy. The combination of these two intervals gives the maximum profit possible by trading. Just like in Chapter 1 till 3, the interval to charge energy has a charging price that is the absolute minimum charging price of all intervals that are available to charge energy, for the interval that can be used for discharging. This process is repeated till there cannot be any other first local maximum found. As proven in Chapter 2 it is optimal to use the interval with the absolute minimum charging price of all intervals available to charge energy in, for every iteration to find an optimal trading strategy. As described above, the SOC is kept within the bounds for minimum cost / maximum profit and the trading strategy used gives a maximum profit. Therefore Algorithm 4 gives an optimal trading strategy. □

4.5 Structure and complexity

Algorithm 4 uses 5 algorithms as described above. The structure of Algorithm 4 is clear, and the complexity can be determined by determining the complexity of each algorithm used by Algorithm 4. Like in Chapter 3 an overview of Algorithm 4 is given in Table 4.2. The flow stream is given in Figure 4.1 Also the calculations for the complexity are described in this section.

Algorithm	Complexity
4.1	$O(n)$
4.2	$O(n)$
4.3	$O(n^2)$
4.4	$O(n^2)$
4.5	$O(n^3)$

Table 4.2: Flow-diagram of Algorithm 4

Algorithm 4.1 determines the SOC for all the intervals, using the *start* SOC. This algorithm has complexity $O(n)$.

Algorithm 4.2 will discharge during intervals for which the SOC is larger than the upper bound. This algorithm starts in the first interval i and if the $SOC(i)$ is larger than the upper bound, this interval will be used to discharge energy. After this the $SOC(i+1)$ is determined and this will be repeated till the $SOC(i+1)$ is less or equal to the upper bound. Once the $SOC(i+1)$ is less or equal to the upper bound, the SOC for all the other intervals will be determined. The complexity of Algorithm 4.2 is $O(n)$.

Algorithm 4.3 sets the $SOC(n)$ not larger than the lower bound. For this, the first local maximum is determined. The maximum amount of energy that can be discharged is determined and the SOC for all intervals since the subsequent local maximum and the last interval n are determined. The complexity of determining the subsequent local maximum, the amount of energy to discharge and the SOC for the intervals has complexity $O(n)$. It is possible that all intervals must be used to discharge energy to get the $SOC(n)$ not larger than the lower bound. Therefore Algorithm 4.3 has complexity $O(n^2)$.

If an interval has a SOC that is less than the lower bound, Algorithm 4.4 uses the absolute local minimum for the current interval for charging energy. The required amount of energy must be determined and the SOC of all intervals since the absolute minimum must be determined. To determine the absolute local minimum of an interval, all intervals must be sorted. It is possible to order the intervals in $O(n)$. Just as for Algorithm 4.3, it is now possible that all intervals must be used for charging and thus the complexity of Algorithm 4.4 is $O(n^2)$.

Algorithm 4.5 uses two algorithms to determine an optimal trading strategy. If the first local maximum is after the absolute local minimum, Algorithm 4.5 uses Algorithm 3 to determine the amount of energy to trade. It was already proven that Algorithm 3 has complexity $O(n^2)$ and thus, this part of Algorithm 4.5 has complexity $O(n^2)$. Once the first local maximum is followed by the absolute local minimum, Algorithm 4.4 as described above is used to get the SOC for every interval within the bounds. It is possible that there are $\frac{1}{2} \cdot n$ combinations of a first local maximum followed by an absolute local minimum, as shown in Figure 4.2. In this figure the difference between using Algorithm 4.4 in Algorithm 4.5 and for not using this Algorithm. By Hand the result is given for not using Algorithm 4.4 and the Program is used to show the result with using Algoritmh 4.4. The complexity of Algorithm 4.5 is thus $O(n^2)+O(n^2)\cdot O(n) = O(n^3)$.

The total complexity of Algorithm 4 is the maximum of the complexity of Algorithms 4.1 till 4.5. This is $O(n^3)$, and is not as good as the complexity of Algorithm 3.

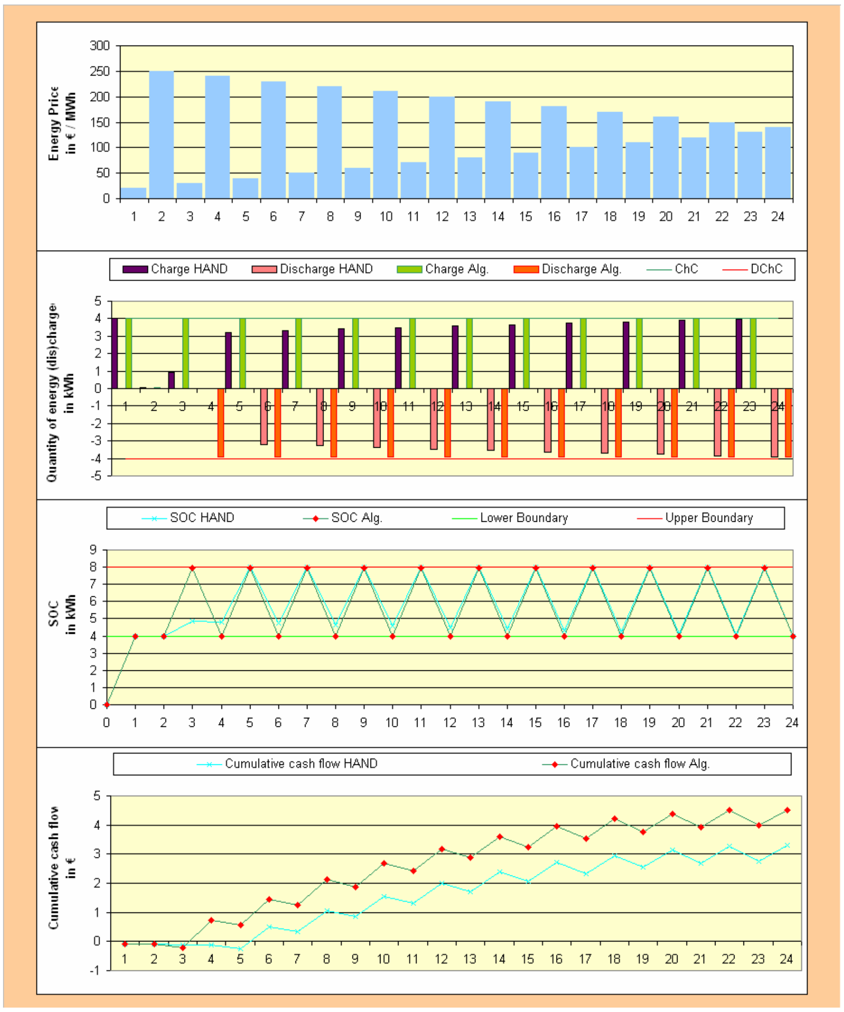


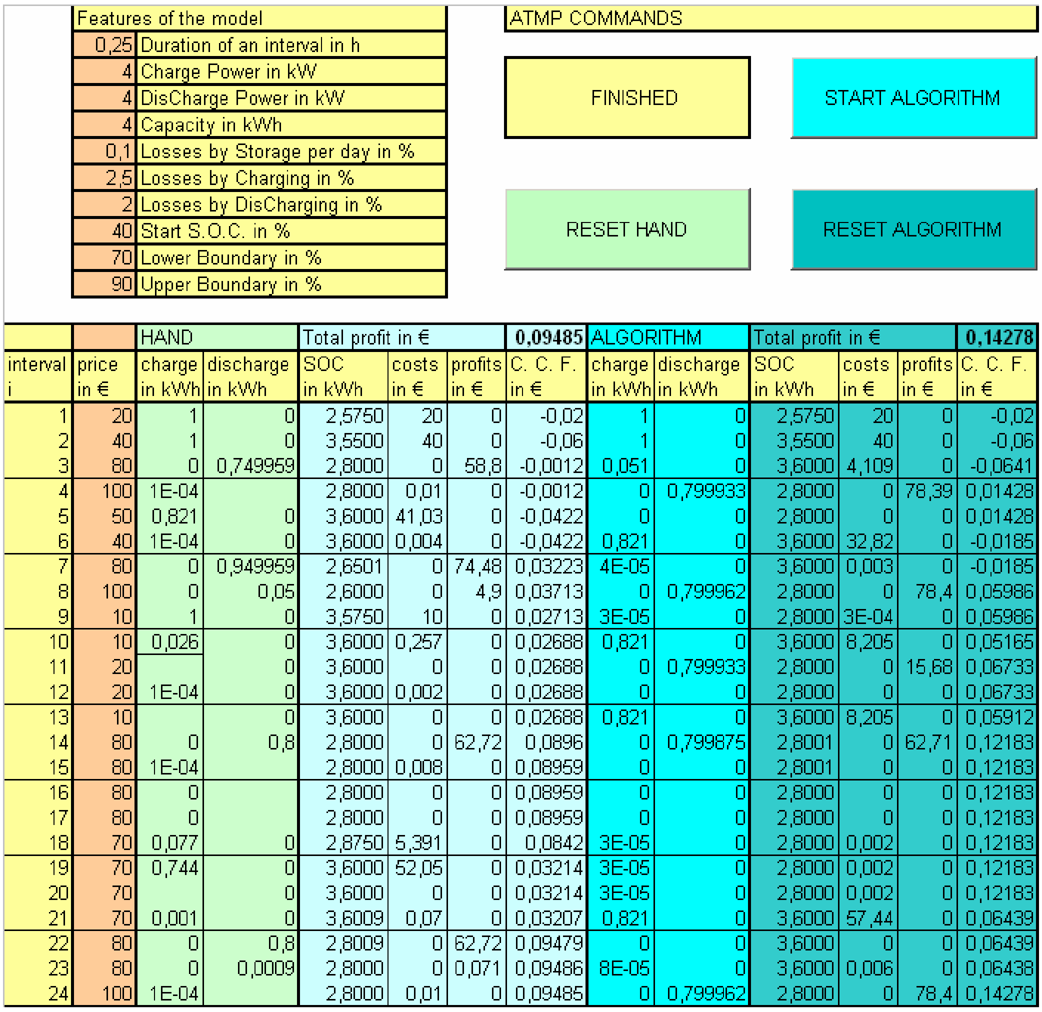
Figure 4.2: With or without use of Algorithm 4.4

4.6 Reflection and result

With the bounds used in the new model in this chapter, there is less storage capacity available for trading. Therefore the profit that can be made is less or equal to the profit that can be obtained with a modeled storage system as in Model C.

It is not expected that a trading strategy, produceded by Algorithm 3 gives a feasible solution for Model D. Once a trading strategy determined by Algorithm 3 is adjusted to the physical constraints of Model D, this trading strategy can be compared to the optimal trading strategy determined by Algorithm 4. The front page of ATMP as shown in Figure 4.3, shows that the adjusted optimal trading strategy obtained by Algorithm 3, called HAND in this example, gives a profit that is less than the profit obtained by Algorithm 4, which is called Algorithm. Of course, when the bounds were not taken into account, both algorithms would give the same solution, which has a profit that is higher than the profit obtained by Algorithm 4 with bounds included. In Figure 4.4 it is shown how these differences affect the charge and discharge activities of the modeled storage systems. Also the effect on the state of charge and the profit is shown in this figure.

With Model D it is possible to use different values for the charge capacity, the discharge capacity, and the capacity of the storage system. The losses by storage, by charging and by discharging are taken into account. And the storage system can be used for solving problems in the low voltage grid, as well as for trading. It is possible to model different sorts of storage systems, with different sorts of physical constraints in Model D. Therefore it is possible to compare these different sorts of storage systems to determine what storage system gives the maximum profit.



Features of the model	
0,25	Duration of an interval in h
4	Charge Power in kW
4	DisCharge Power in kW
4	Capacity in kWh
0,1	Losses by Storage per day in %
2,5	Losses by Charging in %
2	Losses by DisCharging in %
40	Start S.O.C. in %
70	Lower Boundary in %
90	Upper Boundary in %

		HAND		Total profit in €			**0,09485**	ALGORITHM		Total profit in €			**0,14278**
interval i	price in €	charge in kWh	discharge in kWh	SOC in kWh	costs in €	profits in €	C. C. F. in €	charge in kWh	discharge in kWh	SOC in kWh	costs in €	profits in €	C. C. F. in €
1	20	1	0	2,5750	20	0	-0,02	1	0	2,5750	20	0	-0,02
2	40	1	0	3,5500	40	0	-0,06	1	0	3,5500	40	0	-0,06
3	80	0	0,749959	2,8000	0	58,8	-0,0012	0,051	0	3,6000	4,109	0	-0,0641
4	100	1E-04		2,8000	0,01	0	-0,0012	0	0,799933	2,8000	0	78,39	0,01428
5	50	0,821	0	3,6000	41,03	0	-0,0422	0	0	2,8000	0	0	0,01428
6	40	1E-04	0	3,6000	0,004	0	-0,0422	0,821	0	3,6000	32,82	0	-0,0185
7	80	0	0,949959	2,6501	0	74,48	0,03223	4E-05	0	3,6000	0,003	0	-0,0185
8	100	0	0,05	2,6000	0	4,9	0,03713	0	0,799962	2,8000	0	78,4	0,05986
9	10	1	0	3,5750	10	0	0,02713	3E-05	0	2,8000	3E-04	0	0,05986
10	10	0,026	0	3,6000	0,257	0	0,02688	0,821	0	3,6000	8,205	0	0,05165
11	20		0	3,6000	0	0	0,02688	0	0,799933	2,8000	0	15,68	0,06733
12	20	1E-04	0	3,6000	0,002	0	0,02688	0	0	2,8000	0	0	0,06733
13	10		0	3,6000	0	0	0,02688	0,821	0	3,6000	8,205	0	0,05912
14	80	0	0,8	2,8000	0	62,72	0,0896	0	0,799875	2,8001	0	62,71	0,12183
15	80	1E-04		2,8000	0,008	0	0,08959	0	0	2,8001	0	0	0,12183
16	80	0		2,8000	0	0	0,08959	0	0	2,8000	0	0	0,12183
17	80	0		2,8000	0	0	0,08959	0	0	2,8000	0	0	0,12183
18	70	0,077	0	2,8750	5,391	0	0,0842	3E-05	0	2,8000	0,002	0	0,12183
19	70	0,744	0	3,6000	52,05	0	0,03214	3E-05	0	2,8000	0,002	0	0,12183
20	70		0	3,6000	0	0	0,03214	3E-05	0	2,8000	0,002	0	0,12183
21	70	0,001	0	3,6009	0,07	0	0,03207	0,821	0	3,6000	57,44	0	0,06439
22	80	0	0,8	2,8009	0	62,72	0,09479	0	0	3,6000	0	0	0,06439
23	80	0	0,0009	2,8000	0	0,071	0,09486	8E-05	0	3,6000	0,006	0	0,06438
24	100	1E-04		2,8000	0,01	0	0,09485	0	0,799962	2,8000	0	78,4	0,14278

Figure 4.3: The front page of ATMP(4)

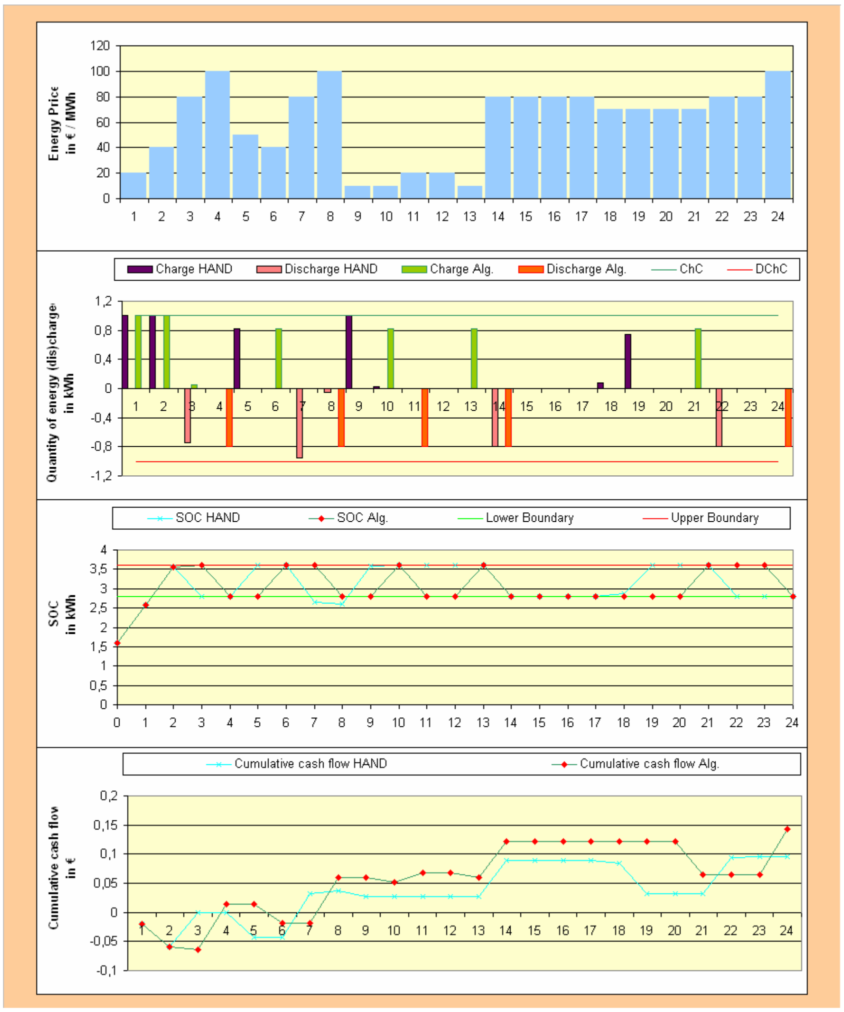


Figure 4.4: The graphs produced by ATMP(4)

Chapter 5

Algorithm 5: Reducing the complexity

In this chapter, Algorithm 4 as described in Chapter 4 is improved by reducing the complexity. To develop an optimal trading strategy for a storage system a phased approached is chosen to give a clear view on the method. Once the aimed goal is reached, it is often possible that another path to reach the aimed goal can be found. Since KEMA is interested in the phased approach in this thesis both the final algorithm as result of the phased approach is described as well as an algorithm that was developed once the aimed goal was reached. The algorithm that was developed after the aimed goal has complexity $O(n^2)$ while the found algorithm by the phased approach as described in Chapter 4 has complexity $O(n^3)$. If the algorithm is used to determine an optimal trading strategy for one day ahead the difference in the complexity between the algorithms is not very interesting, but when the algorithm is used to determine the possible costs for using a storage system to overcome problems in the low voltage grid for more than 5 years, this difference is very important. Therefore the new found algorithm described next is interesting.

5.1 Model of the storage system

Model D as given in Chapter 4 is used and not changed, therefore, the same parameters and decision variables as used in Chapter 4 are used, to give a good overview, they are summarized in Table 5.1.

Name	Abbreviation	Value	Unit
Energy price for interval i	$p(i)$	input	€/kWh
Charge Capacity	ChC	input	kWh
Discharge Capacity	$DChC$	input	kWh
Capacity of the Storage System	C	input	kWh
Quantity of energy charged in interval i	$ChQ(i)$	$0 \leq ChQ(i) \leq ChC$	kWh
Quantity of energy discharged in interval i	$DChQ(i)$	$DChC \leq DChQ(i) \leq 0$	kWh
State of Charge for interval i	$SOC(i)$	$0 \leq SOC(i) \leq C$	kWh
Charging price for interval i	$cp(i)$	calculated	€/kWh
Discharging price for interval i	$dcp(i)$	calculated	€/kWh
Losses by Charging	LBC	$0-100$ % of ChQ	%
Losses by Discharging	$LBDC$	$0-100$ % of $DChQ$	%
Losses by Storage	LBS	$0-100$ % of SOC	%
Residual after Losses by Charging	$RLBC$	$\frac{100-LBC}{100}$	
Residual after Losses by Discharging	$RLBDC$	$\frac{100-LBDC}{100}$	
Residual after Losses by Storage	$RLBS$	$\frac{100-LBS}{100}$	
Lower Bound	LB	$0-100$ % of C	kWh
Upper Bound	UB	$LB-100$ % of C	kWh

Table 5.1: Parameters and decision variables for Model D

5.2 Mathematical model

The optimization problem is the same as in Chapter 4. This optimization problem can be described just as before with a mathematical model.

$$\max \sum_{i=1}^{n} (-(DChQ(i) \cdot RLBDC + ChQ(i)) \cdot p(i)) \tag{5.1}$$

$$s.t.\ SOC(i) = \sum_{i=1}^{m} ((ChQ(i) \cdot RLBC + DChQ(i)) \cdot RLBS^{m-i}) \qquad ; 1 \leq i \leq n \tag{5.2}$$

$$LB \leq SOC(i) \leq UB \qquad ; 1 \leq i \leq n \tag{5.3}$$

$$0 \leq ChQ(i) \leq ChC \qquad ; 1 \leq i \leq n \tag{5.4}$$

$$DChC \leq DChQ(i) \leq 0 \qquad ; 1 \leq i \leq n \tag{5.5}$$

5.3 Approach

As described in Chapter 4, there are 5 algorithms developed to determine an optimal trading strategy. To develop an Algorithm with a smaller complexity, first the algorithm used with the largest complexity is observed. Algorithm 4.5 has complexity $O(n^3)$, since it can use Algorithm 4.4 so often, and thus the SOC of all intervals has to be recalculated to often. If it is possible to use Algorithm 4.4 different, the complexity could be reduced. Combining Algorithm 4.1 with Algorithm 4.4 gives an improvement to Algorithm 4. By first discharging the maximum discharge capacity, for the interval of every iteration, an algorithm to determine an optimal trading strategy, with complexity $O(n^2)$, is developed.

5.4 Optimal trading strategy

Every iteration of Algorithm 5 has five main actions. The first action of iteration i it to discharge in interval i. This means: $ChQ(i) = ChC$. The second action is to determine the $SOC(i)$ as in expression 5.6

$$SOC(i) := SOC(i-1) \cdot RLBS + DChQ(i) \tag{5.6}$$

The third action is to add the charging price of interval i, $cp(i)$ to the sorted list of charging prices. The fourth action is to charge the required amount of energy for the lowest cost. Therefore it is determined if the $SOC(i)$ is smaller than the lower bound. If the $SOC(i)$ is not smaller than the lower bound, the next iteration is started. If the $SOC(i)$ is smaller than the lower bound, the absolute local minimum of interval i, as defined in Definition 4.2, is determined to be used to charge energy to get the $SOC(i)$ equal to the lower bound, just like Algorithm 4.4. The cp(k) of intervals used to charge is if required recalculated using expression 3.20 and the list is sorted again. Once there is no new absolute local minimum or the $SOC(i)$ is equal to the lower bound, the next iteration is started, until the last interval. The flow stream of Algorithm 5 is given in Figure 5.1.

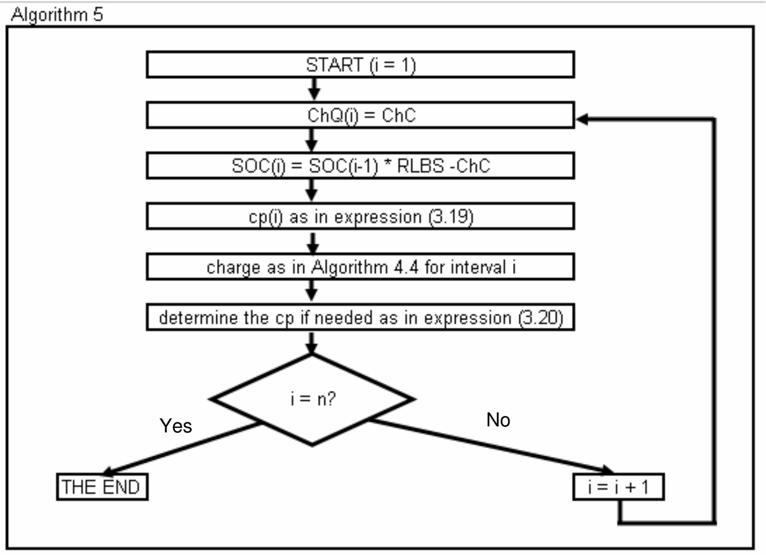

Algorithm 5
START (i = 1)
ChQ(i) = ChC
SOC(i) = SOC(i-1) * RLBS -ChC
cp(i) as in expression (3.19)
charge as in Algorithm 4.4 for interval i
determine the cp if needed as in expression (3.20)
i = n?
Yes
No
THE END
i = i + 1

Figure 5.1: Flow stream of Algorithm 5

Proposition 5.1. *Algorithm 5 determines an optimal trading strategy for a modeled storage system by Model D.*

Proof. Since every interval is first used to discharge the maximum amount of energy that can be discharged by the discharge capacity, also the intervals with a maximum discharge price are used to discharge energy. After there is energy discharged, it is determined if it is required to charge energy. Only if it is required to charge energy, energy will be charged for the absolute minimum price possible. As proven in Chapter 1 it is best to charge for the minimum price, and to discharge for the maximum price to obtain a maximum profit. Therefore Algorithm 5 determines an optimal trading strategy. □

5.5 Structure and complexity

To give a good overview, the structure and the complexity are shown in Table 5.2. The steps for every iteration i are described in this table. The calculations for the complexity are described in this section.

step	Determine	next step	
1.	$ChQ(i) := ChC$	2.	$O(1)$
2.	$SOC(i) := SOC(i-1) \cdot RLBS + DChQ(i)$	3.	$O(1)$
3.	add $cp(i)$ to the sorted list	4.	$O(1)$
4.	charge for minimum costs to get $SOC(i) = LB$	5.	$O(n^2)$
5.	$i := i + 1$	1.	$O(1)$

Table 5.2: Flow-diagram of Algorithm 5

For interval i, first we discharge the total amount of energy that can be discharged during an interval, this is the discharge capacity. After this we determine if the $SOC(i)$ still is larger or equal to the lower bound. The charging price for interval i, $cp(i)$ is determined and this new price is appended to the ordered charging price list. For interval i this is $O(i)$ since for all intervals before interval i the charging prices are already sorted.

If the $SOC(i)$ is larger or equal to the lower bound, Algorithm 5 will continue the process with interval $i + 1$ if $i + 1 \leq n$. If the $SOC(i)$ is not larger or equal to the lower bound, only the amount of energy that is required to get the $SOC(i)$ equal to the lower bound will be charged, for the minimum costs. To determine the complexity normally it would be sufficient to determine the step with the largest complexity, step 4 with complexity $O(n^2)$, and to multiply this with $O(n)$, which is the amount of intervals and thus also the amount of iterations. This would give a complexity of $O(n^3)$. But to determine the complexity of this process, it is required to determine how often in total intervals can be used to charge energy. If for the current interval i to get the $SOC(i)$ equal to the lower bound, all previous intervals are used to charge energy, the intervals are at most 2 times used to charge energy, before they are used for the last interval. This gives that it is possible that there are $3n$ acts of charging wanted to get the SOC within the bounds for all intervals. For every iteration it is also required to determine the new $SOC(j)$ for all intervals j since interval m that is used to charge energy in, till the current interval i. Since it is only possible to have $3n$ iterations, the complexity of Algorithm is $O(n^2)$, this is better than the complexity of Algorithm 4.

5.6 Reflection and result

With the complexity reduced and the same optimal trading strategy determined as before, Algorithm 5 is a real improvement of Algorithm 4.

Chapter 6

Finding a trading strategy using other methods

This chapter will give a description of two heuristics to give a trading strategy. The heuristics described are the heuristics that were looked at before developing an algorithm to give an optimal trading strategy. These heuristics gave a good insight in the problem and were useful to analyse, to see how to develop an algorithm to determine the optimal trading strategy.

6.1 Heuristic A

Heuristic A is a heuristic developed for the GROW-DER'S project by one of the partners. This heuristic was not developed to obtain the maximum profit, it was only intended to be able to use the storage system for trading in the model of the low voltage grid developed by KEMA.This heuristic is included in this thesis to show what elements of this heuristic are used in the other algorithms. Some parts of the heuristic that can be adjusted have been an important source of inspiration. First the heuristic will be described, the complexity of the heuristic will be given and at the end the result of this heuristic will be determined.

The APX gives an energy price list, like in the Chapters 1 till 4, and the *start SOC* is given for the moment the heuristic starts, like in Chapter 4. There also will be costs taken into account for trading that are constant, that is called the costs for trading, CFT. In the Models C and D these costs depend on the amount of energy that was used for trading. There are also losses by charging/discharging taken into account, these losses determine the amount of energy that can be used to be discharged. Therefore this is taken into account in this heuristic by determining what percentage of the amount of energy that is charged can be used to be discharged, this is the percentage left for discharging, LFD.

For this heuristic the price list must be sorted in a list L. This is done in MATLAB with the function "sort". With the *start SOC* the sorted price list L with $L(i) \leq L(j)$ with $1 \leq i \leq j \leq n$ the trading strategy for the storage system can be determined. First it is determined if it is profitable to charge energy for the energy price of the first in row of L and to discharge this energy for the energy price of the last in row of L. After this, it must be determined what the maximum amount of energy is that can be charged. Therefore the SOC and the maximum power that can be used to charge, $Pmin$ are used. The amount of energy that can be discharged is determined by the amount of energy that is charged and the losses by charging/discharging. There is a new SOC determined with use of the SOC and the amount of energy that is charged. After this, for the intervals next in line again it is determined if it is profitable to trade. Once it is not profitable to trade or the storage system is fully used, the heuristic is finished. The code of this heuristic is given in Heuristic A.

Heuristic A

Declarations

C	the Capacity of the storage system
SOC	the State Of Charge of the storage system
$SOCmax$	the maximum SOC
CFT	the costs for trading
LFD	percentage left for discharging
$L(1) \leq \cdots \leq L(n)$	the sorted energy prices
$I(1) \cdots I(n)$	interval in time for the sorted energy prices
$P(I(1)) := \cdots := P(I(n)) := 0$	the Power for interval $I(1) \cdots I(n)$
$Pmin, Pmax$	the minimum Power and the maximum Power for every interval
dT	the time of an interval

Program

```
$i := 0$
$j := 1$
$k := 1$
: This WHILE LOOP is the outer loop :
while $k = 1$ do
  if $L(n-i) - L(j+i) > CFT$ then
    $P(I(j+i)) := -min(-Pmin), \frac{\frac{100-SOC}{100} \cdot C}{dT}$
    $P(I(n-i)) := -P(I(j+i)) \cdot LFD$
    $SOC := SOC - \frac{P(I(j+i)) \cdot dT}{C} \cdot 100$
    if $SOC > SOCmax$ then
      $k := 0$
    end if
  else
    $k := 0$
  end if
  $i := i + 1$
end while
```

The outcome of Heuristic A is a list of intervals with the power that must be delivered by the storage systems during these intervals. In this list both possitive and negative power are included.

This heuristic has complexity $O(n)$ since the heuristic only goes through the sorted price list once. The SOC is determined after the storage system is used for trading. This is not the real SOC since it is possible that the storage system is used for discharging earlier but it is a maximum. The price list can be sorted in $O(n)$ using MATLAB and thus this heuristic has complexity $O(n)$.

It is not guaranteed that this heuristic gives an optimal trading strategy, but analyzing this heuristic gave a start to develop an algorithm that determines an optimal trading strategy.

This heuristic uses the absolute minimum and maximum for charging and discharging. This is the basic of trading and can be used in other algorithms. Heuristic A did not take into account that the storage system will need energy charged before it can discharge. This is a good extension to the problem. In Heuristic A the losses by charging/discharging are taken into account but only for the SOC, not for determining the profit. These are some parts of this heuristic that can be used to develop an algorithm that determines an optimal trading strategy.

6.2 Heuristic B

In this section the optimization problem, to find an optimal trading strategy for a storage system, is described as the well known shortest path problem. This is a well known problem with several algorithms to solve the problem. First there will be a description given of how the problem can be rewritten as a single source shortest path problem. After this, some well known algorithm's that can be used to solve the single source shortest path problem will be discussed. At last we will describe what problems will occur for a more complex model of the storage system.

6.2.1 Rewriting the problem to find an optimal trading strategy

To be able to describe the problem as a shortest path problem, a graph is developed. An example of such a graph is given in Figure 6.1. This graph has vertices, which represent the SOC for a given interval. The SOC is represented in discrete stages. To get a clarifying graph, the vertices are ordered. The vertices for different SOC for one interval are piled in order of size with the smallest SOC at the bottom and the largest SOC at the top. The intervals are next to each other in order of time with the first interval in time in front. The direction of every edge is from an interval i to the interval $i+1$. At $i = 0$, the $SOC = 0$. Charging (buying) energy means "going up" one or more SOC-vertices, limited by the maximum charge capacity. Likewise discharging (selling) energy goes "downward". The edges represent the price to go from an interval with a given SOC to the next interval with a given SOC. In this situation we need to make a difference between charging and discharging for the edges. In this graph, the weight of an edge is the price for energy multiplied with the amount of energy to be charged. It is minus the price for energy multiplied with the amount of energy we want to discharge, and the weight of an edge is 0 when we wish to do nothing.

6.2.2 Well known algorithm's

Let $G := (V, E), |V| = v, |E| = e$, be a graph as described above. Since an edge that is used to discharge energy has a negative weight, it is logical to think of the Bellman-Ford Algorithm, to find the shortest path, since this is an algorithm that can be used for solving the single source shortest path problem for a graph with negative edge's. The upper bound for the running time of this algorithm can be expressed with the big O notation in the amount of vertices and the amount of edges, which would give $O(v \cdot e)$.

It is also possible to solve this single source shortest path problem with dynamic programming. Since the graph is very well ordered it is possible to find an optimal trading strategy in linear time. For every vertex it is determined what edge must be used to obtain the maximum profit. For this graph, the complexity for dynamic programming is $O(e)$.

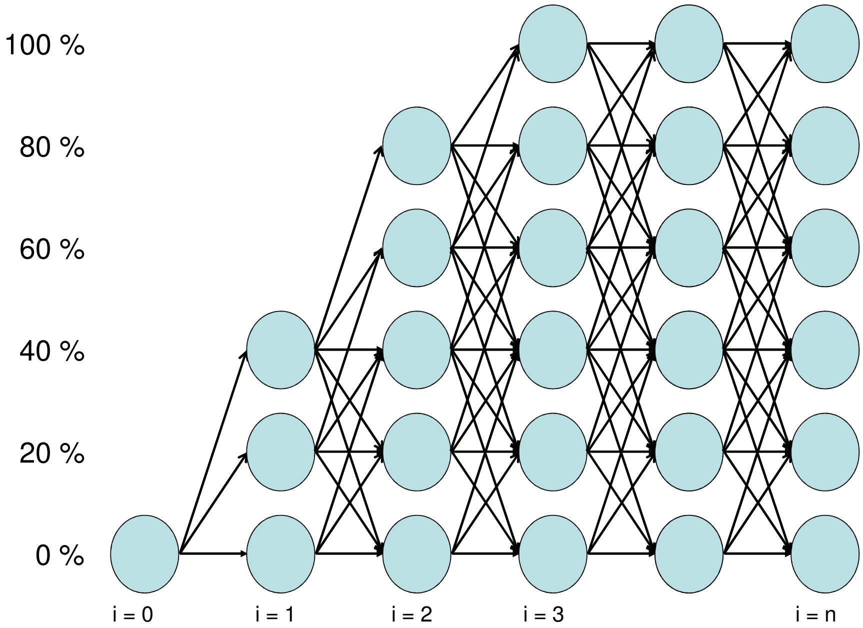


Figure 6.1: Graph of the optimization problem

6.2.3 Problems with more complex models

For the GROW-DER'S project a storage system is used with a charge power of 10 kW, a discharge power of 10 kW and a capacity of 40 kWh. There is a list of energy prices for 24 hours, given by the APX one day ahead. These energy prices calculated by the APX are constant for intervals of 15 minutes, which gives 96 intervals for the graph. Using a storage system with a charge power and a discharge power of 10 kW and a capacity of 40 kWh gives, because of the intervals of 15 minutes, 16 different states of charge. Knowing the amount of the different status of charge and the amount of intervals, we can calculate v for the graph. We use $\#SOC$ as the amount of the different status of charge and n as the amount of intervals.

$$v := (\#SOC \cdot n) - (\#SOC \cdot (\#SOC + 1))$$

This gives a graph with over 1000 nodes. This would give an upper bound of 1000 iterations for finding the shortest path, which gives the optimal trading strategy, using dynamic programming. Since Heuristic B has complexity $O(e)$, this is much better than Algorithm 2 as described in Chapter 2 which has complexity $O(n^2)$, with n still 96 thus about $10,000$ iterations for Algorithm 2. Though this seems very promising, the size of the graph depends on the amount of status of charge. In this situation the charge capacity, the discharge capacity and the capacity of the storage system are very fortunate for Heuristic B. If the charge capacity, the discharge capacity and the capacity of the storage system are chosen less fortunate the amount of status of charge increases drastically.

If we would want to extend the model we could take losses by storage, losses by charging and losses by discharging into account, which would give Model C. Taking these losses into account, the graph becomes more complex. The number of status of charge will grow significantly since it becomes possible that the charge capacity of an interval is partly used. To give an indication of v we use:

$$v := \#SOC \cdot \ n$$

Taking losses into account would have an effect on the upper bound of the running time. The smallest change of the SOC possible must be determined to calculate the $\#SOC$. For example, a small change in the SOC would be charging energy in an interval i to be discharged in interval $i + 2$ because of the losses for charging in interval $i + 1$. The losses caused by charging in interval $i + 1$ will be a percentage of the charged energy, this will be in about 2%. Therefore the number of states of charge will be multiplied with 50. This would give an upper bound of over $\frac{100}{2} \cdot 1000 = 50.000$ iterations for finding the shortest path in this graph and thus the optimal trading strategy using dynamic programming. If we would want to take losses by storage into account, it is more likely that we think of 0.001% losses per day. This would give an upper bound of over $\frac{100}{\frac{0.001}{96}} \cdot 50.000 = 500.000.000.000$ iterations for finding the shortest path. Even with this extension to the graph, the exact optimal solution cannot be found, since this is essentially a continuous problem and not a discrete problem. Therefore it is interesting to find an algorithm that can solve this problem exact, and has a running time that is not depending on the number of status of charge. In the Chapters 1-4 a new algorithm to find an optimal trading strategy for a storage system for which the complexity does not depend on the number of status of charge is described.

Chapter 7

The optimal location in the low voltage grid for a storage system

This chapter describes the overall problem of which the research, in this thesis, is a part of. Though this overall problem is out of scope, it gives a clear view on how the algorithms developed during the research may find their application. In addition, in this chapter a description will be given of interesting research opportunities regarding to the overall problem. Though the research in this thesis is about short term planning for the storage system, the problem in this chapter is about long term planning for the low voltage grid.

First an additional short introduction will be given regarding the low voltage grid. There are many different low voltage grids worldwide. The GROW-DERS project focuses on the low voltage grids in Europe. These low voltage grids will be discussed, to give a good insight in the problems that can be solved by using storage systems. Some of the possible negative effects of distributed generators in the nearby future will be given.

7.1 Introduction

In Europe there was a great variety of low voltage grids. In 1973 the Comité Européen de Normalisation Électrotechnique, CENELEC, was founded. This organization is responsible for European standardization in the area of electrical engineering. The up to date standard European low voltage grid is an Alternating Current, AC, circuit with a frequency of 50 Hz and a voltage of 230 volt (+6%/-10%). Electrical distribution utilities use regulating equipment at electrical substations or along the distribution line, to maintain the voltage at the customer's service within the acceptable range. This regulating equipment is an autotransformer with on-load tap-changers, to adjust the ratio depending on the observed voltage changes. An autotransformer is only used to adjust the long-term average voltage at the service points. It is not actively used to regulate the voltage seen by the utility customer. Though it might solve problems in the low voltage grid, changing the standard, like going from 110 volt to 220 volt in the beginning of the twentieth century, would have great impact now, since there are much more users connected to the low voltage grid in the beginning of the twenty-first century than in the beginning of the twentieth century. Electrical equipment is developed to be connected to the low voltage grid that meets the European Standard. For the lifespan of the electrical equipment it is important that the voltage does not vary too much, for instance voltage dips must be prevented if possible. This means that the power quality must be guaranteed.

The grid is exploited in a way that energy generated in the high voltage grid, is transported in the medium voltage grid and is distributed in the low voltage grid. Therefore the low voltage grid is fed by the medium voltage grid with a transformer. During the twentieth century the electricity grid has grown to a large network. In the Netherlands there is over 100.000 km cable in the medium voltage grid as well as in the low voltage grid, and there is almost 10.000 km cables and overhead lines for the high voltage grid. In the Netherlands most of the network is underground cable, while in the rest of Europe most is overhead line, with the exception of urban areas.

Building the electrical network is a growing process. Not all demographic developments and increases of the demand for electricity were foreseen. Therefore it is possible that the structure of the grid is not optimal. In the beginning of the development of the grid, the main focus was to fulfill the technological reliability. Besides the technological reliability the economic efficiency and at last the environmental quality was taken into account. At the end of the twentieth century the most important characteristic of energy supply decision making was about the economic efficiency, then the environmental quality and at last the technological reliability were taken into account. Now the main focus is the environmental quality, followed by the economic efficiency and at last the technological reliability is taken into account. With environmental quality as an important characteristic of energy supply, there has been a great development in renewable energy. Distributed generators as photovoltaics, PV, windmills and combined heat and power, CHP, had their onset in the low voltage grid. For more details we refer the reader to [2]. A typical layout of a low voltage grid is given in Figure 7.1, this is a radial operated low voltage grid, which is the majority of the low voltage grid's in Europe. In this example there is no distributed generation shown.

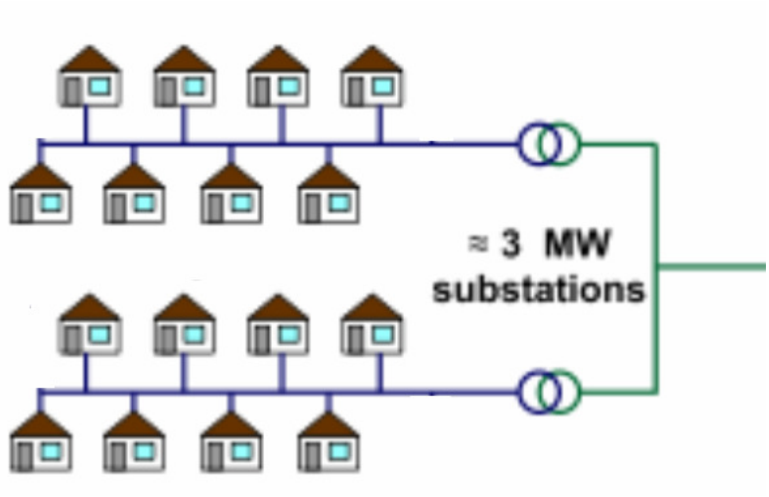


Figure 7.1: An example of a low voltage grid

7.2 Problems in the low voltage grid

KEMA is able to model the low voltage grid and to analyze the low voltage grid with the computer program ”PowerFactory”. An example of a model used in the GROW-DERS project is given in Figure 7.2 and 7.3. In these models distributed generation is included. As can be seen in Figure 7.3 there are more storage systems used in this grid. To prevent these storage systems to work competitive, a partner in the GROW-DERS project developed the EMS, Energy Management System. EMS is out of scope of this research. For trading, storage systems will not become competitive, but it is possible that not all storage systems can be used optimal for trading, when this is limited by the grid.

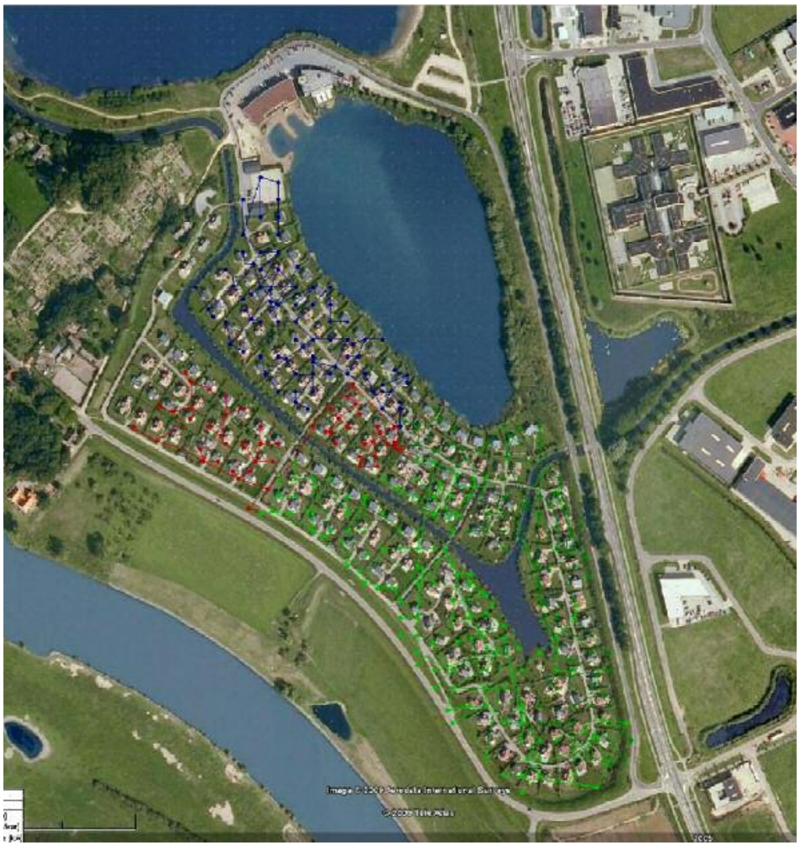


Figure 7.2: The modeled low voltage grid (Bronsbergen)

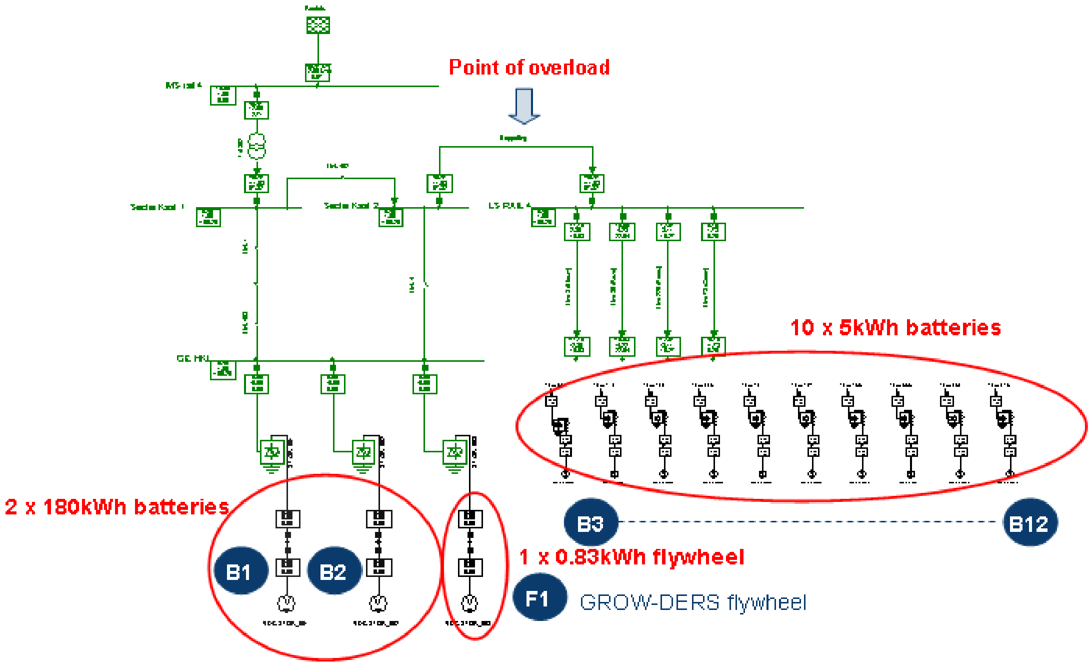


Figure 7.3: The modeled low voltage grid

PowerFactory can determine if there are problems in the low voltage grid concerning the technical reliability. This is a nice tool to use for the long term planning of the low voltage grid. Problems that will occur in the future can be determined in advance and therefore it is possible to make a better planning. With the new developments in the low voltage grid it is important to be able to oversee the effects of these developments.

The problems that occur in the low voltage grid, because of distributed generation and other changes in the low voltage grid, that can be solved using a storage system are described in this section. The problems described in this section are summarized in Table 7.1. There are other problems that occur, that cannot be solved with a storage system, these problems are not in the scope of the GROW-DERS project and will not be described in this thesis.

Problem	Solution	using
Capacity	more capacity	cable/line, transformer
Capacity	peak shaving	storage system
Fluctuating generating	back up supply	storage system
Voltage dips	fast resolving energy supply	storage system
Power Quality	power electronics	part of the storage system

Table 7.1: Problems in the low voltage grid

With the upcoming distributed generation, the function of the low voltage grid is changing. The idea of having many small generators near the consumers was already envisaged by Thomas Edison but not comprehensive applied. In 2020 it is expected that the main supply of energy is distributed generated. Already in Poland more than 25% of the energy comes from distributed generation. With distributed generation, the low voltage grid will get a new functionality. The new functionality of the low voltage grid might give problems since the low voltage grid was not developed for this application.

It is possible that in a low voltage grid the supply of energy is larger than the demand of energy. Because of this the load flow from the low voltage grid to the medium voltage grid might even be larger than the capacity of the transformer. If the capacity of the mv/lv transformer is temporarily exceeded, this problem can be solved by using a storage system to level the load flow. During the overload period the storage system will charge energy that can not be transported. When there is less energy produced by distributed generation, the storage system can discharge the energy that was stored and during this more convenient time the energy can be transported to the medium voltage. Similarly other overloaded components may be relieved, depending on the location of the storage system.

Generation of renewable energy depends on a variety of factors, therefore it is not a constant supply. Because of the fluctuation of renewable energy, there is no technical reliability, and the power quality cannot be guaranteed. For this problem it is possible to use a storage system to level the supply. When there is much renewable energy generated, some energy can be stored for times when there is less renewable energy generated. Doing this gives a more constant supply. When the storage system is used to supply energy and reaches a certain SOC, the centralized generators can be activated to contribute in the supply.

Thanks to the large variety of electrical equipment in the households and the growing industry, the demand of energy is growing. A storage system cannot generate energy but it is able to supply energy when the demand is larger than can be distributed. Therefore it is important that it is possible to distribute enough **energy** while it is not possible to give enough **power** at a certain time. The energy can be stored in the storage system and when the demanded power can not be delivered, the storage system can be used to reach the demanded power.

Due to faults or disturbances in the grid, there may be short periods of time that there is not enough energy. If these periods are very small, this phenomenon is called ”voltage dip”. A storage system may supply the missing energy, using it’s own power controls, provided they respond fast enough. The low voltage grid is fed by the medium voltage grid. The medium voltage grid is sensitive to interruptions because of e.g. excavation. If the medium voltage is interrupted it cannot transport energy to the low voltage grid. Because of this, for a longer period of time, a storage system may act as a back-up power supply, provided that the protection of the grid allows this.

For all these problems, the storage system will reserve capacity. This capacity cannot be used for trading. Therefore in Chapter 4 the SOC has to be within bounds.

7.3 Optimal location

KEMA developed PLATOS for the GROW-DERS project. PLATOS is a program developed in PowerFactory that determines the best locations for storage systems in a low voltage grid. This gives locations for storage systems and types of storage systems that can be used best to help overcome problems. Such a combination of locations and types is called a solution. To be able to determine which solution is better to use, the problems described above are scaled. If a solution helps solving problems, the value of the problems will be linked to this solution. There are also costs for the storage system that are taken into account in PLATOS. To determine the total contribution of the storage system in the low voltage grid, all costs and profits must be taken into account. Therefore Algorithm 4b to determine an optimal trading strategy as described in this thesis will be used in PLATOS to determine the total profit of using a storage system in the low voltage grid.

Using a genetic algorithm, a good location for a storage system in the low voltage grid can be found. This is a very well known heuristic to find a good solution for a combinatorial problem for which there is no structure known. Once there are several solutions found, the best solutions are used to search for new solutions. When the difference between the solutions becomes small enough, the program stops. For every possible solution PLATOS has to do many complex calculations to determine if the new location is a good solution to help overcome the problems in the low voltage grid. This is very time consuming and therefore it is needed to develop a better heuristic to determine what locations can be used best to help overcome the problems in the low voltage grid.

To determine what locations can be used best to help overcome problems, it is important to know where the problems are in the grid, and of course it is important to know what kind of problems there are. PowerFactory can be used to determine the locations of problems in the low voltage grid. For every problem in the low voltage grid it is possible to determine what the problem is, and it is possible to determine if this problem can be solved with a storage system. It can be determined, what locations for storage systems in the low voltage grid can be used to help overcome problems. The type of storage system that is required can be determined.

It is possible that to overcome more problems the same locations for storage systems can be used. The locations that can solve more problems give a possible better solution than separate locations. At the same time it is possible that it is not possible to solve all problems with one storage location even though they all have this location for a storage system as possible solution.

Because of the large amount of variety, it is very hard to find structure in this problem. Therefore it is important to use a phased approach to come to a good solution. By adding structure to the problem the search for a good solution becomes less complex. The amount of possible locations for a storage system decreases and therefore the running time decreases. When the type of storage system is determined in advance for every location, several types of storage systems will not be used to determine if there is a positive result by using this location. The algorithms developed in this thesis can help to reduce the possible types This helps to decrease the running time as well. The structure explained above is taken into PLATOS. The code of PLATOS is KEMA ownership and therefore this is not in this thesis. If it would be possible to determine in advance all locations that cannot be used for a storage system to help overcome the problems in the low voltage grid, the running time of PLATOS would decrease. It would be very interesting to find more locations that cannot be used.

Chapter 8

Conclusions and recommendations

This chapter first describes the assignment, followed by conclusions regarding the results of the research and some recommendations. The algorithm developed during this research is developed to be included in PLATOS.

8.1 Assignment

The assignment is to develop a practical and mathematically correct algorithm that gives an optimal trading strategy for an electricity storage system. For this problem, the solution must meet the physical constraints of the low voltage grid. It was desired to start with a simple model and to make it more realistic using several modeling steps. In the final model the trading strategy must take into account that a storage system has a power to charge, a power to discharge and a capacity to store energy. There will be energy losses by using the storage system. The storage system can be used for trading as well as for solving problems in the low voltage grid, therefore the state of charge of the storage system must be within bounds determined by the grid while the storage system is used for trading. The phased approach gives KEMA the insight they want in all the intermediate results.

In the following overview the conditions of the assignment are summarized:

1. A phased approach is desired to give KEMA the insight they want.
2. For every model a practical algorithm must be developed.
3. Every algorithm must:
   (a) determine an optimal trading strategy for an electricity storage system.
   (b) be proven to be mathematically correct.
   (c) meet the physical constraints of the model of the grid.
   (d) meet the physical constraints of the model of the storage system.
4. The final model of the storage system must contain:
   (a) different values for the power to charge, to discharge and the capacity of the storage system.
   (b) energy losses by using the storage system.

   (c) bounds for the state of charge of the storage system, determined by the grid.

5. Preferably the models are translated into a tool KEMA can use in its projects, and can help to communicate with future users of storage systems

8.2 Conclusions regarding the results of the research

The main conclusion is that all conditions have been fulfilled. This thesis describes a number algorithms that are used in the phased approach to develop the final algorithm that determines an optimal trading strategy for the final model. These algorithms are programmed in Visual Basic Applications Excel, therefore these algorithms can easily be included in PLATOS. This makes it very practical for KEMA to use. For every algorithm this thesis proved that these algorithms determine an optimal trading strategy for the given models. These algorithms act to the constraints of the model of the grid and of the storage system. Chapters 1 till 3 describe several models created during the phases. For every model a new dedicated algorithm was developed that gives an optimal trading strategy.

The final model in Chapter 4 has a storage system with a power to charge, a power to discharge and a capacity to store energy. There are energy losses by using this storage system and this storage system can be used for trading as well as for solving problems in the low voltage grid. This model meets the requirements of the final model determined by KEMA as described above. For this storage system model Algorithm 4a as described in Chapter 4 gives the optimal trading strategy. The complexity of Algorithm 4a is $O(n^3)$. Besides the phased approach, because of insight on resolution, Algorithm 5 also gives an optimal trading strategy. The complexity of Algorithm 5 is $O(n^2)$.

The algorithms in this thesis are used in ATMP, Algorithm for Trading with a Maximum Profit. ATMP is developed to show the user an optimal trading strategy. The effects of this strategy are shown by the charging and discharging activities of the storage system, by the state of charge and by the profit. To make this an interactive program, the user can try to make a better strategy that obtains a better profit. Once a user has tried to develop a strategy to obtain a maximum profit, the user will be convinced that it would be better to use a program to determine an optimal trading strategy. The algorithm used in ATMP can be activated to show the user the optimal trading strategy. This is of course not a proof that the algorithm used determines the optimal trading strategy, but it shows the user that the program is very fast in determining a good strategy. Since PLATOS is very complex and it takes quite some time to get a result, it is important to show the possible users that with ATMP, it is possible to use the storage systems optimal for trading in real time. Therefore it is important to show that within a few seconds ATMP determines an optimal trading strategy. ATMP shows users the possibilities of using a storage system for trading.

To give a clear overview, the conclusions regarding the results of the research are summarized:

1. A phased approach is used to develop an algorithm that determines an optimal trading strategy.
2. For every model a practical algorithm is developed.
3. Every algorithm:
   (a) determines an optimal trading strategy for an electricity storage system.
   (b) is mathematically correct.
   (c) meets the physical constraints of the model of the grid.
   (d) meets the physical constraints of the model of the storage system.
4. The final model of the storage system contains:
   (a) different values for the power to charge, to discharge and the capacity of the storage system.
   (b) energy losses by using the storage system.
   (c) bounds for the state of charge of the storage system, determined by the grid.
5. ATMP is a program that can be used in PLATOS

This overview shows that all conditions of the assignment are fulfilled, still there is always room for extensions and thus there are recommendations.

8.3 Recommendations

The recommendations for further development for ATMP can be applied to different levels:

- details inside ATMP
- development of ATMP
- application of ATMP

For these levels some recommendations to further develop ATMP:

- Though the code of ATMP is checked repeatedly, it is not unthinkable that there might be a little typing error in this code. You can never say that there is not any typing error in a code of over 20 pages, but reviewing and testing the program always helps to improve the code.
- With feedback of users of ATMP the desires of the user can be used to improve this program. New calculations can be included and the interface can be modified.
- Multiple storage systems can be taken into account. Since it is possible to determine an optimal trading strategy for one, it is easy to do this for several but this is not programmed.
- Sequential applications or periodic ($SOC(n) = SOC(0)$)

- The ATMP program can determine an optimal trading strategy for every sort of storage system, therefore this even can be used for storage systems that can be used in the medium voltage or even in the high voltage grid.

For PLATOS, there still are some interesting uncertainties. It can be possible that with further research the running time of PLATOS can decrease drasticcally. One can think of an heuristic that uses:

- calculations to determine the minimum capacity of the storage that must be used.
- knowledge of the grid to determine locations that cannot be used.
- knowledge of multiple storage systems, to determine the possible combinations that can be used.
- etc.

All these extensions to PLATOS could contribute in decreasing the running time for determining the best locations for storage systems in the grid. These extensions should contribute more structure to the problem.

At this moment, the losses taken into account still are very important and cannot be ignored. In the future with the technical developments on electrical storage systems it is possible that these losses become negligible. This would make it possible to determine an optimal trading strategy with dynamic programming (as described in Chapter 6 in Section 6.2) that has complexity $O(n)$.

PLATOS is developed for the low voltage grid but in the future it could be interesting to develop a similar model for the medium or the high voltage grid. Especially with the new research to large-scale energy island offshore energy storage systems. But there are already large energy storage systems such as artificial water reservoirs.

8.4 Reflection

During the research I learned to use known heuristics and algorithms to get a good insight in the problem. Of course there is more than applying known algorithms. When known heuristics and algorithms are not sufficient it is important to be able to expand these heuristics. Working at KEMA gave me the opportunity to contribute in the GROW-DERS project. In this project I learned to contribute in consultations and to present the progress I made. During my internship at KEMA I had the opportunity to learn from colleagues, as well as from my supervisors. This was very instructive for me. This research was a very positive experience for me.

Appendix A

Heuristic 2.1

Declarations

$i = 1..n$	the set of intervals
$p(i)$	the energy price of interval i
$ChQ(1)..ChQ(n)$	a list with the quantity to charge for every interval
$DChQ(1)..DChQ(n)$	a list with the quantity to discharge for every interval

Data & Initialization

$p(1)..p(n)$
$p(n+1) := 0$
$SOC(0) := .. := SOC(n) := 0$
$ChC, DChC, C$
$ChQ(i) \leq ChC$
$DChQ(i) \leq DChC$
$SOC(i) \leq C$

Program

```
$i := 1$
: This WHILE LOOP goes chronologically through the intervals starting with interval 1 :
while $i \leq n$ do
  : IF the current interval is a last local minimum, CHARGE as much as possible :
  if $p(i) < p(i+1)$ and $SOC(i-1) < C$ then
    $ChQ(i) := min(ChC, C - SOC(i-1))$
    $SOC(i) := SOC(i-1) + ChQ(i)$
  else
    : IF the current interval is a subsequent local maximum, DISCHARGE as much as possible :
    if $p(i) > p(i+1)$ and $SOC(i-1) > 0$ then
      $DChQ(i) := min(DChC, SOC(i-1))$
      $SOC(i) := SOC(i-1) - DChQ(i)$
    else
      $SOC(i) := SOC(i-1)$
    end if
  end if
  $i := i + 1$
end while
```

Appendix B

Heuristic 2.2

Declarations

$i = 1..n$	the set of intervals
$p(i)$	the energy price of interval i
$ChQ(1)..ChQ(n)$	the list with the quantity to charge for every interval
$DChQ(1)..DChQ(n)$	the list with the quantity to discharge for every interval
$CHQ, DCHQ, Q$	used to determine the amount of energy to charge, discharge and store
$m = 1$	the local minimum is searched
$m = 0$	the local maximum is searched
$m = 2$	there is no local minimum/maximum found
$m = 3$	the amount of energy to charge and discharge is determined

Data & Initialization

$p(1)..p(n)$

$p(n+1) := 0$

$SOC(0) := .. := SOC(n) := 0$

$ChC, DChC, C$

$ChQ(i) \leq ChC$

$DChQ(i) \leq DChC$

$SOC(i) \leq C$

Program

```
$i := 1$
: This WHILE LOOP goes chronological through the intervals starting with interval 1 :
while $i \leq n$ do
  $m := 1$
  : In this WHILE LOOP, the last local minimum is determined :
  while $m = 1$ do
    if $p(i) < p(i+1)$ and $SOC(i) < C$ then
      $j := i + 1$
      $m := 0$
    else
      $i := i + 1$
    end if
    if $i \geq n$ then
      $m := 2$
    end if
  end while
  : In this WHILE LOOP, the subsequent local maximum is determined :
  while $m = 0$ do
    : In this WHILE LOOP an interval subsequent to the last local minimum, to discharge energy is determined :
    while $DChQ(j) \geq DChC$ and $j \leq n$ do
      $j := j + 1$
    end while
    $k := j + 1$
    : In this WHILE LOOP an other interval subsequent to the previous found interval, to discharge energy is determined :
    while $DChQ(k) \geq DChC$ and $k \leq n + 1$ do
      $k := k + 1$
    end while
    : If there is no subsequent local minimum is found for the last local minimum, this last local minimum cannot be used :
    if $j > n$ then
      $m := 2$
    else
      : If interval(j) has an energy price larger than the energy price of interval(k), interval(j) is the subsequent local maximum :
      if $p(j) > p(k)$ then
        $m := 3$
      else
        : ELSE, for interval(k) must be determined if it is the subsequent local maximum :
        $j := k$
      end if
    end if
  end while
  : If (m=3) there is a last local minimum with a subsequent local maximum determined, therefore the amount to charge and discharge is determined in this part of the program:
  if m=3 then
    $CHQ = min(ChC + DChQ(i), C - SOC(i))$
    $DCHQ = min(DChC - DChQ(j), SOC(j))$
    $Q = min(CHQ, DCHQ)$
    $ChQ(i) = max(0, Q - DChQ(i))$
    $DChQ(i) = max(0, DChQ(i) - Q)$
    $DChQ(j) = DChQ(j) + Q$
    : In this WHILE LOOP for all intervals since the last local minimum, till the subsequent local maximum are determined :
    while $j > i$ do
      $j := j - 1$
      $SOC(j) := SOC(j) + Q$
    end while
  end if
  : The WHILE LOOP begins with the interval after the last local minimum :
  $i := i + 1$
end while
```

Appendix C

Algorithm 2

Declarations

$i = 1..n$	the set of intervals
$p(i)$	the energy price of interval i
$ChQ(1)..ChQ(n)$	the list with the quantity to charge for every interval
$DChQ(1)..DChQ(n)$	the list with the quantity to discharge for every interval
$CHQ, DCHQ, Q$	used to determine the amount of energy to charge, discharge and store

Data & Initialization

$p(1)..p(n)$
$p(n+1) := 0$
$SOC(0) := .. := SOC(n) := 0$
$ChC, DChC, C$
$ChQ(i) \leq ChC$
$DChQ(i) \leq DChC$
$SOC(i) \leq C$

Program

```
$d := 0$
: This WHILE LOOP goes chronologically through the intervals every time starting with interval d :
while $d \leq n$ do
   $i := d + 1$
   $g := 0$
   : In this WHILE LOOP the last local minimum is determined :
   while $g = 0$ do
      : This WHILE LOOP determines the first interval that can be used for charging energy :
      while $ChQ(i) \geq ChC$ or $SOC(i) = C$ and $i \leq n$ do
         $i := i + 1$
      end while
      $k := i + 1$
      $f := 0$
      : In this WHILE LOOP the subsequent interval to charge energy is determined. :
      while $f = 0$ do
         while $ChQ(k) \geq ChC$ and $k \leq n$ do
      $k := k + 1$
    end while
    : If the subsequent interval to charge energy is fully used to discharge energy, there is a special situation. :
    if $DChQ(k) = DChC$ then
      : This subsequent interval cannot be used to discharge energy. Because of this, even if the energy price of the first interval that can be used to charge energy is less than the energy price of this interval, it is possible that there is another interval to charge energy with an energy price less than the first found interval, before the energy to be charged can be discharged :
      if $p(k) > p(i)$ then
        $k := k + 1$
        if $k > n$ then
          $f := 1$
        end if
      else
        $f := 1$
      end if
    else
      $f := 1$
    end if
  end while
  : If the subsequent interval to charge energy has an energy price that is larger than the energy price of the first interval to charge energy, the first interval is the last local minimum. :
  if $p(i) < p(k)$ then
    $g := 1$
  else
    : Else, the subsequent interval to charge energy in is used as the first interval to charge energy. :
    $i := k$
    if $i > n$ then
      $g := 1$
    end if
  end if
end while
: It can be determined in what interval we should start to look for the subsequent local maximum. Therefore we want to be able to start in interval j, if $j > 0$. :
if $j \leq i$ then
  $j := i + 1$
end if
$m := 0$
: In this WHILE LOOP the subsequent local maximum is determined :
while $m := 0$ do
  : In this WHILE LOOP the first interval to discharge energy is determined. :
  while $(DChQ(j) \geq DChC$ or $ChQ(k) > 0)$ and $j \leq n$ do
    $j := j + 1$
  end while
  $k := j + 1$
```

```
    : In this WHILE LOOP the subsequent interval to charge energy is determined. :
    while ($DChQ(k) \geq DChC$ or $ChQ(k) > 0$) and $k \leq n + 1$ do
      $k := k + 1$
    end while
    : If the subsequent interval to discharge energy has an energy price that is less than the
    energy price of the first interval to discharge energy, the first interval is the subsequent
    local maximum. :
    if $p(j) > p(k)$ then
      $m := 1$
    else
      : Else, the subsequent interval to discharge energy is used as the first interval to discharge energy. :
      $j := k$
      if $j > n$ then
        $m := 1$
      end if
    end if
  end while
  : If there is a last local minimum and a subsequent local maximum found, the amount of
  energy to charge, discharge and store must be determined. :
  if $j < n$ then
    $CHQ := min(ChC + DChQ(i) - ChQ(i), C - SOC(i))$
    $DCHQ := min(DChC - DChQ(j), C - SOC(j))$
    $Q := min(CHQ, DCHQ)$
    $k := i + 1$
    :In this WHILE LOOP for all intervals from the LLM to the SLM the maximum amount
    of energy to be stored is determined. :
    while $k < j$ do
      $Q := min(Q, C - SOC(k))$
      $k := k + 1$
    end while
    : If the amount of energy to be charged, discharged and stored is larger than zero, the
    charge quantity, the discharge quantity of the LLM and SLM are determined. :
    if $Q > 0$ then
      $ChQ(i) := max(0, Q - DChQ(i))$
      $DChQ(i) := max(0, DChQ(i) - Q)$
      $DChQ(j) := DChQ(j) + Q$
      $k := j$
      : In this WHILE LOOP, the SOC of all intervals from the LLM to the SLM are
      determined. :
      while $k > i$ do
        $k := k - 1$
        $SOC(k) := SOC(k) + Q$
      end while
    end if
    $d := j$
    :In this WHILE LOOP, the last interval with a $SOC = C$ is determined. :
    while $SOC(d) < C$, and $d > 1$ do
      $d := d - 1$
```

```
        end while
        : In this WHILE LOOP, the first interval that can be used as SLM is determined. :
        while $DChQ(j) \geq DChC$, and $j > d$ do
            $j := j - 1$
        end while
        : Else, there is no LLM and/or no SLM. Therefore we can stop the program. :
    else
        $d := n + 1$
    end if
    $Q := 0$
end while
```

Appendix D

Algorithm 3.2

Data & Initialization

LLM	the last local minimum
SLM	the subsequent local maximum
$cp(1)...cp(n)$	
$dcp(1)...dcp(n+1)$	
$ChQ(1)...ChQ(n)$	
$DChQ(1)...DChQ(n)$	
ChC	
$DChC$	

Program

```
$i := 1$
: In this WHILE LOOP it is determined if there is an interval that can be used to charge energy in. :
while $ChQ(i) \geq ChC$ or $SOC(i) = C$ and $i \leq n$ do
  $i := i + 1$
end while
$k := i + 1$
if $k < n$ then
  $f := 1$
  : In this WHILE LOOP it is determined if there is a last local minimum with a subsequent local maximum. :
  while $f = 1$ do
    if $k > n$ then
      $f := 2$
      $g := 0$
    else
      : If there is an interval that can be used to discharge energy with a profit for this last local minimum, there is a subsequent local maximum. :
      if $DChQ(k) < DChC$ and $dcp(k) > cp(i)$ then
        $s := k$
        $d := 1$
      else
```

```
        : If there is an interval that can be used to charge energy with a price to charge energy that is less than the price to charge energy in the last local minimum, there is a new last local minimum. :
        if $ChQ(k) < ChC$ and $cp(k) < cp(i)$ then
          $i := k$
        end if
      end if
      $k := k + 1$
      while $d = 1$ do
        if $DChQ(k) < DChC$ and $dcp(k) > dcp(s)$ then
          $s := k$
        end if
        if $DChQ(k) < DChC$ and $dcp(k) < dcp(s)$ then
          $d := 0$
          $f := 0$
        end if
        if $ChQ(k) < ChC$ and $cp(k) < dcp(s)$ then
          $d := 0$
          $f := 0$
        end if
        if d $= 0$ then
          $LLM := i$
          $SLM := s$
        end if
        $k := k + 1$
        if $k > n$ then
          $d := 0$
          $f := 0$
        end if
      end while
    end if
  end while
end if
```

Appendix E

Algorithm 3.3

Declarations

$CHQ, DCHQ, Q$ used to determine the amount of energy to charge, discharge and store

b used to see if energy is charged during an interval or if it was already there to

Data & Initialization

j the last local minimum, given by Algorithm 3.2

l the subsequent local maximum, given by Algorithm 3.2

$SOC(j)..SOC(l)$

$DChQ(j), DChQ(l)$

$ChQ(j)$

$DChC, ChC$

Program

```
: If there was energy discharged in interval $j$, the maximum amount of energy that can be used is determined. :
if $DChQ(j) > 0$ then
  $CHQ := min(DChQ(j), C - SOC(j))$
  $b := 0$
  : Else, the maximum amount of energy that can be charged is determined. :
else
  $CHQ := min(ChC - ChQ(j), C - SOC(j))$
  $b := 1$
end if
: the maximum amount of energy that is desired to be able to discharge the maximum amount of energy. :
$DCHQ := \frac{(DChC - DChQ(l))}{RLBC^b \cdot RLBS^{(l-j)}}$
$Q := min(CHQ, DCHQ)$
$i := j + 1$
: The maximum amount of energy that can be stored is determined. :
while $i < l$ do
  if $SOC(i) + Q \cdot RLBC^b \cdot RLBS^{(i-j)} > C$ then
    $Q := \frac{(C - SOC(i))}{RLBC^b \cdot RLBS^{(i-j)}}$
  end if
  $i := i + 1$
```

```
end while
if $b = 0$ then
   $DChQ(j) := DChQ(j) - Q$
   if $DChQ(j) = 0$ then
      : Determine the new price for charging as in 3.20. :
   end if
else
   $ChQ(j) := ChQ(j) + Q$
end if
if DChQ(l) = 0 then
   : Determine the new price for charging as in 3.19. :
end if
$DChQ(l) := DChQ(l) + Q \cdot RLBC^{b} \cdot RLBS^{(l-j)}$
$i := j$
while $i < l$ do
   $SOC(i) := SOC(i) + Q \cdot RLBC^{b} \cdot RLBS^{(i-j)}$
   $i := i + 1$
end while
```

Appendix F

Algorithm 4.2

Declarations

$i = 1..n$	the set of intervals
$DChQ(1)..DChQ(n)$	a list with the quantity to discharge for every interval
$DChC, UB$	
$DChQ(i) \leq DChC$	
$SOC(i) \leq UB$	

Data & Initialization

$SOC(0)..SOC(n)$	determined by Algorithm 4.1

Program

```
$i = 1$
: This WHILE LOOP will discharge energy if the $SOC > UB$ :
while $SOC(i) > UB$ do
   $DChQ(i) = min(DChC, UB - SOC(i))$
   $SOC(i) := SOC(i) - DChQ(i)$
   $i := i + 1$
   $SOC(i) := SOC(i-1) \cdot LBS$
   : IF the $SOC \leq UB$ for the subsequent intervals the SOC can be determined :
   if $SOC(i) \leq UB$ then
      $j := i + 1$
      while $j \leq n$ do
         $SOC(j) := SOC(j-1) \cdot LBS$
         $j = j + 1$
      end while
   end if
end while
```

Appendix G

Algorithm 4.3

Declarations

$i = 1..n$	the set of intervals
$DChQ(1)..DChQ(n)$	a list with the quantity to discharge for every interval

Data & Initialization

$cp(1)..cp(n)$	determined by Algorithm 3.1
$dcp(1)..dcp(n+1)$	determined by Algorithm 3.1
$SOC(0)...SOC(n)$	determined by Algorithm 4.2
$DChC, LB$	
$DChQ(i) \leq DChC$	

Program

```
$i := 1$
$k := 1$
: While the SOC of the last interval is larger than the LB, energy is discharged in the first local maximum :
while $SOC(n) > LB$ do
  $m := 1$
  : This WHILE LOOP determines the first local maximum :
  while $m = 1$ do
    if $k = 1$ then
      while $DChQ(i) \geq DChC$ do
        $i := i + 1$
      end while
      $j := i + 1$
    else
      $j := j + 1$
    end if
    while $DChQ(j) \geq DChC$ do
      $j := j + 1$
    end while
    if $dcp(i) > dcp(j)$ then
      $m := 0$
    else
```

```
      if $dcp(i) = dcp(j)$ then
        $j := j + 1$
        $k := 0$
      else
        $i := j$
      end if
    end if
  end while
  $DCHQ := minimum(DChC - DChQ(i), (SOC(n) - LB) \cdot LBS^{(i-n)})$
  if $DChQ(i) = 0$ then
    : Determine the new price for charging as in 3.20. :
  end if
  $DChQ(i) := DChQ(i) + DCHQ$
  $j := i$
  : This WHILE LOOP will set the SOC for all intervals since the first local maximum :
  while $j \leq n$ do
    $SOC(j) := SOC(j) - DCHQ \cdot LBS^{j-i}$
    $j := j + 1$
  end while
  $i := 1$
end while
```

Appendix H

Algorithm 4.4

Declarations

$i = 1 \ldots n$	the set of intervals
$ChQ(1) \ldots ChQ(n)$	a list with the quantity to charge for every interval
$DChQ(1) \ldots DChQ(n)$	a list with the quantity to discharge for every interval

Data & Initialization

$cp(1) \ldots cp(n)$	determined by Algorithm 4.3
$SOC(1) \ldots SOC(n)$	determined by Algorithm 4.3
$UFC(1) := \ldots := UFC(n) := 0$	a list to know if an interval can be used for charging
ChC, LB and UB	
$ChQ(i) \leq ChC$	
$DChQ(i) \leq DChC$	
$LB \leq SOC(i) \leq UB$	
LBC, $LBDC$ and LBS	

Program

```
$i := 1$
$k := 1$
: This WHILE LOOP will check for all intervals if the SOC is larger or equal to the LB :
while $i \leq n$ do
  : If the SOC is smaller than the LB, if possible, energy will be charged for the smallest price :
  if $SOC(i) < LB$ then
    : This WHILE LOOP will search for the first interval that can be used for charging energy :
    while $ChQ(k) \geq ChC$ or $UFC(k) \neq 0$ and $k \leq i$ do
      $k := k + 1$
    end while
    $j := k + 1$
    : This WHILE LOOP will search the interval with the smallest price to charge energy :
    while $j \leq i$ do
      while $(cp(j) > cp(k)$ or $ChQ(j) \geq ChC)$ and $j \leq i$ do
        $j := j + 1$
      end while
```

```
        if $j \leq i$ then
            $k := j$
            $j := j + 1$
        end if
    end while
    : If there is an interval found that can be used for charging, the amount of energy to charge is determined and with the list UFC the interval that is used for charging is characterized and therefore this interval will not be used again to charge for interval i :
    if $k \leq i$ then
        $UFC(k) := 1$
        $CHQ := min(ChC - ChQ(k), (SOC(j) - LB) \cdot LBC \cdot LBS^{(j-k)}$
        $l := k$
        : This WHILE LOOP is used to determine the maximum amount of energy to charge to not get the SOC larger than the upper bound :
        while $l < i$ do
            if $SOC(l) + CHQ \cdot LBC \cdot LBS^{(l-k)} > UB$ then
                $CHQ := \frac{(UB-SOC(l))}{LBC \cdot LBS^{(l-k)}}$
            end if
            $l := l + 1$
        end while
        : If it is possible to charge energy, the SOC for all intervals since interval k will be determined. :
        if $CHQ > 0$ then
            $ChQ(k) := ChQ(k) + CHQ$
            $m := k$
            while $m \leq n$ do
                $SOC(m) := SOC(m) + CHQ \cdot LBC \cdot LBS^{(m-k)}$
                : If there is an interval with a SOC that is equal to the UB, it is not possible to charge energy in an interval before this interval :
                if $SOC(m) = UB$ then
                    $n := m$
                    while $n \geq 1$ and $UFC(n) < 2$ do
                        $UFC(n) := 2$
                        $n := n - 1$
                    end while
                end if
                $m := m + 1$
            end while
        end if
    else
        : If there is no interval found to charge energy in for interval i, it is not possible to get the SOC in interval i larger or equal to the lower bound :
        $i := i + 1$
    end if
else
    $i := i + 1$
    $j := i$
    : This WHILE LOOP will reset the UFC, till an interval with a SOC that is equal to the UB :
        while $UFC(j) < 2$ do
          $UFC(j) := 0$
          $j := j - 1$
        end while
      end if
    end while
```

Appendix I

Algorithm 4.5

```
Declarations
$i = 1...n$                              the set of intervals
$ChQ(1)...ChQ(n)$
$DChQ(1)...DChQ(n)$

Data & Initialization
$cp(1)...cp(n)$
$dcp(1)...dcp(n)$
$SOC(1)...SOC(n)$
$UFC(1) := ... := UFC(n) := 0$
$ChC$, $LB$ and $UB$
$ChQ(i) \leq ChC$
$DChQ(i) \leq DChC$
$LB \leq SOC(i) \leq UB$
$LBC$, $LBDC$ and $LBS$

Program
 $i := j$
 $DCHQ := (SOC(i) \cdot LBS^{(m-i-1)} - LB) \cdot LBS^{-(m-j-1)}$
 while $i \leq m$ do
    $DCHQ := minimum(DCHQ, (SOC(i) \cdot LBS^{(m-i-1)} - LB) \cdot LBS^{-(m-j-1)})$
 end while
 $DCHQ = minimum(DCHQ, DCHC - DChQ(j), \frac{ChC-ChQ(m)}{(LBC \cdot LBS^{(m-j-1)})})$
```

Appendix J

Algorithm 5

Declarations

$i = 1...n$	the set of intervals
$ChQ(1)...ChQ(n)$	a list with the quantity to charge for every interval
$DChQ(1)...DChQ(n)$	a list with the quantity to discharge for every interval
$ocp(i)$	the charging price ordered
$interval(i)$	the interval of the ordered charging price i

Data & Initialization

$cp(1)...cp(n)$	determined by Algorithm 3.1
$SOC(0)$	*start SOC*
$SOC(1) := .. := SOC(n) := 0$	
$UFC(1) := ... := UFC(n) := 0$	a list to know if an interval can be used for charging
ChC, LB and UB	
$ChQ(i) \leq ChC$	
$DChQ(i) \leq DChC$	
$LB \leq SOC(i) \leq UB$	
LBC, $LBDC$ and LBS	

Program

```
$i := 1$
: This WHILE LOOP is the first outer-loop :
while $i \leq n$ do
  $DChQ(i) := DChC$
  $SOC(i) := SOC(i-1) \cdot LBS - DChC$
  : Determine the new price for charging as in 3.20. :
  $j := 1$
  : This WHILE LOOP determines the place for interval $i$ in the ordered charging price list :
  while $cp(i) < ocp(j)$ and $j < i$ do
    $j := j + 1$
  end while
  $x := i$
  : This WHILE LOOP will set the intervals before interval $i$ in the new place in the ordered charging price list :
  while $x > j$ do
  $ocp(x) := ocp(x-1)$
  $interval(x) := interval(x-1)$
  $x := x-1$
end while
$ocp(x) := cp(i)$
$interval(x) := i$
: If the SOC is smaller than the lower bound, if possible, energy will be charged for the smallest price :
if $SOC(i) < LB$ then
  $q := 0$
  $h := i$
  : This WHILE LOOP will search for the interval with the lowest charging price in the ordered charging price list, that can be used for charging energy :
  while $q = 0$ do
    $k := interval(h)$
    if $ChQ(k) < ChC$ and $UFC(k) = 0$ then
      $q := 1$
    else
      $h := h-1$
      if $h < 1$ then
        $q := 2$
      end if
    end if
  end while
  : If there is an interval found that can be used for charging, the amount of energy to charge is determined and with the list UFC the interval that is used for charging is characterized and therefore this interval will not be used again to charge for interval i :
  if $q = 1$ then
    $UFC(k) := 1$
    if $DChQ(k) > 0$ then
      $CHQ := min(DChQ(k), \frac{(SOC(i)-LB)}{LBC \cdot LBS^{(j-k)}})$
    else
      $CHQ := min(ChC - ChQ(k), \frac{(SOC(i)-LB)}{LBC \cdot LBS^{(j-k)}})$
    end if
    $l := k$
    : This WHILE LOOP is used to determine the maximum amount of energy to charge to not get the SOC larger than the upper bound :
    while $l < i$ do
      if $SOC(l) + CHQ \cdot LBC \cdot LBS^{(l-k)} > UB$ then
        $CHQ := \frac{(UB-SOC(l))}{LBC \cdot LBS^{(l-k)}}$
      end if
      $l := l+1$
    end while
    : If it is possible to charge energy, the SOC for all intervals since interval k will be determined. :
    if $CHQ > 0$ then
      if $DChQ(k) > 0$ then
        $DChQ(k) := DChQ(k) - CHQ$
        if $DChQ(k) = 0$ then
```

```
                  : Determine the new price for charging as in 3.19. :
                  $j := h$
                  while $cp(k) > ocp(j)$ and $j > 0$ do
                     $j := j - 1$
                  end while
                  $x := h$
                  : This WHILE LOOP will set the charging prices from place $j$ till place $h$ in the new place in the ordered charging price list :
                  while $x > j$ do
                     $ocp(x) := ocp(x - 1)$
                     $interval(x) := interval(x - 1)$
                     $x := x - 1$
                  end while
                  $ocp(x) := cp(k)$
                  $interval(x) := k$
               end if
            else
               $ChQ(k) := ChQ(k) + CHQ$
            end if
            $m := k$
            while $m \leq i$ do
               $SOC(m) := SOC(m) + CHQ \cdot LBC \cdot LBS^{(m-k)}$
               : If there is an interval with a SOC that is equal to the UB, it is not possible to charge energy in an interval before this interval :
               if $SOC(m) = UB$ then
                  $n := m$
                  while $n \geq 1$ do
                     $UFC(n) := 2$
                     $n := n - 1$
                  end while
               end if
               $m := m + 1$
            end while
         end if
      else
         : If there is no interval found to charge energy in for interval i, it is not possible to get the SOC in interval i larger or equal to the lower bound :
         $i := i + 1$
      end if
   else
      $j := i$
      : This WHILE LOOP will reset the UFC, till an interval with a SOC that is equal to the UB :
      while $UFC(j) < 2$ do
         $UFC(j) := 0$
         $j := j - 1$
      end while
      $i := i + 1$
   end if
```

```
end while
$i := 1$
: This WHILE LOOP is the outer-loop :
while $i \leq n$ do
    if $cp(i) < 0$ and $ChQ(i) < ChC$ then
        $CHQ := ChC - ChQ(i)$
        $j := i$
        while $j \leq n$ do
            if $CHQ \cdot LBC \cdot LBS^{j-i} + SOC(j) > UB$ then
                $CHQ := \frac{UB - SOC(j)}{LBC \cdot LBS^{j-i}}$
            end if
            $j := j + 1$
        end while
    end if
    $i := i + 1$
end while
```

Contact details

University of Leiden		University of Leiden Faculty of Sience Leiden Institute of Mathematics Niels Bohrweg 1 2333 CA Leiden The Netherlands
	Floske Spieksma	spieksma@math.leidenuniv.nl
	Dion Gijswijt	dion.gijswijt@gmail.com
TU Delft		Delft University of Technology Faculty Electrical Engineering, Mathematics and Computer Science Mekelweg 4 2628 CD Delft The Netherlands
	Karen Aardal	K.I.Aardal@tudelft.nl
KEMA		KEMA Nederland B.V. Utrechtseweg 310 P.O. Box 9035 6800 ET Arnhem The Netherlands
	Gabriël Bloemhof	Gabriel.Bloemhof@kema.com
Author	Arjan Snel	a.c.snel@gmail.com

List of Figures

List of Tables

Bibliography

[1] Waumans R.J.R.: *Openbare netten voor elektriciteitsdistributie*, ISBN 90 201 1887 0, (1986)

[2] Combrink F.M. and Vaessen P.T.M.: *Low voltage, but high tension*, (2006)

[3] Kallenberg LC.M.: *Besliskunde 1*, Dictaat

[4] Kallenberg LC.M.: *Besliskunde 2*, Dictaat

[5] Kallenberg LC.M.: *Besliskunde 3*, Dictaat

[6] http://www.growders.eu/

[7] http://www.apxendex.com/

[8] http://intouch.kema.intra/Pages/Default.aspx