A robust approach to Meta-EMS

Author:
Robin Swinkels BSc

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Operations Research and Management Science.

Supervisors:
Prof.Dr.Ir. Dick den Hertog
Frank Phillipson MSc PDEng
Bas Marban PhD

Tilburg School of Economics and Management

Tilburg University

October 10, 2012

Abstract

This study was concerned with the meta-Energy Mediation System (meta-EMS) problem. We consider several companies from the horticulture industry, for which we have to match supply and demand of different commodities a few days ahead. We have introduced the problem and considered the relevant literature for the current subject. The problem is formulated as an MIP-model, and we have shown it to be NP-complete. The performance measure was minimal costs, and we have shown our cooperative model to improve on the current, non-cooperative situation. The computation time is exponential with the size of the instance. We have solved given test cases to optimality and improved on the heuristic that was made for these cases. Next, we have constructed our own, more complicated test cases. The model runs fast enough for models of smaller size, but when the size increases some heuristics might be required. We have proposed two heuristics, one based on aborting the solver and obtaining a very good bound, the other based on an LP-relaxation. A sensitivity analysis was performed and it appeared that that the model is insensitive with respect to changes in demand. However, the model is not robust with respect to changes in demand, which is shown by a Monte-Carlo analysis. We have given a small introduction to robustness and we have made the model robust against changes in demand by implementing the Affinely Adjustable Robust Counterpart (AARC). This improves on the worst case RC solution, but the size of the improvement is dependent on which portion of data is used to adjust the variables to the uncertain demand. Our research indicates that the most effective improvement is given when taking the current time step as the portion of data. This approach is suitable for small models. When the number of companies or time steps increases, the computation time becomes too large.

Acknowledgments

This thesis would not have been written without a internship possibility at TNO. Here I have enjoyed a nice atmosphere and it was good to be able to take part in department meetings and special workshops designed for students. It was a good place to work on this thesis. My thanks go to many people. Thanks to Dick for the bilas. To Frank for coming up with the subject and your sense of humor. To Daniel for helping me in the first months of my thesis and for being a good roommate. To Bas for your helpful comments on my thesis and the walks to the coffee machine. To Jacob for always being enthusiastic, to Trivik for taking me climbing, to Laura for always being cheerful, to Timo for inspiring me to work hard, to Sjoerd for the stars on my whiteboard, and to all the others of the department.

Thanks to Tilburg University, and specifically to prof. Dick den Hertog for taking your time and extensively commenting on my thesis, which was encouraging and very helpful. I would also like to thank prof. Ashayeri who has agreed to take part in the examination committee.

Thanks to my friends and family who have supported me during the process of writing my thesis. Special thanks go to Marieke who has supported me with a lot of patience and understanding.

Contents

Chapter 1

Introduction

1.1 Energy and electricity

The demand for energy and electricity keeps on growing. This is caused by the increase in world population, the growth in the number of devices that use electricity and the increase in usage of electrical devices. On the other hand, the supply also keeps on growing. This has resulted and will result in changes within the energy and electricity sector.

Because of environmental issues, but also because resources like oil and gas will deplete at some moment in time, it is becoming more and more popular to generate energy in other ways than by fossil fuel generation. Well-known energy sources are solar and wind energy. Slightly less well-known is cogeneration, or combined heat and power (CHP). In this case, heat is generated, for example by gas, as commonly done in the Netherlands. Some of the heat is also used to generate electricity. This leads to a huge increase in efficiency, namely 85-90%. If only electricity is produced, only half of this efficieny can be achieved, according to El Bakari, Myrzik and Kling [3].

There are not only some changes on the demand side, but also on the supply side of energy and electricity changes occur. The share of the renewable energy resources is increasing while the energy is more and more generated in a decentral way. This has some consequences. The output of electricity from renewable energy sources, especially wind and solar energy, fluctuates a lot. It is harder to predict and it cannot always be produced, for instance if the sun does not shine. It will become increasingly difficult to match this with the also fluctuating demand. The number of units producing electricity grows fast as a result of the decentral generation. Obviously, also the location of the small-scale energy generators is changing, leading to a smaller distance between generation and consumption of energy. However, the distance from large-scale generation to consumption of energy is increasing, because huge wind parks are built off-shore and large solar power plants are built in desert areas, according to Kok [14].

In this thesis, we focus on the horticulture industry, where the CHP was introduced as an energy-efficient way to give enough heat and light to the plants. The co-produced and leftover electricity can be sold back to the net. When the demand for energy keeps increasing, in the future a micro-CHP could become popular in every household. If this would be the case, then we would have millions of households supplying energy. An anticipation of this trend is necessary, because supply and demand of electricity need to be equal at every moment in time. Therefore, energy-efficiency is an important subject to study.

The thesis is written at TNO (Nederlandse Organisatie voor Toegepast Natuurwetenschappelijk Onderzoek), an independent research organization, mainly based in the Netherlands, with a focus on "connecting people and knowledge to create innovations that boost the sustainable competitive strength of industry and the well-being of society". One of their key areas of expertise is energy. Within the department 'Performance of Networks and Systems', a project in this area was performed, but there were still some open questions. This thesis continues the research in this area, by building a more extensive model which takes more factors into account but still is generic. In this case, generic means that it should applicable to different companies with a CHP and be flexible with respect to different input data. Moreover, the model should be fast enough.

1.2 The problem in general

In this section we briefly describe the general problem that we consider in this thesis. We consider several (about 20) companies from the horticulture industry, for which we need to match supply and demand of different commodities with a planning of a few days ahead. All companies we are considering have a CHP. They can import gas from a power plant. By using the gas in the CHP, they produce three commodities, namely heat, electricity and CO_2. Although all three are necessary, the main product is heat and the other two are byproducts. Commodities can be sold to other companies, or put in a storage for later use. Any leftovers can be sold back to the net. Any shortage can be ordered from an independent power plant. The power plant also uses gas to produce heat, electricity and CO_2. Thus, it is also possible to purchase commodities directly from the power plant, but this is relatively expensive. Prices for buying and selling commodities are fixed. In reality the three commodities are imported from three different sources, but for simplification we have assumed that it comes from one and the same power plant. This does not restrict the model in any way.

The problem just described can be seen as a system of companies which all have certain demand requirements for each of the three commodities. The companies also can import gas, electricity, heat and CO_2, and sell the latter three to each other or back to the net.

The objective is to minimize the total cost for this Meta-Energy Mediation System (Meta-EMS). The decisions in this problem can be put into two categories: CHP and transport. The CHP decisions are: which CHP should run, at what moment in time and at what speed? The transport decisions include all the importing from and exporting to the power plant, storage, and other companies.

Because we are considering a system of about 20 horticulture companies, already a small improvement of the solution can have a big impact on the profits. A solution which is just a few percent better than the default option results in a big change of the profits.

Because not all data are certain, it can happen that the found (nearly-) optimal solution is very vulnerable with respect to a small change in some of the parameters. To prevent this, we also develop a more robust approach for the model, thus modeling the problem while taking a certain degree of robustness of the solution into account. The 'traditional' robustness method is a bit too conservative and only optimizes the worst case. We make the model affinely adjustable robust, which is a generalization of the robust model and it allows some of our decision variables to be adjustable with respect to the uncertain parameters.

1.3 Research questions

This section discusses the research questions of this thesis. First, we would like to model the problem mathematically. This raises questions about the computation time to solve the model. If the complexity of the model grows, by increasing the number of companies or time steps, it will take an increasing amount of time to solve the problem. At some point this computation time will become too large. Moreover, we are interested in the sensitivity and the robustness of the outcomes. Our main goal is to *construct a mathematically robust model for the horticulture industry problem with uncertain demand.*

We can now formulate the following research questions and goals:

- We would like to construct a mathematical optimization model for the underlying problem.
- What are the computational aspects of this problem? How difficult is it?
- How does the computation time relate to the amount of companies and time steps?
- What are good heuristics and methods to solve the underlying problem quickly?
- How sensitive and robust is the nominal solution for different types of instances?
- If the model is not robust, how can we account for this?

1.4 Thesis outline

We start with a literature review in Chapter 2. A thorough study of the current literature is done. Many papers have been written about cogeneration, distributed generation and energy efficiency. We have studied this literature, summarized it and implemented some useful assumptions.

In this thesis, we present a good working model for the current problem. Chapter 3 describes this model. It starts with a detailed problem definition and a list of the assumptions made, followed by the corresponding mathematical model.

Chapter 4 discusses the numerical results of the current research. The test problems are introduced, we calculate the effectiveness of the model by considering the computation time, or comparing it to relaxations. We have also discussed ways to improve the time required for solving the model. Moreover, this chapter contains a sensitivity analysis and a Monte-Carlo analysis for robustness.

In Chapter 5, we give an introduction to robustness. Then we introduce the concept of Adjustable Robust Optimization. We describe how this is implemented to our model and we discuss the results. Finally, Chapter 6 concludes this thesis and gives recommendations for further research.

Chapter 2

Literature

This chapter is divided into two parts. Section 2.1 summarizes the relevant literature about the current subject, which type of models are used and which assumptions are made. Section 2.2 is about what can be useful for, or implemented in, our model.

2.1 Literature summary

In this section we are going to discuss the present literature about energy efficiency. Energy efficiency is the goal of reducing the amount of energy used in making products and providing services. Here we will give a short summary of relevant papers. As energy efficiency is a very broad subject, there is a lot of literature on it and it seems like every paper looks at a different aspect of the complex problem. A good summary of different aspects of energy efficiency can be found in Bosman et al. [6]. The different aspects mentioned in this paper are: production, distribution, consumption, storage and load management. More specifically, we will focus on distributed generation and cogeneration.

First, let us focus on the dissertation of Molderink [19]. In this dissertation, smart grids are discussed. Smart grids are electrical grids that gather and use the information of their participants to improve the efficiency of electrical services. On a micro-level, the author discusses the possibilities of implementing smart grids in a domestic environment. All issues concerning smart grids are discussed extensively. However, implementing a smart grid only makes sense if there is an algorithm to control the devices connected to it. Therefore, the core of the dissertation is a three-step control methodology. The first step consists of a local offline prediction. This is a hierarchical, bottom-up process. For every device a predicted energy profile is generated using so-called 'neural networks', based on the historical usage pattern of the residents and external factors like the weather. The second step is a global offline planning, which is hierarchical, top-down and iterative. A global offline planning is made. The root node sends steering signals or objective bounds. The children respond by adapting their profile and send it back upwards. This goes on iteratively. Hence, a planning for the coming day is made as well as a general idea of the global electricity streams. The third step is the local real-time control. Here a real-time control algorithm decides when devices should be switched on or off, how much energy flows from where to where, etcetera. Dynamic programming is used to find a local planning. This is fed back to reduce the global mismatch as much as possible. The local real-time control has a major drawback since it is computationally intensive.

Next, let us discuss the Power Matcher, on which various papers ([2], [16], [17], [14], [15]) are written. Akkermans et al. [2] consider the agent-based microeconomic control model of a market. This means that they are looking to the market from a microeconomic viewpoint, and every actor or building is represented by an agent. A proof on the optimality of this model is given. They show that for this type of Multi-Agent System the market-based solution is equal to that of a centralized omniscient optimizer. There are no special conditions or very strict assumptions and the proof is given for a generic case. This model can be applied to practice and is the base foundation of the Power Matcher.

In Kok et al. [16], several subjects like the background of electricity, how the electricity net works, how imbalances are prevented, etcetera, are discussed. Also the problems which can arise are addressed. The authors elaborate a bit on sustainable energy sources and distributed generation and the paper ends explaining how the power matcher works. In short, every agent has to hand in a demand/price curve. Supply is seen as negative demand. The aggregation of all curves leads to a (virtual) equilibrium price. This is the price that clears the market. An important assumption is that they incorporate electricity storage devices, which are not really used yet. Storing big amounts of electricity is not much done yet due to the large losses that occur at converting. Therefore, it is hardly profitable. The authors admit that this can only be economically viable when this is a reaction on a time-varying electricity tariff, thus buying at the times that it is cheap, storing it and selling when the prices are high (Kok et al. [16], page 22). This can be a reaction on changing production of wind and solar energy. Of course, in the future this could change. At some point storing electricity might become viable. The authors further mention that the power lost in long-range electricity transmission is about 7%. When electricity is produced nearby the consumption area, transport losses are avoided.

In Kok et al. [17], the principle of matching is elaborated more extensively, with some examples. The price is updated every two minutes or more, which seems reasonable and lowers the amount of data communicated. A simulation is performed to check the validity of the concept. The result is an on average 40% lower production of the generator and a reduction of approximately 45% in the peak.

Kok [14] elaborates more about the bids and how they are created. Depending on the type of device, these bids can be completely different. For instance, for a gas generator the bidding just depends on the marginal costs, while in case of a battery the bidding depends on historical prices. These are the extreme ends of the bidding spectrum, the rest is more in-between these two extremes and the bids are more complex for a CHP for instance. In case of a CHP combined with a heat buffer there are different subcases. There are three different threshold values. One is a critically low buffer level, there is one low level and

one high level. If the buffer level is critically low, the CHP and heater should be switched on, no matter the cost. If the buffer is between this critical level and the lower level, there is a demand of heat and this has to be fulfilled by either the CHP or the heater. Above the high buffer level, there is no heat demand. There is a choice of running the CHP for electricity and dumping the heat. This depends on the marginal cost for electricity. Finally, if the buffer level is between the high level and the low level, there is a high degree of freedom. This depends on the risk-aversity. The authors also do not use any penalty for dumping the resulting heat. Note that in this paper CO_2 is not taken into account.
The author elaborates about two simulations, one of which was already discussed in the paper of Kok et al. [16]. The other simulation is for 5 CHPs. The author claims that the Power Matcher concept is successful and will be applied to a large Power Matcher project.

The paper of Kok [15] is another Power Matcher paper. A large part is a copy of their previous papers. The authors have done the following assumptions for scalability:

- No peer to peer communication, only with the auctioneer.
- Auctioneer is trusted and the communication is only done once (just hand in your full demand curve at once).
- Use of a hierarchical structure to aggregate different demands more quickly.

This implies central decision making, which is done to decrease the amount of computations and the complexity of their model.

Concluding, the following remarks about the Power Matcher can be made: Every agent has to hand in a demand curve. The curves show how much the companies are willing to supply or receive at every price. Then the Power Matcher aggregates all these curves and finds the equilibrium price. It could be seen as 'auctioning' the price, but in a very fast way. The authors prove that this is the Pareto-optimal solution. This means that there does not exist any other solution which is better or neutral for all other agents. The proof is based on utility functions, for which some assumptions need to be made. These assumptions are reasonable and the proof is convincing. However, we think that this proof is a bit misleading. Only given the demand curves, the solution given by the Power Matcher is optimal. In the paper of Kok [14] there is quite a bit of elaboration about the bids and the corresponding demand curves. Difficulties that occur in our situation, like using different losses in the model, can be dealt with, for instance by including these in the marginal costs. Also the option of selling to another company could be incorporated as the price for which this happens does not matter. Still, we do not see why the Power Matcher would be optimal for the whole system. The demand curves described take only the considered CHP into account and only optimize for themselves. Moreover, if the Power Matcher were optimal for the whole system, the authors would provide an optimal solution to a NP-hard problem in very short time. This seems highly unlikely. The difficulty is in producing these demand curves. Given optimal curves, the Power Matcher gives an optimal solution.

The paper of Wright and Firth [23] is about calculation of domestic energy profiles. Alanne and Saari [3] mainly focus on CHP for households, and the obstacles and solutions for implementing small-scale CHP. The paper of El Bakari et al. [10] focuses on the virtual power plant. They aggregate different Distributed Generation units into a Virtual Power Plant (VPP) and discuss how this VPP should be combined with the electricity grid.

Can we consider the group of horticulture companies as a huge VPP? Bosman et al. [6] first define the Micro-CHP planning problem for one household, which is similar to the CHP problem for one horticulture company. Then they solve this problem with Dynamic Programming (DP). Because the number of possibilities grows exponentially with the number of companies, they need a heuristic because enumerating all possibilities simply takes too much time. They solve all planning problems of the individual households with a Dynamic Programming algorithm and then try to adapt this solution a bit to obtain a feasible overall solution. For the small instances they further apply an iterative search, but this takes too much computation time when the instances grow. The paper elaborates a bit on how to evaluate the quality of the heuristic. The authors state that given a certain generation technology, the electricity to heat ratio is known.

Caldon et al. [9] aggregate a number of small generators into a VPP. They consider thermal and electrical energy and optimize the operation for VPP, by means of a nonlinear minimization.

Bosman et al. [7] start with continuous models. They begin with modeling the scheduling problem for one Micro-CHP. They include a startup and shutdown time, and a minimal running time. The authors assume linear generation functions during startup and shutdown. Because the authors specify four different types of intervals, (namely starting up, running, shutting down, and off) the generation function is piecewise linear. They also assume a linear heat loss because the heat buffer is kept above a certain threshold that lies far above the environmental temperature. The authors also deal with the end level of the buffer and how to incorporate this into the model. This might be useful, otherwise the model will always make sure that the buffer is empty at the final period. The fleet scheduling is treated, in which the a group of houses cooperates in a so-called fleet. The focus is on balancing imbalance and this is not of direct interest for our research.
Afterwards the focus shifts to discrete models. The authors have to add a few more constraints because of the four interval types. This model can be formulated as an Integer Linear Programming (ILP). However, using Dynamic Programming is much faster for their instances.
When looking at multiple houses, the computation time grows exponentially. To decrease the computation time, they introduce the concept of restricted Discrete Single House Scheduling Problem with n houses (n-DSHSP restricted). It is proven that the n-DSHSP restricted problem is NP-complete in the strong sense. They reduce the solution space to a set of locally ‘good’ production patterns and try to find a selection of production patterns that adds up to the total desired production. This is equivalent to a partition problem.

The paper of Houwing et al. [13] considers a Multi-Level Decision Making (MDLM) approach, on the micro level. This means that every household performs their own optimization, but the degree to which they can optimize their own objectives is highly dependent on the decisions of other households. They apply the MLDM approach to the supplier and household levels. CO_2 is not considered in this paper. The authors assume imbalance and relatively high costs associated with that. Other assumptions are that the authority can influence the electricity price, that a household constantly knows its electricity consumption, and the supplier knows what each household produces and consumes. Also important is that it is assumed that a CHP has an efficiency of .85 for heat and .15 for electricity. Moreover, the authors assume that heat can be blown off into the environment.

The paper of Wille-Haussmann et al. [22] is about decentralized optimization of cogeneration in virtual power plants. With the usage of thermal storage systems, it is possible to decouple electric and heat production. In that case the production can be optimized individually and this would simplify the model greatly. This paper optimizes the CHP to sell electricity when the prices are high. They estimate the data with a multiple regression function. In our model this approach is not applicable as these commodities really cannot be decoupled. A CHP produces all three commodities at the same time. Therefore, decoupling the three commodities is an assumption we cannot make. In this paper, the CHP has a maximum of two starts per day, which is a very reasonable assumption, as a CHP depreciates a lot when turned off and on much. It is not profitable to do this more often than a few times per day.

Bosman et al. [8], consider a VPP on micro-level. They present a planning approach using column generation, and also look into making a desired pattern for a group based on global parameters. They start with a small set of patterns and add new locally feasible patterns which improve the existing relaxed solution. In this way, lower bounds of the solution are found and according to their research the solutions found are very close to the found lower bounds. This model is an MIP.
They are looking at the day-ahead market and do not consider startup, or shutdown times, in contrast to other papers by the same authors. Moreover, they use a relatively small group size.

We will finish with some remarks about the papers just discussed:
It is important to note that the big CHPs we are considering are different from micro-CHPs. Not only because of the size, which is obvious, but because of the environment in which they are used. The horticulture companies can better predict their demands in advance while the households have a big volatility in the error of the predicted demand. For a horticulture company it is known how many hours of heat the plants need each day. The amount of heat required is still dependent on the weather, which can be predicted relatively accurately. The predictions can be false and will still have some volatility, but much less than in one household. Even if we make only a one-hour prediction ahead of one household, we can be completely wrong because the inhabitants are in a traffic jam and the predicted televisions, computers, lights, etcetera are not turned on as our model predicted. Many papers treat the micro-CHP case. There are a lot of similarities, but in every paper we must be aware of important differences

that will occur while trying to implement the methods for a micro-CHP in this case.

2.2 Useful ideas from the literature for our research

In this section, we discuss what and how much of the above papers is implementable in this thesis. Molderink [19] has a lot of interesting ideas. Depending on the size of our model it is a possibility to implement step 1 and 2, which are local offline prediction followed by global offline planning. In this thesis, there is more direct interaction between the companies, which for instance also work with opening and closing times. In the paper of Molderink [19] the houses can turn their CHP on at any moment and sell this production back to the electricity net. In that paper more usage is made of a hierarchical structure to deliver, while in our thesis this is not realistic with for instance heat and CO_2. For the global planning the profiles are made and sent down the tree. In our setting this is only possible if we make a fictional point. That is because the power plant has other interests than the companies. It might however be possible to make a global online planning, like they do. In that case some heuristics are needed and dynamic programming is usable as well. There is company-to-company delivery in our case. For robustness it is possible to implement a local real-time control or to take robust values of the data. Or just to vary the data by Monte Carlo analysis and verify that the quality of the solution does not change too much. We have decided not to go for this approach because too many adaptations had too be made. The hierarchical structure to deliver is not realistic and we would have to make a fictional point. Instead of making a local prediction, we have chosen to take the demand given but with a certain uncertainty.

The paper of Kok et al. [16] is a good reference paper for background information. We are unsure about the 'must on' bid in the freezer example of Kok et al. [17]. In this example the freezer reaches maximum temperature and has to be turned on. Therefore, it must accept any price. We think this is a problem for the system, because this could become very expensive for the owner of the freezer.

There are some problems for adapting the Power Matcher in our setting:

- The buying and selling price are different.
- There have to be incentives for the residents to allow some discomfort.
- Needs local intelligence and id not transparent.
- 'Must on'- signal can be very costly. There is not too much ahead-thinking. Each unit makes a curve for itself based on marginal costs and market prices but there is not really a planning. The price can hence be flexible, whereas we will be looking at a fixed price for multiple periods.

All these problems convince us that the power matcher approach does not work for the current project.

The paper of Wright and Firth [23] is about calculation of domestic energy profiles. This might be useful later because we consider the horticulture companies first, which consume much more energy. If it appears that the model is scalable enough, perhaps a slight adaption can be made towards households. For now, it is interesting to see that they are using different time steps and the differences in results because of this. Therefore, it might be a good idea to test different time steps to find the optimal time step for our model.

Alanne and Saari [3] mainly focus on CHP for households, and the obstacles and solutions for implementing small-scale CHP. It is good to know which parties have to deal with the CHP. For a general background for these parties and the types of micro-CHP, this is a good reference paper.

The paper of El Bakari et al. [10] is useful for background about the VPP, and not directly for our research.

That they need a heuristic in Bosman et al. [6], because enumerating all possibilities simply takes too much time, is similar to our problem. Therefore, the Dynamic Programming part and search heuristic might be an interesting option to look at in our further research. In this paper, the authors create local solutions and try to adapt them a little bit to obtain a global feasible solution. For the small instances they further apply an iterative search, but this takes too much computation time when the size of the instances increases. The paper elaborates a bit on how to evaluate the quality of the heuristic. This might all be more or less implementable in the current research project, but we have chosen different heuristics. The possibility of transport between the companies makes our problem not fit for creating local solutions and trying to add them to make a good global feasible solution.

Caldon et al. [9] assume that a CHP primarily produces heat, and the electricity is something generated 'extra', which can be used or sold back to the net. Also a thermal storage is assumed, but no electric one. This is in line with our project. The objective function is to optimize the short-term variable production cost. That is not such a great idea for this thesis.

The paper of Bosman et al. [7] features a discrete model which is writable as an Integer Linear Programming problem (ILP). However, using Dynamic Programming is much faster in their instances. When looking at multiple houses the computation time grows exponentially. It is proven that the n-DSHSP restricted problem is NP-complete in the strong sense. To decrease the computation time, they introduce the concept of n-DSHSP restricted. The authors reduce the solution space to a set of locally 'good' production patterns. Then they try to find a selection of production patterns that adds up to the total desired production. This paper could be used to prove the complexity of our problem to be NP-hard. The production patterns idea cannot be used in this thesis as there is no total desired production level, rather each company has a demand to fulfill. Moreover, with three commodities which all have their own costs, revenues and demands finding a good global production plan will not be that easy at all.

Wille-Hausmann et al. [22] incorporate storage losses per time step and for the charging and discharging processes. Also incorporated are costs for running the CHP and for running a boiler. This is not in our model. To have a storage and a boiler seems a bit over the top. With the usage of thermal storage systems, it is possible to decouple electric and heat production. In that case the productions can be optimized individually and this would simplify the model greatly. This would just lead to two parallel optimization models. It would be interesting to decouple heat, electricity and CO_2 and solve these three simple models sequentially. For instance, we firstly solve optimize for heat. Given this solution, we optimize for electricity. This will give a new solution, which can be used to optimize one more time for CO_2. This method could provide a good starting solution. In the end we decided not to go for this approach as an LP-rounding heuristic seemed more interesting. The paper optimizes the CHP to sell electricity when the prices are high. In our model we are going to work with a fixed price for electricity. They estimate the data with a multiple regression function. In our model, this is not necessary. The demands of the companies are known long before and do not change much. We will however incorporate some robustness in our model.

The idea of using a column generation heuristic seems to be an interesting one, judging from the results of Bosman et al. [8]. Column generation might be applicable in this thesis, too. That startup and shutdown times are not incorporated makes life slightly more difficult but this should not be an insurmountable issue. It is an interesting heuristic and it might give us good upper bounds (we are maximizing) for the actual optimal solution. Because we found other interesting heuristics, in this thesis no column generation based heuristic has been developed. It is an interesting subject for further research.

Chapter 3

Model

In this chapter we give a detailed problem formulation in Section 3.1, followed by a list of assumptions in Section 3.2. This is followed by a description of all the variables, parameters, sets and indices used in Section 3.3. We formulate the mathematical model in Section 3.4. Then we discuss the complexity of the model and show it to be NP-hard in Section 3.5. The chapter finishes with some notes about the model in Section 3.6.

3.1 Problem formulation

We consider several companies from the horticulture industry, which have a CHP. The companies have certain demands of heat, electricity and CO_2, which need to be met and they are able to use their CHP for generation of their needed commodities. For running the CHP, gas is required, which can be imported in an unlimited amount from an independent power plant against a certain price. Shortages of any of these commodities can also be ordered from the same plant, but this is much more expensive. This power plant also uses gas to produce heat, electricity and CO_2. Companies can transport commodities to each other. For convenience, each company has a separate storage for each of the commodities. Losses occur in two cases, namely over time when storing the commodities and also when transporting commodities. These losses can be equal to zero. For instance (almost) no losses incur when transporting gas [16]. A bounded amount of leftovers can be sold back to the net, which yields some revenues. This amount is bounded as to prevent a very large flow from the companies to the power plant, because in reality the amount of commodities that can be sold back is limited as well. The objective now is to maximize the profit of the companies, which is the revenues minus the costs.

A schematic overview is presented in Figure 3.1. Here the building on top is the power plant and the other two buildings are companies. The three commodities are seen inside the buildings. 'C' stands for CO_2, 'E' for electricity and 'H' for heat. Companies can also import gas through the 'Gas'-node, use this gas to run their CHP and produce the commodities 'C', 'E' and 'H'. The arrows from and to the power plant denote the transport of extra commodities that are sold to, or bought from, the power plant. The arrows between companies

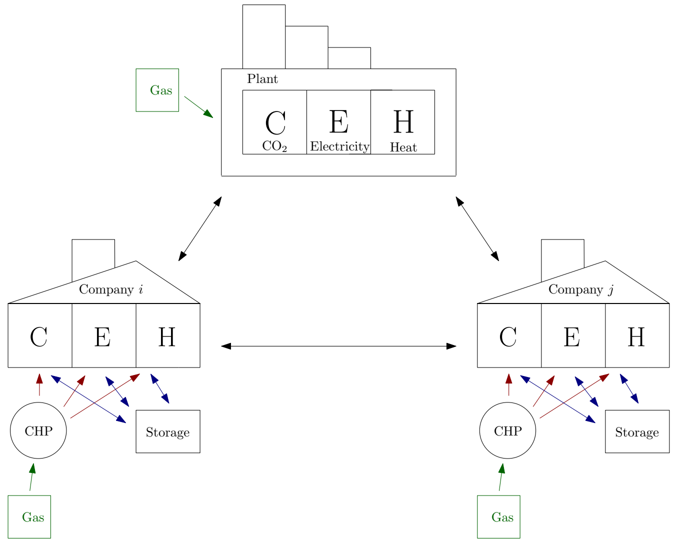

Figure 3.1: Schematic overview of the problem formulation

indicate the transport of commodities between these companies.

Every CHP can be either turned off, working on full capacity (100%) or working on medium capacity (70%). We will call this the intensity of the CHP, which can have the values 0, 1 and 0.7. Switching the intensity of the CHP between 0.7 and 1 can be done freely, but turning a CHP on (from 'off' to 'on') costs relatively much, and also a warm up time is needed. The same applies for shutting down a CHP.

Since a CHP can be only in three states, there will be integer variables in our linear model. Therefore, the problem will be modeled as an MIP (Mixed Integer Program). These are the only integer variables in the MIP. Later on, we see whether this assumption, that the CHP can only be in three states, not only heavily influences the computation time, but also the value of the objective function.

3.2 Assumptions

In this section we discuss the assumptions made for the model. We start with the assumptions about the *CHP*:

1. Every company has one CHP and the amount of output of a CHP can be different per company.

2. There are different types of CHP, but the ratio of output of the different commodities is known.

3. The value of the CHP depreciates a lot when turned on and off very often.

4. Every CHP only has three states, either ‘off’, or ‘on 70%’, or ‘on 100%’.

   The first assumption is made for simplicity. That every company has exactly one CHP is much easier to model, but this does not restrict the model. We have copied the second assumption from Bosman et al. [6]. For simplicity, we have assumed that there are three types of CHP, namely, small, medium and large. The third and fourth are done to resemble reality. We have taken the third assumption from Wille-Hausmann et al. [22]. We have assigned a cost to turning the CHP on or off in the model, as to make sure that a CHP is not turned on and off more than a few times each day. The fourth is made because in reality a CHP can also only be in these three states. This comes from LTO Noord Glaskracht [18].
   We follow up with the assumptions on *commodities*:

5. Every company can get a fixed amount of every commodity from the power plant at every moment in time, against a certain, relatively high, price. This price is higher than the revenues generated by selling commodities because otherwise the companies could make profit by buying commodities form the plant and directly selling it back, making the solution meaningless.

6. For companies it is possible to sell commodities for a fixed price, after signing a contract with the power plant. Because the amount of commodities sold back to the plant cannot be too big, we have included an upper bound on this in the model.

7. No CO_2 is lost during transport.

8. There is a constant heat loss factor, because the buffer is kept above a certain threshold that lies far above the environmental temperature.

9. The power lost in long-range transmission of electricity is 7%, when electricity is produced near the consumption area, transport losses are avoided.

   The fifth and sixth assumption are made because this also happens in reality. For companies it is possible to sign a contract with the power plant, to buy or sell a fixed amount of electricity for a certain price in a certain time range. We have assumed that this also holds for other commodities. Assumption 7 is also what happens in reality, according to OCAP [20]. The eighth assumption about the constant heat loss factor is also done in Bosman et al. [7]. Assumption 9 comes from Kok et al. [16].
   Next, we will treat the assumptions of *startup and shutdown*:

10. The production during startup and shutdown is linear. This means that the production of the CHP increases linearly from 0 to full capacity during startup, and vice versa during shutdown.

11. Startup and shutdown takes 15 minutes. We will also take this as our default time step.

12. During startup and shutdown, production of the CHP is halved.

The tenth assumption is taken from Bosman et al. [7]. The eleventh assumption is done for simplicity. It makes modeling easier if starting up or shutting down takes exactly one time period. In practice it takes about 15 minutes for a CHP to fully start up, according to Bosman et al. [6]. Modeling in time steps of 15 minutes is also computationally fine, as taking steps of one hour might make the model too shallow and superficial while taking steps of 5 minutes makes it too computationally intensive. Assumption 12 follows from the previous two. If the startup time is linear, from zero to full capacity, the average production in this period is halved. In practice, the startup time will not be exactly 15 minutes. Hence, we parameterized the factor by which the production is reduced. We are able to quickly adapt the model in case the time step would change. Moreover, we have tested whether it matters for the solution if this 0.5 would be a bit higher or lower. We discuss this issue in Chapter 4.
Finally, there are some *general* assumptions:

13. 'Full information availability assumption'. This means, in our case, that we constantly know how much of each commodity every company demands, as well as the prices of gas and the commodities at every time step.

14. The starting and stopping costs of the CHP are dependent on company and time.

15. The output factor is also dependent on time.

Assumption 13 comes from Houwing et al. (2006) [13]. The prices are known in advance. The demand is also known in advance, but with a certain degree of uncertainty, which is why we have also performed a Robust Optimization. Regarding the fourteenth assumption, it is certainly dependent on the type of CHP, but whether the starting and stopping costs depend on the time is debatable. We have decided to keep it in the model, in order to keep the model more generic. In the input data, we have constant costs over time. Regarding the fifteenth assumption, the same applies. We do not think that the output factor depends on time but to keep the model as generic as possible we have included this possibility.

For the same reason, we have not assumed anything about the storage of electricity. Neither have we assumed that heat can be blown off into the environment, as done in Houwing et al. (2006) [13]. It is however possible to incorporate this assumption in our model by introducing a penalty cost. In our case it is not necessary, as companies can put heat in their storage or transport it to other companies. In that case the heat can be sold in a later time step or by an other company, which is more profitable than incurring a penalty cost.

3.3 Indices, parameters and variables

3.3.1 Indices and sets

The set of all the companies is $\{1, \dots, I\}$, where I is the number of companies. The index i denotes a company, $i \in \{1, \dots, I\}$.

The set of all companies is a subset of the set of all nodes, $\{1, \dots, I\} \cup \{P\} \cup \{S\}$, where index P stands for the power plant and index S denotes storage. The index j denotes a node, $j \in \{1, \dots, I\} \cup \{P\} \cup \{S\}$. For modeling purposes there is only one storage node, where each company has its own storage per commodity.

The set of all speeds is $\{0, 0.7, 1\}$. The index s denotes the speed of the CHP, $s \in \{0, 0.7, 1\}$.

The set of all time periods is $\{0, 1, \dots, T+1\}$, where T is the number of time periods. The index t denotes the time period, $t \in \{0, 1, \dots, T+1\}$. At $t = 0$ and at $t = T+1$ the company is closed and the CHPs are turned off. We could say that $t = 0$ is the starting time step and $t = T+1$ the stopping time step.

Index c denotes the type of commodity, which can be either CO_2, *electricity*, or *heat*, i.e., $c \in \{CO_2, \mathit{electricity}, \mathit{heat}\}$. Gas is not treated as a commodity because companies do not have demands for gas. Gas is only used to produce CO_2, *electricity*, or *heat* and is available in an unlimited amount. One unit of CO_2 is a tonne, one unit of *Electricity* equals one megawatt (MW) and finally one unit of $Heat$ is one kilowatt hour (kWh).

Index g denotes gas.

3.3.2 Parameters

In this subsection, we will discuss the parameters used in our model. Parameters are in lowercase, while variables are capital letters.

$c_{P,i,c}$ are the costs per unit incurred by company i for importing commodity c from the power plant.

$c_{P,i,g}$ are the total costs incurred by company i for importing gas g from the power plant.

$ca_{i,t}$ are the total costs incurred by starting the CHP of company i at time t.

$cu_{i,t}$ are the total costs incurred by shutting down the CHP of company i at time t.

$d_{i,t,c}$ is the demand of company i at time t for commodity c.

$f_{i,t,c}$ is the amount of commodity c out of 1 unit of gas (output factor) for company i at time t.

$h_{i,t}$ indicates whether company i is open at time t.

$$h_{i,t} = \begin{cases} 1 & \text{if company } i \text{ is open at time } t \\ 0 & \text{otherwise} \end{cases}$$

$l_{j_1,j_2,c}$ is the loss factor for transporting commodity c from j_1 to j_2. $l_{i,i,c}$ denotes the loss over time of a stored commodity c.

m is a very large number.

$ms_{i,t,c}$ is the maximum level of commodity c that company i can store at time t.

$r_{i,c}$ is the revenue of company i for selling one unit of commodity c to the power plant.

σ is the factor that the production is reduced with during startup and shutdown.

τ_i is the type of the CHP of company i. The type is used for data generation and is explained in more detail in Section 4.1.

$w_{t,c}$ is the upper bound for the amount of commodity c sold to the power plant at time t.

3.3.3 Variables

$A_{i,t}$ is a binary variable that indicates whether the CHP of company i at time t is starting up or not. $A_{i,t} = 1$ if the CHP is starting up.

$$A_{i,t} = \begin{cases} 1 & \text{if } B_{i,t-1,s=0} = 0 \text{ and } B_{i,t,s=0} \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

$B_{i,t,s}$ is a binary variable which equals 1 if the CHP of company i is on speed s at time t.

$I_{i,t}$ is the intensity of CHP of company i, at time t.

$P_{i,t,c}$ is the production of CHP of company i, at time t of commodity c.

$S_{i,t,c}$ is the amount of commodity c in storage of company i at time t.

$T_{j,i,t,c}$ is the transport of commodity c from j to company i at time t.

$U_{i,t}$ is a binary variable that indicates whether the CHP of company i at time t is shutting down or not. $U_{i,t} = 1$ if the CHP is shutting down.

$$U_{i,t} = \begin{cases} 1 & \text{if } B_{i,t-1,s=0} \neq 0 \text{ and } B_{i,t,s=0} = 0 \\ 0 & \text{otherwise} \end{cases}$$

3.4 Mathematical formulation

Now that we have introduced all the parameters, variables, indices and sets we can finally formulate our model.

Objective function:

$$
\begin{aligned}
\max \sum_{i,t,c} & \left(l_{i,P,c} \cdot r_{i,c} \cdot T_{i,P,t,c} - c_{P,i,c} \cdot T_{P,i,t,c}\right) \\
& - \sum_{i,t} \left(c_{P,i,g} \cdot I_{i,t} + ca_{i,t} \cdot A_{i,t} + cu_{i,t} \cdot U_{i,t}\right).
\end{aligned}
$$

We maximize the total revenues minus the total costs. First the objective function sums the total revenues minus the commodity costs, after which the total gas, starting up and shutting down costs are subtracted.

The objective function is subject to some constraints. We start with the *equality constraints*:

$$
\begin{aligned}
d_{i,t,c} &= P_{i,t,c} + \sum_j l_{j,i,c} \cdot T_{j,i,t,c} - \sum_j T_{i,j,t,c} && \forall\, i,t,c && (3.1) \\
S_{i,t,c} &= l_{i,i,c} \cdot S_{i,t-1,c} + l_{i,S,c} \cdot T_{i,S,t,c} - T_{S,i,t,c} && \forall\, i,t,c && (3.2) \\
\sum_s B_{i,t,s} &= 1 && \forall\, i,t && (3.3) \\
I_{i,t} &= \sum_s \left\{ s \cdot B_{i,t,s} \right\} && \forall\, i,t\,. && (3.4)
\end{aligned}
$$

Equation (3.1) is the *demand* equation. The demand for a certain commodity is given, and equals the production plus all inflow minus all the outflow. At import, the loss is included, while at export, it is not. To illustrate this, suppose you order 100 units at the power plant and only 90 arrive, such that the inflow is 90. Next, suppose you sell 100 units, then 100 units flow out. Equation (3.2) is the *storage* equation. The amount of commodity g stored at time t at company i equals the amount in the previous period, multiplied by the loss over time, plus inflow minus outflow. Every company can put a bounded amount of each commodity in the storage. Because in equation (3.2) the amount of storage depends on both the company and commodity, a company cannot use commodities stored by other companies. This can also be visualized as a large storage building where every company has its own storage for each commodity. The third equation (3.3) is for the *speed* of the CHP, it can only have one speed at every time step. Therefore, the binary variables which account for the speed must add up to one. Equation (3.4) makes sure that the *intensity* of the CHP equals 1 (which is the production when the engine is on 100%) times the actual speed.

There are a few constraints concerning the *production*:

$$
\begin{aligned}
P_{i,t,c} &\leq f_{i,t,c} \cdot I_{i,t} && \forall\, i,t,c && (3.5) \\
P_{i,t,c} &\leq \sigma \cdot f_{i,t,c} \cdot I_{i,t} + m \cdot (1 - A_{i,t}) && \forall\, i,t,c && (3.6) \\
P_{i,t,c} &\leq \sigma \cdot f_{i,t,c} \cdot I_{i,t} + m \cdot (1 - U_{i,t}) && \forall\, i,t,c && (3.7) \\
P_{i,t,c} &\geq f_{i,t,c} \cdot I_{i,t} - m \cdot A_{i,t} - m \cdot U_{i,t} && \forall\, i,t,c\,. && (3.8)
\end{aligned}
$$

Now (3.5), (3.6), (3.7) and (3.8) bound the production. Every CHP has its own

output factor, $f_{i,t,c}$, which depends on time, on the engine and on the commodity. Normally, the production would equal this factor times the intensity, but because the production is smaller when the engine is starting up or shutting down, we have changed this to less than or equal. When the engine is indeed starting up or shutting down, its production is reduced, as mentioned in Section 3.2. This is the σ in (3.6) and (3.7). Constraint (3.8) becomes redundant. To get a linear form of these constraints, the large number m is used. When the engine is not starting up or shutting down, (3.6) and (3.7) are redundant. Moreover, in this case, (3.8) makes sure that the production is equal to the output factor times the intensity. We elaborate more on this subject in Section 3.6.
There are two constraints which bound the storage and outflow amount to a maximum level:

$$\sum_i \left\{ l_{i,P,c} \cdot T_{i,P,t,c} \right\} \leq w_{t,c} \qquad \forall\, t, c \tag{3.9}$$

$$S_{i,t,c} \leq m s_{i,t,c} \qquad \forall\, i, t, c\,. \tag{3.10}$$

Constraint (3.9) makes sure there is a maximum amount that can be supplied back to the power plant. Constraint (3.10) keeps the amount of commodity stored below a certain maximum level.
We now discuss the *starting up* and *shutting down* constraints:

$$A_{i,t} \leq 1 - B_{i,t,s} \qquad \forall\, i, t \; and \; s = 0 \tag{3.11}$$

$$A_{i,t} \leq B_{i,t-1,s} \qquad \forall\, i, t \; and \; s = 0 \tag{3.12}$$

$$B_{i,t-1,s} - B_{i,t,s} \leq A_{i,t} \qquad \forall\, i, t \; and \; s = 0 \tag{3.13}$$

$$U_{i,t} \leq 1 - B_{i,t-1,s} \qquad \forall\, i, t \; and \; s = 0 \tag{3.14}$$

$$U_{i,t} \leq B_{i,t,s} \qquad \forall\, i, t \; and \; s = 0 \tag{3.15}$$

$$B_{i,t,s} - B_{i,t-1,s} \leq U_{i,t} \qquad \forall\, i, t \; and \; s = 0\,. \tag{3.16}$$

Constraints (3.11), (3.12) and (3.13) make sure that the variable $A_{i,t}$ is either 0 or 1, and only equals 1 if the CHP is starting up. Starting up means that the engine was off in the previous period and on now. $B_{i,t,s}$ namely only equals 1 when the CHP is turned off. Now (3.14), (3.15) and (3.16) are similar, but for shutting down the engine. Because these equations already make the variables $A_{i,t}$ and $U_{i,t}$ binary, we do not have to include this as an additional constraint in the model. This is illustrated in Table 3.1.

Variable	Value			
$B_{i,t,s=0}$	0	0	1	1
$B_{i,t-1,s=0}$	0	1	0	1
$A_{i,t}$	0	1	0	0
$U_{i,t}$	0	0	1	0

Table 3.1: Values of $A_{i,t}$ and $U_{i,t}$ for all combinations of $B_{i,t,s=0}$ and $B_{i,t-1,s=0}$.

At the starting and closing time step, we have some constraints which make sure that certain variables are zero:

$$
\begin{aligned}
T_{j,i,t,c} &= 0 \quad \forall\, j,i,t,c \,|\, h_{i,t}=0 && (3.17)\\
B_{i,t,s} &= 0 \quad \forall\, i,t,s \,|\, h_{i,t}=0 && (3.18)\\
S_{i,t,c} &= 0 \quad \forall\, i,c\, and\, t=0 && (3.19)\\
I_{i,t} &= 0 \quad \forall\, i\, and\, t=0 && (3.20)\\
I_{i,t} &= 0 \quad \forall\, i\, and\, t=(T+1)\,. && (3.21)
\end{aligned}
$$

If company i is closed, (3.17) and (3.18) make sure that respectively the transport and the binary variables are equal to zero. This automatically implies that the intensity is equal to zero as well. Constraint (3.19) makes sure that the storage is empty at the starting time step. The final two constraints, (3.20) and (3.21) force the intensity equal to zero at the starting and stopping time step.

We finish with *binary and nonnegativity* constraints:

$$
\begin{aligned}
B_{i,t,s} &\in \{0,1\}\\
T_{i,j,t,c} &\geq 0 \quad \forall\, i,j,t,c\\
P_{i,t,c}, S_{i,t,c} &\geq 0 \quad \forall\, i,t,c\\
A_{i,t}, I_{i,t}, U_{i,t} &\geq 0 \quad \forall\, i,j\,.
\end{aligned}
$$

3.5 Complexity of the model

Note that the problem can also be formulated as a multi commodity network flow (MCF). This is depicted in Figure 3.2. One large rectangle symbolizes one time step. The companies, which are called i_1, i_2, i_3, etcetera, can import their commodities directly from their CHP (dashed red arrow) or indirectly from the plant (black arrow). Both arrows are connected to a dummy source. The companies can exchange commodities (denoted by the arrows in-between the companies), use them (every company node has a certain demand), sell them back to the plant (arrows to the plant node), or put them in storage for later use (blue arrows transporting commodities to the next time step). The sell-option is connected with a sink. The companies have a certain demand which is to be met. The only capacity restriction on these arcs is the amount of flow from the companies to the plant or storage.

These arrows and flows resemble Figure 3.1. The discrete problem of using the CHP which can only be in three positions is different in the MCF-case. The dashed, red arrow from the dummy source to a company incorporates the CHP, which is depicted in the lower right corner. The company can use either the node 'CHP 1', which corresponds to an intensity of 100 % of the CHP, with corresponding output (for instance 1000). Or the company can use the node 'CHP 0.7', which corresponds to an intensity of 70% of the CHP, with corresponding output (for instance 700). A capacity constraint of 1 on the arc from the 'dummy source'-node to the 'gas'-node should make sure that maximally one of these options is used. Requiring the flow through this arc to be integer makes

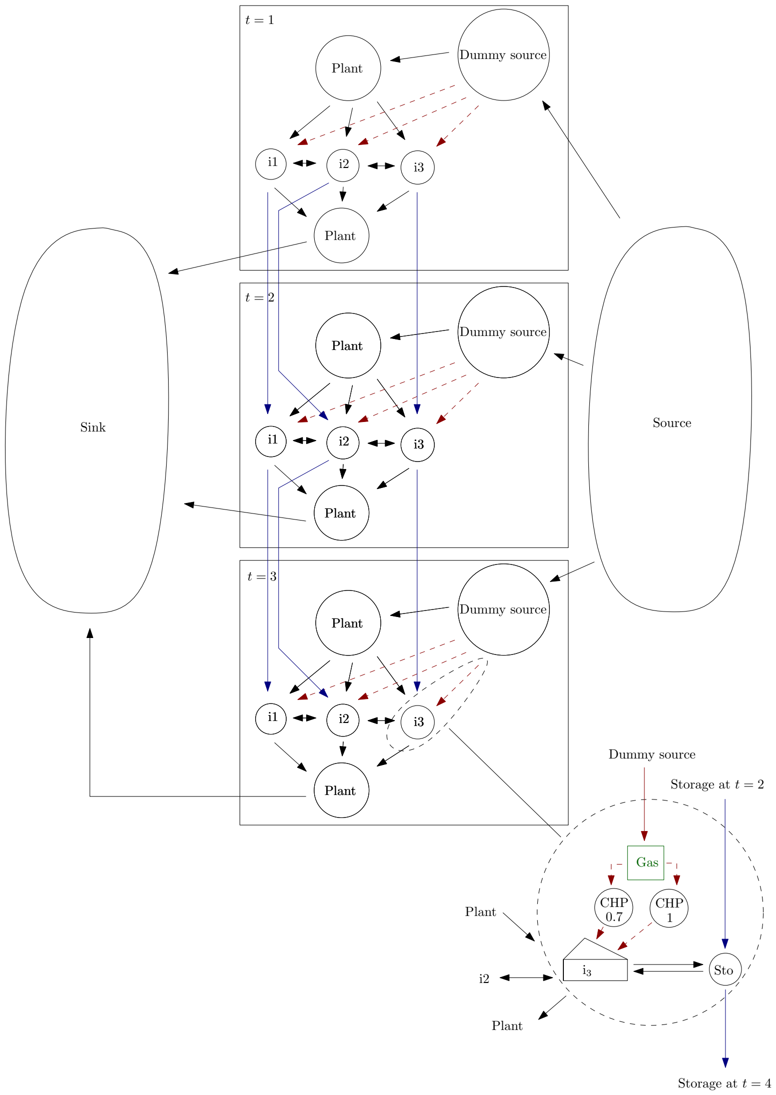

Figure 3.2: The problem written as a network flow model

sure that no fractions are taken. Finally, both flows through 'CHP 1' and 'CHP 0.7' can be zero. This means that the CHP is off, and the flow is indeed equal to zero in this case. The use of arc multipliers makes sure the amount of flow going from the CHP to he company is indeed this corresponding output mentioned before. Using arc multipliers makes the problem a *generalized network flow* [5], which is an extension of the normal flow model. If all arc multipliers would be equal to one in a generalized network flow model, a pure network flow model is obtained.

For the next time step, the companies have the same choices. They start with an amount of commodities which were already in their inventory. The arcs have their own multipliers, also called gains, which are greater than one in case of a revenue. The losses are also incorporated by adding arc multipliers. The same holds for the transport costs. Finally, the objective is to minimize the costs while fulfilling the demand restrictions.

Please note that not all arrows are drawn in the picture, because of clarity reasons. Every company has its own storage and can exchange goods with every other company.

It is known that a MCF with integer constraints is NP-complete [11]. As a MCF with integer constraints can be reduced to our problem, our problem is also NP-hard. This reduction is based on the following insight: a part of our model (within the dashed circle in Figure 3.2) is a MCF with integer constraints. The possibilities of storage, losses, and transporting to other companies are a generalization and do not reduce the complexity of the problem.

3.6 Notes

An alternative way to prove that our model is NP-hard is by reducing the n-DSHSP-restricted problem as in Bosman et al. (2010b) [7] to our problem. This would prove that our problem is also NP-hard in the strong sense. This reduction is beyond the scope of this thesis. We have shown that our model is NP-hard in an other way.

It seems that constraint (3.8) is not necessary. Suppose the production is only bounded from above, but could be lower. Thus, the proportion of CO_2, electricity and heat does not have to be the same for the same CHP. For instance, if one unit of gas produces one unit of CO_2, two units electricity and two units of heat, it is allowed to produce only one unit of electricity. However, it would be most profitable to produce as much as is maximally allowed, because all superfluous commodities can be sold back to the plant, put in storage for later use, or transported to other companies. Hence, this problem only occurs when both the maximal amount of selling the commodities back to the plant is reached and when the storage capacity has reached its limit for all companies. This happens only in very rare cases. However, that would be undesirable. It seems that this problem cannot be solved by fixing this proportion. If we do, the model can always lose the extra unit of commodity by transporting it from one node to another as often as needed. A transport loss will incur and the unit of commodity will be lost in the system anyway. To prevent this from happening, we have set the loss factor for transport from a company to another company to 1. Constraint (3.8) fixes the proportion when the CHP is running. Only when the CHP is starting up or shutting down, this proportion is not fixed. In this case, it is not necessary as the production is smaller and the very rare case of the storage being full and the limit on the outflow being reached is not possible.

The model is very generic. It is possible to remove the option of storing electricity and/or create a fictional pool for electricity, which has to be balanced at every moment. Also we can remove the option of transporting heat over great distances as this causes huge losses and is therefore not profitable.

It is possible to plan ahead for a number of days, execute the first day of the planning and plan again. Only the amount of storage at the starting time step has to be adjusted, and constraint (3.19) has to be deleted as this constraint forces the starting inventory to be zero. In this way a rolling horizon could be incorporated.

Chapter 4

Numerical results

This chapter is about the numerical results of the model. We start with describing the test problems and the data generation in Section 4.1. Then we analyze the solution in Section 4.2. In practice, companies already optimize for themselves. The metamodel, allowing the possibility of transporting commodities to other companies is more complex, but does it also give a better outcome? For varying data, our formulated model is compared with the corresponding non-cooperative model, where there are no connections in-between the companies and every company simply optimizes for itself. Section 4.3 discusses the computation time of our model and how this depends on the size of the model. Section 4.4 considers two heuristics, one based on the solver itself and an LP rounding heuristic. Section 4.5 is about changing the settings of the solver to try to improve the speed in which the model is solved. Section 4.6 discusses some test cases on which a simplified version of our model was tested. The chapter ends with two sections about the quality of the solution, namely in Section 4.7 a sensitivity analysis and in Section 4.8 a Monte Carlo analysis for robustness are performed.

The model was built in AIMMS (Advanced Integrated Multidimensional Modeling Software). AIMMS contains a lot of solvers, and will be used to solve our instances. MIP and LP problems will be solved by the well-known CPLEX solver, the current version being CPLEX 12.4. By default it uses dual simplex and Branch & Cut. We will see below whether other methods than these yield faster results. AIMMS solves the model to optimality. All experiments were performed on a Dual Core Processor running at 2.8Ghz with 8GB Ram (Intel Core i7-2640M. 2.8GHz, 8GB Ram, 64-bit operating system).

4.1 Data generation

Since the model has been built in AIMMS, the test data were also generated in this program. Because we do not have real data, we have had to generate it. We know a bit about the losses, which can be found in Section 3.2. A problem is that the transport losses depend on the distance between the companies, which are unknown. There is also not much known about the demands of the companies or the production factors. Therefore, the data is generated in such a way, that it can be easily adapted in case the real values appear to be different. We also do not know the values of most of the costs. Here, the same applies.

We have assumed that there are three types of CHP, namely small, medium or large. The type τ is randomly generated (small = 1 , medium = 2, large = 3). We have taken the costs of gas $c_{P,i,g}$ at a base level of 1000. This corresponds to running a small CHP at 100% level for 1 time step. We have taken $c_{P,i,g} = \tau(i) \cdot 1000$. Therefore, the costs of a medium CHP at 100% level are 2000. If the small CHP is at 70%, the costs will be 700. This is incorporated in the objective function by multiplying the gas costs with the intensity $I_{i,t}$. The costs of importing a commodity directly from the plant $c_{P,i,c}$ are a factor 1.5 higher, namely fixed at (150, 75, 15) respectively. This is necessary, as there would otherwise be a trivial solution of importing as many commodities as possible, and then selling these back with profits. The costs of turning a CHP on or off, $ca_{i,t}$ and $cu_{i,t}$, are fixed at 500. The revenues $r_{i,c}$ are fixed at 100 per unit of CO_2, 50 per unit of *electricity* and 10 per unit of *gas*.

The loss factor for the losses which occur from the plant to the companies and vice versa $l_{i,P,c}$ and $l_{P,i,c}$ is fixed at 0.9. The loss factor for the transport from company to storage and vice versa $l_{i,S,c}$ and $l_{S,i,c}$ is 0.95. The loss factor for the loss over time in storage $l_{i,i,c}$ is fixed at 0.95. The loss factor of transporting from company to another company $l_{i_1,i_2,c;i_1 \neq i_2}$ is set to 1. Finally, a loss factor of 0 from plant to storage and storage to plant $l_{P,S,c}$ and $l_{S,P,c}$ prohibit the direct transport from plant to storage and vice versa.

The demand $d_{i,t,c}$, and output factor $f_{i,t,c}$, are based on the ratio of CO_2, *electricity* and *heat* from a CHP. The demands are set at (5, 15, 80) for CO_2, *electricity* and *heat* respectively, for every time step when the company is open. The output factor needs to be constant per CHP and is dependent on the type. For a small CHP it is (3, 9, 48) for CO_2, *electricity* and *heat* respectively. A medium CHP has an output vector of (10, 30, 160) and finally a large CHP has an output vector of (25, 75, 400). The big m is set to 5000.

The maximal amount of storage $ms_{i,t,c}$ is set to (100, 300, 1600) for CO_2, *electricity* and *heat* respectively. The maximal amount of outflow back to the plant $w_{t,c}$ is set on a random integer between 100 and 150 for CO_2, between 300 and 450 for *electricity* and between 1600 and 2400 for *heat*, for every time step. This is the default case with standard data.

In the various experiments we change the amount of companies I and time steps T as will be indicated. The companies are open at every time step unless otherwise indicated, so $h_{i,t} = 1\,, \forall\, i, t\,;\; t \neq 0$ and $t \neq (T{+}1)$. It is easy to change the opening times $h_{i,t}$ in the model, which we have done on several occasions, but the computation time of the solution did not change.

4.2 Analysis of the solution

In this section we compare our model solutions with a non-cooperative version of the model where companies are not allowed to work together and only optimize for themselves. In the non-cooperative model we have added the constraint:

$$T_{j,i,t,c} = 0 \quad \forall\, j, i, t, c \,|\, j \in \{1, \ldots, I\}. \tag{4.1}$$

prohibiting the transport between companies. We analyze the difference between this non-cooperative model and our, cooperative, model which would indicate the improvement of our model on the existing situation.
Firstly, we consider the differences for datasets of different sizes using data as described in Section 4.1. The results are summarized in Table 4.1.

Standard data						
Size(I/T)	10/12	10/24	20/12	20/24	15/96	35/12
Cooperative	-83,671	120,764	-81,406	67,220	829,933	68,928
Non-cooperative	-116,215	71,495	-159,707	-69,694	582,489	-176,133
Difference	32,544	49,269	78,301	136,914	247,244	245,061
Gap(%)	28.00	40.80	96.19	203.68	29.81	255.53

Table 4.1: Comparison of the cooperative model with the non-cooperative model (OF value).

For small data cases, the difference in revenues is about 28% to 40%. When the number of companies or time steps increases, this percentage grows rapidly. If the number of companies increases further, the percentage grows as well. The only exception is the (15/96), which might be an outlier. Our cooperative model yields significantly higher revenues than the non-cooperative model. The number of time steps does not seem to have such a big influence on the percentual difference as the number of companies. Note that in this case the companies with a small CHP can not get any commodities from other companies and had to buy from the plant, which is expensive. This explains why the differences can be that big.

We now change the demand by making it much smaller, namely (2, 6, 32) for CO_2, *electricity* and *heat* respectively. This is more fair as every company can take care of itself now. In this case, the number of CHPs which have to be productive at every time step is lower in the cooperative case. We analyze whether the difference between the models changes. The results are in Table 4.2.

We observe that the values of the percentages smaller, but still very high. From Table 4.2 we can draw the same conclusions as from Table 4.1. The number of time steps does not seem to have a big influence on the percentual difference, but the number of companies does. Whether there is relatively much or little demand does not matter for these observations. With 20 companies or more, the difference is at least a very significant 17%. Our model improves on the existing situation and the size of the improvement is mainly dependent on

Adapted demand						
Size(I/T)	10/12	10/24	20/12	20/24	15/96	35/12
Normal	200,706	421,678	305,602	671,026	2,466,623	248,494
Naive	197,079	420,975	250,008	556,562	2,345,548	126,921
Difference	3,627	703	55,594	114,464	121,075	234,868
Gap(%)	1.81	0.17	18.19	17.06	4.91	48.92

Table 4.2: Comparison of the cooperative model with the non-cooperative model (OF value).

the number of companies (the more, the higher).

Note that the cooperative model raises a set of interesting questions from game theory perspective, such as: is every company better off? How to divide the profits made by cooperating? Answering these questions is beyond the scope of this thesis, but would be a very interesting area for further research.

4.3 Computation time

In this section we compare the computation time of the model on two instances. The first instance is a simpler version of the instance described in Section 4.1. There is no limit on the amount of outflow and maximal storage. We will refer to it as 'less realistic data'. The second instance is the one described in Section 4.1. We will refer to it as 'standard data'. For a varying number of time steps and companies it was measured how long CPLEX took to solve these problem instances.

T \ I	10	20	30	40
48	2.1	9.2	17.4	40.3
96	4.3	18.8	44.9	89.6
192	9.7	41.5	95.0	301.1
288	19.6	89.7	209.1	731.6

Table 4.3: Computation time of the MIP model with less realistic data (in seconds).

As can be seen from Table 4.3, these instances are very quickly solved by CPLEX. The solutions are not trivial, but the absence of a limit on the maximal amount of commodity sold back to the plant makes it relatively easy for the solver. The computation time grows exponentially with the number of companies and the number of time steps, e.g., the computation time gets more than 4 times larger than the number of companies doubles. This can be explained by the fact that companies can also transport to each other and this increases the complexity a lot. When the number of time steps doubles the computation time increases by slightly more than a factor 2, but when the model gets bigger this factor grows. The increment from 192 to 288 time steps is a factor 1.5, but the computation time more than doubles.

Three days ahead with time steps of 15 minutes equals 288 time steps. Hence, with 20 companies the model runs in about 1.5 minute, which is considerably good. In the large case with 40 companies and 288 time steps, the computation time of 731.6 seconds seems to be a lot. This is not a problem as the solution takes the data for three days in advance into account. Then the model has to be run at most a few times each week. When the input of the model gets even bigger it becomes arguable whether the computation time is still fine. For sure, input much bigger than 40 companies and 288 time steps cannot be solved to completion in reasonable time.

The data is made more realistic, by putting a lower limit on the maximal amount of commodity sold back to the power plant, and making the possible production much higher than the actual demand. This is the test instance described in Section 4.1. As a result, the computation time increases a lot. The results can be seen in Table 4.4.

T $\setminus$ I	10	20	30	40
12	<1	14.7	27.3	119.8
24	4.4	47.3	>900	>900
48	29.6	>900		
96	116.6			
192	696.3			
288	>900			

Table 4.4: Computation time of the MIP model with standard data (in seconds).

The data in Table 4.4 should give us a feeling about the speed of the model. We can conclude that the running times are not really acceptable for our model. This is always a bit subjective, but when CPLEX has not provided a solution within 900 seconds, it will be unclear when it will finish. For very small models, CPLEX can solve the problem to optimality, but when the model gets only a bit larger, optimizing can take a long time. For instance, 20 companies and 48 time steps, which is done in 9 seconds with less realistic data, now takes a very long time. For this reason, we will consider two heuristics in the next section. We have seen that the model runs considerably faster with less realistic data. However, we think that the solutions are too trivial and unrealistic. Therefore, we will only consider the standard data in what follows.

We have also tested the effect of removing the startup and shutdown constraints and costs, to see whether this would improve the speed of solving the model a lot. In fact, the reverse happens. The model takes longer to solve. With a size of $(I, T) = (10, 12)$ our model takes less than a second to solve, while the model with startup and shutdown removed was still not finished after three minutes. This might be because starting up and shutting down costs a lot, and it is not profitable to startup or shutdown more than a few times each day. This makes it easier for the solver as it limits the possibilities. However, when these constraints are removed, there are many more reasonable possibilities and in every time step it is to be determined whether the CHPs should run or not. Therefore, this removal shall not be considered in this thesis. Changing the factor with which the production decreases during startup or shutdown does not have a big effect on the results. When adapting this factor with 20%,only the value of the objective function changes, while the solution remains unchanged. Therefore, we shall keep it on 0.5.

4.4 Heuristics

4.4.1 Using CPLEX

The standard solution method of CPLEX can be used as a very good heuristic in itself, because it maintains the best upper bound found while solving. For the smaller cases it holds that the model quickly finds a good solution, and takes relatively much time to find the optimal solution. This is typical for an MIP and is depicted in Figure 4.1.

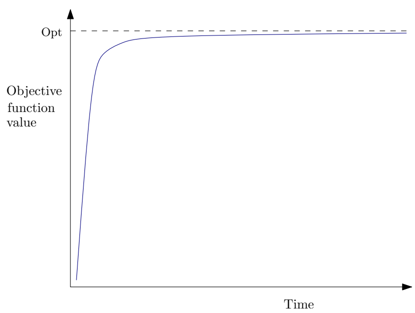

Figure 4.1: The objective function value of a MIP compared to the time when maximizing.

Already for $(I,T) = (20,48)$, which did not lead to a solution within 900 seconds in the previous section, the solver finds a solution that has an objective function (OF) value which is 99, 89% of the optimal OF value in 32 seconds. For the same values of I and T as in Table 4.3 we have looked how quickly CPLEX finds a nearly-optimal solution. This is a bit subjective, but a solution of at least 99% of the optimal value is considered as nearly-optimal. The nearly-optimal solution is found by aborting the solve manually. The results can be found in Table 4.5.

T \ I	10	20	30	40
48	opt / 29.6	0.11 / 32	0.31 / 180	0.14 / 220
96	opt / 117	0.07 / 73	0.13 / 680	0.61 / 300
192	opt / 960	0.08 / 32	0.37/ 200	0.92 / 620
288	0.01 /60	0.12 / 130	0.59 / 360	0.27 / 810

Table 4.5: Gap between LP bound and best found solution (in %) and computation time of the MIP model with standard data (in seconds).

How is the gap measured? CPLEX performs a LP-relaxation, which means relaxing the integer constraints to linear ones. The solution now obtained, called the LP-bound, is an upper bound for the solution of the MIP. The bound is compared with the best obtained ILP-solution up to that point, which is a lower bound for the optimal solution. The real optimal solution lies somewhere in-between. When the difference between the best ILP-solution and the best LP-bound is small, the difference between the best found ILP-solution and the optimal one is even smaller. Therefore, when this distance is small the model is very close to finding the optimal value. To be sure that the best found ILP solution is optimal however, the solver has to consider all possibilities. When the upper bound for a certain branch is lower than the solution already obtained, clearly this branch is not going to give a better solution and needs no further examination. This saves a lot of time, but still going through all branches is a lot of work. It explains why CPLEX is so fast in generating a nearly-optimal solution and slow in generating the optimal one.

In Table 4.5, firstly, the gap is shown. For example, a gap of 0.02% means that the gap between the best found solution and the best LP bound is 0.02%. Hence, the solution is at least 99,98% close to optimal. The time in seconds show how long CPLEX took to find this solution. As we can see, for the standard dataset we can still find a solution that is 99% slose or more in reasonable time. That is, for 40 companies and 288 time steps (3 days ahead in quarters). For this very large instance of $(I, T) = (40, 288)$ CPLEX finds a solution that is at least 98.73 % of the optimal value.

4.4.2 LP rounding heuristic

For a solution, two parts are required. One is the transport matrix, $T_{i,j,t,c}$. The other is the binary matrix, $B_{i,t,c}$. The latter is the integer part of the model. An idea for many heuristics will be to generate a solution for the B. This fixes the integer part of the model and the rest can be solved by means of an LP algorithm, which is known to be much faster than an ILP. Afterwards, the optimal, non-integer values found for the linear case can be used to obtain integer values. The model can now be solved again, with the integer values fixed, to optimize for the T. This by no means gives the optimal solution, and it is not even a guarantee for being close. However, it could head to a good solution value and thus a good lower bound.

The most natural and straightforward way to implement this method is to relax the integer constraints. In our case this would be removing the binary constraint on $B_{i,t,s}$ and replacing it by the requirement that $B_{i,t,s}$ has to be in-between zero and one. Unfortunately, this straightforward relaxation of the integer constraints does not work. If $B_{i,t,s}$ is between zero and one, so will $A_{i,t}$ and $U_{i,t}$ be. This leads to trouble in constraints (3.6), (3.7) and (3.8), because the big scalar m is used there, which is based on $A_{i,t}$ and $U_{i,t}$ being strictly binary. The constraints result in the production being smaller or equal than a scalar between zero and one times m, which is completely meaningless.

Because the problem just described cannot be solved easily, at least not by keeping things linear, a more rigorous relaxation is required. A linear problem cannot optimize over the $A_{i,t}$ and $U_{i,t}$ variables, which is why these will be left out. Consequently, constraints (3.6),(3.7) and (3.8) are deleted as well as constraints (3.11) up to (3.16). Instead we have made (3.5) an equality. The resulting linear model is much less complex than our model, which was described in Chapter 3. How fast exactly is this LP? We will look at the same number of companies and time steps as in Table 4.4, such that we can make a good comparison.

T ╲ I	10	20	30	40
48	0.26 (29.6)	1.04 (>900)	2.29 (>900)	3.81 (>900)
96	0.58 (116.6)	1.95 (>900)	5.06 (>900)	8.67 (>900)
192	1.40 (696.3)	6.68 (>900)	10.25 (>900)	31.33 (>900)
288	2.28 (>900)	10.84 (>900)	27.64 (>900)	93.38 (>900)

Table 4.6: Computation time of the LP model compared to the computation time of the MIP model (within brackets). The data are in seconds.

In Table 4.6 we see that the computation time of the LP model is much lower than of the ILP model. Even a very large model with 40 companies and 288 time steps can be solved in about 1.5 minute. For the same (standard) data, the ILP model ran for more than one hour without finding the optimal solution. This instance has 1.6 million variables and 3.5 million nonzeros, so it is really big. To illustrate this, let us do a small calculation. There are 40 companies, 288 time steps and 3 commodities. The companies can transport to each other company. Hence, we have

$$40 \cdot 40 \cdot 288 \cdot 3 \approx 1.400.000$$

variables only for transport. An obvious drawback is that this approach is further away from our original model and is a lot simpler, so we cannot call it an LP-relaxation any more. However, this method can be used to build an interesting heuristic.

The challenge is to use this linear solution to construct a good solution for the original model. In the solutions generated by the linear program, the speeds of many CHPs can easily be translated into an integer solution, because some are on 100% for all time periods, some are off for all time periods. The problem that the production is halved during startup and shutdown, as well as the costs of turning on and off can be solved later, when using the obtained LP solution to construct a solution for the original model. The former is not a big problem anyway because missing commodities can be bought from the plant if necessary. The turning off and on costs are a small fraction of the objective function. Only when a CHP would be turned on and off very often these costs would pile up very high. Therefore, the difficult part are the remaining CHPs, which have such a type of solution (of turning off and on very often) in the LP.

There are many ways to solve this problem. One is to round them as good as possible and solve the original problem with all CHPs fixed. Another option is to fix the CHPs with an (almost) integer solution, and solve the corresponding MIP for the remaining few CHPs. This is a little more accurate, but requires more calculation power. Whether this is possible depends on the size of the model and the number of remaining CHPs. Let us have a look at the amount of CHPs which can be fixed easily.

Size (I/T)	20/24	20/48	30/96	30/192	40/192	40/288
Minimum number of CHPs	9	11	12	15	23	22
Average number of CHPs	13.4	13.8	17.7	18.7	26.9	25.4
Maximum number of CHPs	20	20	22	23	31	28
The above numbers	45.0	55.0	40.0	50.0	57.5	55.0
in percentages	67.0	69.0	59.0	62.3	67.3	63.5
	100.0	100.0	73.3	76.7	77.5	70.0

Table 4.7: Number of CHPs of the LP model with standard data with an integer solution (in absolute numbers and percentages). These numbers are based on averages over 10 runs.

The amount of CHPs which have an integer solution is dependent on the size of the model and is depicted in Table 4.7. We notice that this number is more volatile for smaller models, and even for the largest models of the size of 40 companies and 288 time steps, more than half of the CHPs have an integer solution. We have to note that in the smallest two cases, there were a lot of CHPs with an almost integer solution, just being 'on' for one time step with a very small intensity, or vice versa. The average is about 65%, which is a lot. Fixing these CHPs reduces the difficulty of the model considerably. It is necessary to use this LP solution to construct a solution for the original, ILP, model. Moreover, we notice that the number of time steps does not really influence the number of CHPs with an integer solution, but the number of companies does. This is logical in absolute terms (more companies means more CHPs and also more CHPs with an integer solution), but the percentage of the CHPs with an integer solution remains more or less the same.

For small models, of a size smaller or equal than 30 companies, the option of just fixing the CHPs with an integer solution and then optimizing the remaining original model seems very appropriate and is expected to work well. Alternatively we can find a solution with a very good gap. If the instance is bigger, of a size of 40 companies or more, a smart heuristic is required. An interesting option would be the following: next to fixing the CHPs with an integer solution, round one CHP which is closest to having an integer solution, fix its solution, and run the LP relaxation again. This results in a new solution. In this new solution, again fix one CHP with a solution which is closest to being integer, and iterate until there remains a number of CHPs which is so small that the the model can be solved by the original ILP.

4.5 CPLEX solver settings

In this section we consider other methods to solve the MIP and LP. For LP solving, CPLEX uses Dual Simplex by default. We also consider the alternatives and see how well these perform. Afterwards we look at the MIP solver, where the choice is between Branch & Cut and Dynamic Search. Next to this, we can put emphasis on feasibility over optimality and vice versa, by default this option is on 'balanced'. We investigate whether this option is the best. Firstly, we consider the methods for solving LPs. This seems strange as we are solving an MIP, but the Branch & Cut solves LPs to obtain cuts. The Dynamic search option of CPLEX also makes good use of Branch & Cut algorithms. In this section, we will make use of a starting solution. if the model has already been solved, it can use the found solution as a bound. Solving it again is now faster. To be able to compare different methods, we have to use the same data sets. Then solving the model with this found starting solution is necessary.

4.5.1 LP method

Alternatives in AIMMS for Dual Simplex are the Primal Simplex, Network Primal, Network Dual, Barrier, Barrier + Primal crossover, Barrier + Dual crossover and Sifting. Primal and Dual Simplex are well-known. The CPLEX solver also offers a network optimizer, which is aimed at solving problems which (partly) have a network structure. CPLEX will apply the network optimizer to that (part of) the problem and uses the partial solution it finds as an advanced starting solution to optimize the remaining problem. The barrier method is an interior point-method. Finally, sifting is based on column generation. Sifting solves a sequence of LP subproblems, where the results from one subproblem are used to select columns, from the original model, for inclusion in the next subproblem. This process eventually converges to an optimal solution for the original model, according to the AIMMS help manual [1]. Sifting is especially applicable to models with many more columns than rows. Different LP methods are tested on varying instances.

Method ╲ Size (I/T)	10/12	20/12	10/24	10/48	25/12
Primal Simplex	0.90	3.34	2.95	12.53	33.23
Dual Simplex	0.91	3.34	2.81	11.89	33.07
Network Primal	0.90	3.48	2.89	15.40	33.35
Network Dual	0.90	3.39	2.93	16.44	33.43
Barrier	0.94	3.45	2.93	12.67	33.48
Barrier Primal	0.89	3.35	2.95	13.18	33.41
Barrier Dual	0.91	3.45	2.95	12.25	33.48
Sifting	0.92	3.46	2.98	13.24	34.38

Table 4.8: Computation time for solving the model with different LP methods (total time in seconds).

In Table 4.8 we can see that Primal Simplex, Dual Simplex, and Barrier Dual seem to perform best. Except for the first instance, where the difference is very small, Dual Simplex is the fastest. Looking at the aggregated computation time in Table 4.9, we see that the Dual Simplex method indeed has the lowest aggregated time. Second is the Primal Simplex, which has a slightly lower total time than the Barrier Dual method. These two methods seem reasonable as well, the other methods are too slow for this model. We will keep using the Dual Simplex method as this is the fastest.

Method	Total time
Primal Simplex	52.95
Dual Simplex	52.02
Network Primal	56.02
Network Dual	57.09
Barrier	53.47
Barrier Primal	53.78
Barrier Dual	53.04
Sifting	54.98

Table 4.9: Aggregated computation time for solving the LP model with different methods (total time/solving time (seconds)).

4.5.2 MIP search strategy

We have also looked at other ways of varying the solving process. For the MIP method, the choice is between 'Dynamic search', 'Branch&Cut' and 'Automatic'. The latter is also the default option. Dynamic search is a "new and innovative approach for MIP, which is innovative in its integration and sequencing of the usual branching, nodes and cuts in branch-and-cut algorithms", according to [1]. The computation times for different datasets can be found in Table 4.10.

Our model is solved best by Dynamic search, which is also the choice of the option 'Automatic'. Therefore, these results are very similar.

Search strategy ∖ Size (I/T)	10/12	20/12	10/24	10/48	25/12
Automatic	0.90	3.48	2.90	11.95	33.56
Branch & Cut	0.95	>90	3.39	27.52	34.04
Dynamic Search	0.90	3.48	2.89	11.93	33.56

Table 4.10: Total computation time for solving the MIP model with different search strategies (seconds).

As can be seen from Table 4.10, the 'Automatic' and the 'Dynamic Search'-options, which are the same in this case, perform best. When the size of the instance increases, the Branch & Cut option seems to get worse. Since the 'Dynamic search'-option is at most 0.02 seconds faster, we can leave the option on 'Automatic'.

4.5.3 MIP emphasis

As another option, we have looked at the emphasis of the MIP model. Should this emphasis be on feasibility, optimality, moving best bound, hidden feasibility, or simply balanced? First, let us explain the options of moving best bound and hidden feasibility. According to the AIMMS help index [1], emphasize moving best bound puts great emphasis on finding feasible solutions by moving the best bound. In case of hidden feasibility, the MIP optimizer tries to find high quality feasible solutions that are otherwise very difficult to find. This option is best used when you want a good feasible solution and not necessary the provably optimal one, and when the 'Emphasize feasibility'-option does not provide solutions of acceptable quality. The results can be found in Table 4.11.

MIP emphasis \ Size (I/T)	10/12	20/12	10/24	10/48	25/12
Balance feasibility and optimality	0.91	3.32	2.86	11.87	33.32
Emphasize feasibility over optimality	0.92	2.06	1.20	9.67	26.02
Emphasize optimality over feasibility	1.00	4.60	2.87	26.33	34.99
Emphasize moving best bound	0.72	68.92	4.60	26.93	94.71
Emphasize hidden feasibility	1.01	2.51	2.39	15.52	104.75

Table 4.11: Total computation time for solving the MIP model with the emphasis on different aspects (seconds).

In this case the results are very volatile. We see that the default option, 'Balance feasibility and optimality' performs well and does not have any outliers. The second option, 'Emphasize feasibility over optimality', was even significantly faster. We have tested these two methods for a larger set of $(I, T) = (25, 24)$, but here the former option was much faster with 118.34 seconds against 530.79. Therefore, the second option is considered more risky. It seems that it is better for smaller models, but much slower when the model gets large. The third option, 'Emphasize optimality feasibility', also performs reasonably well, but does have some outliers. The second option would be an interesting way to speed up solving the small-size models. However, these run fast enough, it would be more interesting to speed up solving the models of a larger size. Because we did not want to switch solvers when the model becomes large, we have decided to keep the first option, as it is the most all round.

For all three methods or solving or looking at the MIP or LP, we have decided to stick with the default option. In the third case of MIP emphasis, we have found the only interesting alternative to the default option, as it performs better for small-size models. When the size of our model increases, it becomes much slower.

4.6 Test cases

We have tested our model for five test cases, generated by Frank Phillipson [12] for the project that was already performed in this research area. The cases consist of a few companies, which have a certain demand for each time period, and a few suppliers, with a CHP with a given capacity and output factor. The suppliers can start their CHP at different moments in time and it has to run for exactly 8 periods in a row. Note that this is different from the assumptions done in our model. The number of time steps is 32. The number of companies increases with the size of the test case. Because these requirements are a bit different than the model we have used this far, our model had to be adapted for these cases. Storage, startup, shutdown, sellback and CO_2 are not taken into account. Some other constraints, which are mentioned above, are:

1. The objective function is linear.

2. Demand need not be met, but if demand is not met a penalty cost for each unit short or overproduced incurs. This is solved by setting the revenues of sellback to -1 and the costs of buying extra commodities are set to 1.

In our model, the objective function value is denotes a revenue, but here it denotes a penalty. Hence, the lower the objective function value, the better the solution. The extra requirements are met by adding constraints. The results of the instances can be found in Table 4.12. The test cases can be found in the appendix.

Test case	1	2	3	4	5
Number of demanding companies	6	8	9	11	12
Number of suppliers	6	7	9	10	12
Number of companies	12	15	18	21	24
OF value	34,193	133,843	32,541	67,080	42,938
Computation time (sec)	0.26	0.30	0.55	0.56	0.66

Table 4.12: Size, OF value and computation time of five test cases.

The computation time is very low, the largest instance was solved less than a second. In test case 2 the OF value is much larger than in other cases. This is logical, because there was a large demand in the first periods that could not be met as none of the CHPs was allowed to start. This resulted in a high objective function value. Concluding, our model solved these test cases to optimality very fast.

4.7 Sensitivity analysis

In the previous sections we have considered the time needed to solve the model to optimal completion, but what happens when the demand changes a little bit? Will the optimal solution be very similar or completely different? It is important to consider this as real-life demand cannot be 100% accurately predicted. In this section we have a look at how sensitive the outcome is with respect to small and large changes in demand.

In the original cases, we used a fixed demand of (5, 15 and 80) units of CO_2, electricity and heat, respectively. We are now going to perform an increase and decrease of (5, 10, 20 and 40) % and see whether the optimal solution is going to change a lot. This has been performed on different initial data sets.

The result of one of the examples is given in Table 4.13. The size is $(I, T) = (10, 12)$. It is shown whether the CHP is switched on or not for all time steps, and when it is on 70% power. This is because the model either puts a CHP on for the entire time period, or off. With only 12 time steps it is too expensive to turn the CHP off and on again. 'On' can be at 70% or 100%, and this is indicated in the second row every time. As can be seen from Table 4.13, the model is quite insensitive against changes in demand. If the demand changes downwards, nothing changes in the solution studied. When the demand changes upwards, it puts the CHP more often on full power. This happens only when

Company number ╲ demand change	−40%	−20%	−10%	−5%	0	+5%	+10%	+20%	+40%
1	On 1,12	On 1,12	On 1,12	On 1,12	On 1,12	On 1,12	On 1,12	On 1,12	On
2	On	On	On	On	On	On	On	On	On
3	On	On	On	On	On	On	On	On	On
4	On	On	On	On	On	On	On	On	On
5	Off	Off	Off	Off	Off	Off	Off	Off	Off
6	On 1,12	On 1,12	On 1,12	On 1,12	On 1,12	On 1,12	On 1,12	On 1,12	On
7	On	On	On	On	On	On	On	On	On
8	On 1,12	On 1,12	On 1,12	On 1,12	On 1,12	On 1,12	On 1,12	On 1,12	On
9	On 1,12	On 1,12	On 1,12	On 1,12	On 1,12	On 1,12	On 1,12	On 1,12	On
10	On	On	On	On	On	On	On	On	On

Table 4.13: Sensitivity analysis of the model showing the different states of the CHPs for changes in demand with respect to the nominal value (CHP status On/off; which CHPs are on 70%).

the demand increases with a huge 40%. For the model it is cheaper to change a CHP from 70% power to full power or vice versa, than to put a new one on. The penalty incurred by turning a CHP on is apparently greater than the extra transport costs incurred by keeping the CHP off at company i. Concluding, the nominal solution is insensitive with respect to changes in demand.

This sensitivity analysis is not to be confused with robustness. Both consider uncertainty of the input parameters, but in a different way. A sensitivity analysis checks whether changes in demand, say, one day ahead, result in completely different solutions. We have seen that the initial solution does not change drastically. Robustness studies how vulnerable certain solutions are to changes in for example demand. How good is this solution now? We will treat robustness in the next section.

4.8 Monte Carlo analysis

This section is about the robustness of the model. We perform a Monte Carlo analysis to see how robust the model is with respect to different uncertain parameters. The Monte Carlo analysis method works as follows: after solving the model, we obtain optimal values for the objective function and for the variables. Next, we vary over a selected set of parameters which might be uncertain, while keeping the variables (the 'solution') the same. The idea is to see how much the objective function will change. We have taken the possible values that the parameters can have as an interval around the nominal value. The borders of this interval are at +20% and -20%. The parameters we have studied are $c_{P,i,c}$, $c_{P,i,g}$, $ca_{i,t}$, $cu_{i,t}$, $d_{i,t,c}$ and $r_{i,c}$, since costs and revenues might change over time and the demand might for example be dependent on the weather. As a performance measure, we will consider the change in objective function value caused by a change in the parameter values. All parameters, except for the demand, can be found in the objective function.

The objective function is

$$\max \sum_{i,t,c} \left(l_{i,P,c} \cdot r_{i,c} \cdot T_{i,P,t,c} - c_{P,i,c} \cdot T_{P,i,t,c} \right) \\ - \sum_{i,t} \left(c_{P,i,g} \cdot I_{i,t} + ca_{i,t} \cdot A_{i,t} + cu_{i,t} \cdot U_{i,t} \right).$$

Because the solution remains the same, the variables will not change. Therefore, in the objective function, T,I,A and U will remain fixed. The change in the objective value by a change in these parameters can easily be calculated. For instance, if $ca_{i,t}$ changes, the change in objective function value will be $\sum_{i,t} \Delta ca_{i,t} \cdot A_{i,t}$. All the parameters except for demand are analyzed in this way. We will now consider the demand. Recall the demand equation 3.1:

$$d_{i,t,c} = P_{i,t,c} + \sum_j l_{j,i,c} \cdot T_{j,i,t,c} - \sum_j T_{i,j,t,c} \forall\, i,t,c$$

If the demand changes, (3.1) must still hold. This has to be by a change in the variable T. Moreover, demand affects the objective function through a change in transportation. Therefore, we will now consider a part of the T not to be fixed anymore, in order to hold the equation. This is done by altering the amount of commodities sold to or bought from the plant. It is not allowed to optimize over T given the new parameters, and the transportation between companies and from and to the storage is still fixed. How do we deal with the fact that we now have products of a parameter and a variable with changing values in the objective function? The revenues change and so does the amount of commodities transported. This gives us a change in revenues, $\Delta r_{i,t}$, and a change in the amount of commodities transported, $\Delta T_{i,j,t,c}$. The change of the objective function value is the difference between the 'old' (optimized) and the 'new' (after changing the parameters) objective function value:

$$\begin{aligned}
&\Delta \, OF\, value \,=\, OF\, value_{new} \,-\, OF\, value_{old} \\
&= (r+\Delta r)\,(T+\Delta T) - r\cdot T \\
&= r\cdot T + \Delta r\cdot T + r\cdot \Delta T + \Delta r\cdot \Delta T - r\cdot T \\
&= \underbrace{\Delta r\cdot T}_{\text{A}} + \underbrace{\Delta T(r+\Delta r)}_{\text{B}}
\end{aligned}$$

Part 'A' is already calculated above, at the revenues part. For the change in objective function value caused by the change in demand, we will use part 'B'. Note that $r + \Delta r$ is equivalent to the 'new' r. This also holds for the other parameters which are multiplied by the matrix T in the objective function. Therefore, we will use the 'new' parameters for the costs and the revenues when considering demand.

The demand can be either higher or lower than expected. Let us consider the first case, where the demand is higher than expected. In this case, extra commodities are needed. The company can decide not to sell superfluous commodities back to the plant, and will not make the corresponding revenues. This corresponds to lowering the negative term $T_{i,P,t,c}$ in Equation 3.1.

Example 4.1. Suppose at time t the demand for commodity c is higher than expected, $\Delta d_c = 5$. In the optimal solution company i sells 6 units of commodity c to the plant at time t, $T_{i,P,t,c} = 6$. Now, the change in the objective function will be as follows: $\Delta OF = -5 \cdot r_{new}$.

Because the revenues are lower than the costs of buying extra commodities by assumption, this is the most profitable option. When the amount of commodities sold back is zero, or smaller than the extra demand, the company will first decide not to sell their superfluous commodities and solve the remaining demand by buying extra commodities from the plant. This corresponds to lowering the negative term $T_{i,P,t,c}$ to zero, and then increasing the positive term $T_{P,i,t,c}$ in Equation 3.1.

Example 4.2. Suppose at time t, $\Delta d_c = 5$. In the optimal solution company i sells 4 units of commodity c to the plant at time t, $T_{i,P,t,c} = 4$. Now, the change in the objective function will be as follows: $\Delta OF = -4 \cdot r_{new} - 1 \cdot c_{new}$.

In the second case, the demand is lower than expected and commodities are superfluous. The company will first decide not to buy extra commodities from the plant. When the amount of commodities bought is zero, or smaller than the remaining amount of superfluous commodities, the remaining commodities will be sold back to the plant. There is a given maximum to this. When the maximum is reached the still superfluous commodities are lost. This corresponds to lowering the positive term $T_{P,i,t,c}$ in Equation 3.1.

Example 4.3. Suppose at time t, $\Delta d_c = -5$. In the optimal solution company i buys 6 units of commodity c from the plant at time t, $T_{i,P,t,c} = 6$. Now, the change in the objective function will be as follows: $\Delta OF = 5 \cdot c_{new}$.

Because the costs of buying extra commodities are higher than the revenues of selling commodities by assumption, this is the most profitable option. When the amount of commodities bought is zero, or smaller than the decrease of demand, the company will first decide not to buy the unnecessary commodities and sell the superfluous commodities back to the plant. This corresponds to lowering the positive term $T_{P,i,t,c}$ to zero, and then increasing the negative term $T_{i,P,t,c}$ in Equation 3.1.

Example 4.4. Suppose at time t, $\Delta d_c = -5$. In the optimal solution company i buys 4 units of commodity c from the plant at time t, $T_{i,P,t,c} = 4$. Now, the change in the objective function will be as follows: $\Delta OF = 4 \cdot c_{new} + 1 \cdot r_{new}$.

There is a maximal amount of commodities that can be sold back to the plant at every moment in time. When this amount is reached, we have assumed that still superfluous commodities will be lost. This corresponds to lowering the positive term $T_{P,i,t,c}$ to zero, and then increasing the negative term $T_{i,P,t,c}$ until $T_{i,P,t,c} = wt, c$ in Equation 3.1.

Example 4.5. Suppose at time t, $\Delta d_c = -5$. In the optimal solution company i buys 3 units of commodity c from the plant at time t, $T_{i,P,t,c} = 2$ and the amount of commodities sold back is bounded $w_{t,c} = 1$. Now, the change in the objective function will be as follows: $\Delta OF = 3 \cdot c_{new} + 1 \cdot r_{new}$.

To perform the Monte Carlo analysis, we have to perform different simulations. Therefore, we have added a new set and a new parameter in AIMMS.

The set of all the simulations is $\{1, \ldots, Z\}$, where Z denotes the number of simulations. The index z denotes a simulation, $i \in \{1, \ldots, Z\}$.

Furthermore, we gave all parameters considered an extra dimension and index, z, which denotes the simulation index. Finally, we have adapted the objective function such that it is a vector over all simulations.

For every simulation, we have generated a random value for these parameters within specified intervals. The generation of these numbers is as follows. For the probability distribution of the interval we have chosen three options: this distribution can be uniform, triangular and normal. We consider the change of the objective function with respect to the nominal value. In all three cases the objective function value is 163,682.

Let us start with the uniform case. For each parameter, the probability is distributed uniformly between -20% and +20% of its nominal value. In Figure

	Statistic	Distribution type Uniform	Triangular	Normal
Quartiles	Minimum	-87,839	-59,904	-91,465
	Quartile 1	-24,004	-15,103	-25,253
	Median	-1,161	1,315	-1,177
	Quartile 3	24,358	15,114	23,873
	Maximum	94,097	74,451	93,985
Shape statistics	Sample size	1000	1000	1000
	Average	-113	441	-468
	Stdev	34,068	22,399	33,569
	Skewness	0.022	-0.0068	0.0419
	Kurtosis	-0.4694	-0.2420	-0.3272
95%-CI	Lower CI value	-2,224	-947	-2,549
	Upper CI value	1,999	1,830	1,612

Table 4.14: Statistics about the shape and location of the realizations of the Monte Carlo method for three different distributions, considering the change in OF value.

4.2, the outcomes have a distribution which seems to be triangular with the probability mass centered in the middle. At the edges the probability is lower than in the middle, which is similar to a normal distribution case, for instance as in Figure 4.4. This is confirmed in Table 4.14, when looking at the quartiles. There is a small negative skewness and the mean is not significantly different form zero. In the worst case, the objective function value decreases with 88,000. This is about 54% of the objective function value, which is a significant amount.

Next, we consider the triangular probability distribution. In this case, the probability is distributed triangularly with the peak at the nominal value. In Figure 4.3 the outcomes are also triangular. This can also be seen from the quartiles in Table 4.14. The interquartile range, which is the difference between the third and the first quartile, is much smaller than in the uniform case. This is logical as there is much less probability on the extreme vales and the quartiles are therefore closer to zero. The absolute skewness is smaller than in the uniform case, and in this case negative. Also in this case the mean is not significantly different from zero. The extreme values are there with smaller probability and a smaller magnitude. In this case, the worst value is -60,000, which corresponds to about 37% of the objective function value.

Finally, we will consider the normal probability distribution. We have taken the nominal value as the mean and taken a 10% standard deviation. In this case, about 95% of the cases fall inside of the required interval. If the random number falls outside of the interval [-20%,+20%], we have rejected this number and drawn a new one. This is repeated until the number is inside the above mentioned interval. The results can be seen in Figure 4.4. It is interesting to note that, despite the truncation, this distribution still has the longest tails. Looking at the quartiles again, in Table 4.14, we see that they are very similar to the uniform case. The tails are slightly longer, but with a smaller proba-

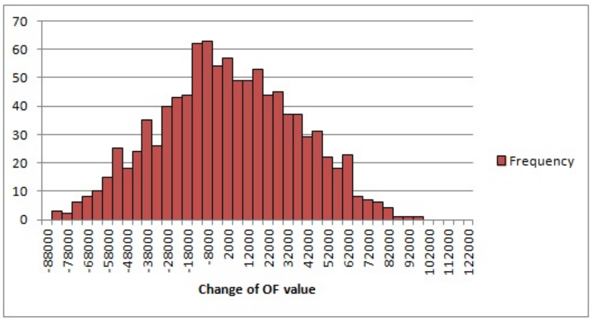

Figure 4.2: Monte Carlo realizations with respect to the change in the nominal value based on a uniform distribution

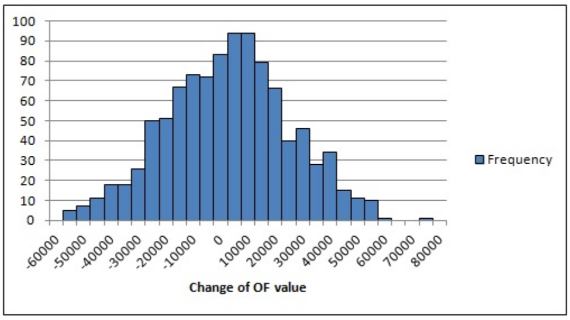

Figure 4.3: Monte Carlo realizations with respect to the change in the nominal value based on a triangular distribution

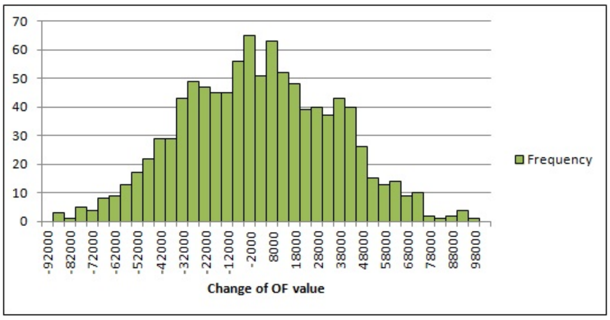

Figure 4.4: Monte Carlo realizations with respect to the change in the nominal value based on a doubly-truncated normal distribution

bility. The skewness is again close to zero, which indicates a very symmetrical distribution. The worst case value is -91,000, which corresponds to about 56% of the objective function value.

In all cases the skewness is small and the mean is not significantly different from zero, which indicates a symmetrical distribution around zero. Therefore, a change in the parameters can also have a positive effect on the objective function value. The results depend heavily on the choice of distribution. Nevertheless, in all cases the extreme values are at about 35% to 55% of the objective function value, which is a significant amount. Therefore, the extreme values seem to indicate that the model is not very robust with respect to changes in the uncertain parameters.

However, it is debatable whether it is realistic to include a lot of parameters, all with an independent interval of 20% around the nominal value. This results in a so-called 'polyhedral uncertainty'. The uncertainty set takes the form of a 6-dimensional polyhedron. In our case, the uncertainty region can be considered a 6-dimensional cube. It might be a bit extreme to consider the areas around the corners of that cube as admissible, as this would correspond to all 6 parameters taking an (almost) extreme value of their uncertainty set. One way to solve this is by taking an ellipsoidal uncertainty, but we have chosen an alternative solution.

We have just considered the impact of parameter uncertainty on the optimal solution of our problem. It occurs that the largest impact is caused by changes in the revenues, gas costs and demand. To overcome the uncertainty in the revenues and gas costs, these can be fixed by signing a contract with the power plant, which is exactly what we have assumed in Chapter 3. The other costs can also be fixed but do not seem to generate a big impact. For instance, of the total deviation, the $ca_{i,t}$ and $cu_{i,t}$ together account for about 3% of the objective function value. Demand can however never be completely accurately predicted. This is a pity, since demand alone gives extreme values of about 6% to 10% of the objective function value. Therefore, in the next chapter, we have chosen to make the model robust with respect to demand.

Chapter 5

Robust optimization

This chapter is about the robustness of the model. We start with an introduction to robustness in Section 5.1. Then we introduce the concept of an Adjustable Robust Counterpart in Section 5.2. We discuss the application of this theory in Section 5.3, which is followed by the description of the implementation to our model in Section 5.4. Finally, the results are shown in Section 5.5.

5.1 Introduction to robustness

In real life, data is often uncertain. Being uncertain means not knowing the exact values of certain input parameters of the problem exactly at the time the problem is being solved, according to Ben-Tal et al [4]. Possible reasons for this uncertainty include impossibility to measure and estimate the data exactly. Weather, for instance, can cause uncertainty. Also complex (technical) processes can hardly be 100%-accurately predicted. Moreover, it is not always possible to implement a solution exactly as it is computed. A nominal solution with an accuracy of 5 digits after the decimal mark, which has to be implemented just as accurately, will lead to trouble. Because of these possible errors, the nominal solution could become completely meaningless. In our case the demand is uncertain and cannot be exactly predicted. As we have seen in Section 4.8, uncertainty in demand can have a big influence on the quality of the solution. Therefore, in this chapter, we make the model robust with respect to demand, that is, we try to find the best solution that is insensitive to changes in demand within the uncertainty region. We start with a small introduction on robustness, robust optimization and adjustable robust optimization. This is to illustrate the methods we have used. Those looking for a more in-depth research on this subject should study the book of Ben-Tal, El Ghaoui, and Nemirovski [4].

Let us start with a basic example to explain robust optimization.

Example 5.1. A company produces cappuccino. This is made by 1 unit of coffee and 1 unit of milk. Both coffee and milk come in packs, which contain 10 units of coffee or milk, respectively. For the packs of coffee, there is a choice. Packs of type A are slightly cheaper than packs of type B. Cappuccino sells for 1.20 euros, while the costs of producing it are simply the added costs of coffee and milk. The company has to decide how many packs to buy of which type, and how much cappuccino to produce. The goal is to optimize profit. The data are given in Table 5.1.

Parameter	Pack Coffee A	Pack Coffee B	Pack of Milk
Price of the pack (euros)	1	1.001	0.1
Content of the pack (units)	1	1	1
Maximal amount of packs	100	100	100

Table 5.1: Coffee example

We can now formulate the problem as an LP. We shall denote the amount of packs of coffee A by 'a', the amount of packs of coffee B by 'b', the amount of packs of milk by 'm' and finally the amount of cappuccinos produced by 'c':

$$\begin{aligned} &objective\, function: && \max\, 1.20c\, -a\, -1.001b\, -0.1m \\ &subject\, to: && c\, -a\, -b \le 0 \\ & && c\, -m \le 0 \\ & && a\, +b \le 100 \\ & && m\, \le 100 \\ & && a, b, c, m \ge 0\, . \end{aligned}$$

The optimal solution for our LP is $a = 100$, $b = 0$, $c = 100$, $m = 100$, with an objective function value of 10 euros.

Even in this very simple problem, the data do not have to be certain. Let us suppose that contents of a pack of coffee are not fixed, but are modeled as the expected contents per pack and the real content is only known after the packs have been ordered. We assume that the real content drifts in a 2% margin around the nominal value for pack A, and in a 0.5% margin around pack B. As a result, the contents of pack A are in the segment of [0.98, 1.02] and the contents of pack B are in the segment of [0.995, 1.005]. Moreover, assume that the contents take the two extreme values with probability 0.5, as is done in [4]. How do these small pertubations of the content affect the solution?

Consider the optimal nominal solution of buying 100 packs of A, producing 100 units of cappuccino, and yielding a profit of 10 euros. With probability 0.5, the actual content of pack A is less than 1 (namely 0.98) and this production plan becomes infeasible. The simplest solution would be to make less cappuccino, namely only 98 units, with the same costs as above. The objective function value becomes 7.60 euros, which is a decrease of 24%.
We see that already a small, unavoidable deviation may make the nominal solution infeasible and the new feasible solution much worse than the nominal one. That is, small changes in the uncertain parameters can heavily influence the quality of the solution. Therefore, it is necessary to generate a robust solution which is immunized against uncertainty.

Let us consider a standard Linear Programming (LP) problem,

$$\min_{x} \left\{ c^T x + d : Ax \le b \right\}, \tag{5.1}$$

with $x \in \mathcal{R}^n$ is the vector of decision variables, $c \in \mathcal{R}^n$ and $d \in \mathcal{R}$ form the objective, $b \in \mathcal{R}^m$ is the right hand side vector. A is an $m \times n$ constraint matrix.

Now let us define an uncertain LP problem as

$$\{\min_x \{c^T x + d : Ax \leq b\} : (c, A, B) \in \mathcal{U}\}. \tag{5.2}$$

This is a standard LP problem, with the data (A, b, c, d) contained in the uncertainty set $\mathcal{U}$. How to handle such a problem? Solving the LPs for all possible data in the uncertainty set leads to solving an infinite amount of LPs, which gives an infinite amount of solutions. Hence, this idea cannot be used.

Let us first create some structure. A Robust Optimization (RO) environment has three implicit assumptions:

1. The decision vector has to represent a 'here and now' decision. The decision has to be made before the real values are known.

2. The decision maker is fully responsible for the consequences of the decisions for all data within the uncertainty set $\mathcal{U}$.

3. The constraints are hard and cannot be violated.

These implicit assumptions lead to a solution which has to be robust against all data in the uncertainty region $\mathcal{U}$. More specifically, the *robust feasible* solution should satisfy the constraints for all realizations of the data in the uncertainty set. Hence, the robust value, which we shall define as $\hat{c}(x)$, is the largest value of the objective over all realizations of the data in the uncertainty set. This can be considered as a sort of minimax problem, namely maximizing the worst case within the uncertainty region. We can make the problem of (5.2) robust. This is called the *Robust Counterpart* (RC) of the LP problem.

Definition 5.1. The Robust Counterpart of the uncertain LP problem is formulated as follows:

$$\min_x \left\{ \hat{c}(x) = \sup_{(c,d,A,b) \in \mathcal{U}} [\, c^T x + d] : Ax \leq b \, \forall \, (c, d, A, b) \in \mathcal{U} \right\}, \tag{5.3}$$

which minimizes the robust value of the objective over all robust feasible solutions to the uncertain problem.

Example 5.2. Let us reconsider the LP problem in Example 5.1 and find its robust optimal solution. We assume that only the contents of the coffee packs are uncertainty-affected. Since the (candidate) solution is only robust feasible when it satisfies all constraints for all the data in the uncertainty set, we will have to take the worst case scenario and subsequently optimize. As such, we reformulate the LP:

$$\begin{aligned} robust\,OF: \quad & \max\ 1.20c - a - 1.001b - 0.1m \\ subject\,to: \quad & c - 0.98a - 0.995b \leq 0 \\ & c - m \leq 0 \\ & a + b \leq 100 \\ & m \leq 100 \\ & a, b, c, m \geq 0\,. \end{aligned}$$

The optimal solution for this LP is $a = 0$, $b = 100$, $c = 99.5$, $m = 99.5$, with an objective function value of 9.35 euros. This is a decrease of only 6.5%, which is much less than the 24% we discussed earlier. This is also easy to explain. The nominal solution will always go for the cheapest option without considering any uncertainty risks, while the robust optimal solution takes into account that the uncertainty in pack B is 4 times smaller than in pack A, while its price is only slightly higher.

We end this section by making some observations:

Observation 1. An uncertain LP problem can be rewritten in such a way that all uncertainty is removed from the objective function:

$$\min_{x,t} \left\{ t: \begin{array}{ll} c^T x - t & \leq -d \\ Ax & \leq b \end{array} \right\}. \tag{5.4}$$

Note that the new objective t is not affected by uncertainty, while the RC remains the same.

Observation 2. Because an uncertain LP problem is with a certain objective, we only have uncertainty in the constraints. We can now replace the original constraints by the Robust Counterpart which has to hold for all the data in the uncertainty set. In this way, we get constraint-wise uncertainty. We can have constraint-wise uncertainty sets $\mathcal{U}_i$ for the uncertain data in constraint i, and let $\mathcal{U}$ be the direct product of these sets. Because the optimal solution will be an extreme point, we can extend the uncertainty set to its convex hull. Any robust feasible solution will still be robust feasible. Therefore, we may assume that the sets $\mathcal{U}_i$ are closed and convex, and that $\mathcal{U}$ is the product of these sets.

Note that (5.4) is still a difficult problem, as allowing for any $\mathcal{U}$ gives us the problem that there are infinitely many constraints. It is shown in [4] that the tractability (complexity of the problem) of a RC of a LP problem is tractable if the set $\mathcal{U}$ in itself is tractable. Therefore, in practice, the uncertainty is often assumed to be polyhedral or ellipsoidal. If $\mathcal{U}$ is a polyhedron (including interval uncertainty), the problem becomes an LP. If $\mathcal{U}$ is an ellipsoid, the problem becomes conic quadratic.

We have given a brief introduction into the world of RO. We have seen that already in a very basic example robustness plays a crucial role. In the previous chapter we have seen that the nominal solution of our problem is not robust with respect to demand. Therefore, it is imperative to make a more robust model. Before doing that, we will go one step further, namely by extending the research to Adjustable Robust Optimization. This means that our decisions can adjust themselves to a part of the uncertain data, which is demand in our case.

5.2 Adjustable Robust Counterpart

There are situations where it is very restrictive to only allow 'here and now' decisions. For instance, in an inventory system, the order on day 30 will depend on the amount of inventory on day 29. On day 1, it will not be possible to plan and fix the order on day 30. However, on day 29, we know the inventory level of that day and the demand of day 28. In the planning on day 1 we can take this knowledge into account, that we will have more information on day 29. In this way, some of the decision variables can adjust themselves to actual values of the data. This is called a 'wait and see' decision, which could be made after a part of the data has revealed itself.

In this section, we relax the first assumption of the robust optimization environment; not all variables have to be 'here and now' decisions anymore, but some are allowed to be 'wait and see' decisions. We will allow the variables to depend on a prescribed portion $P_j\zeta$ of the true data ζ: $x_j = X_j(P_j\zeta)$, where $P_1, \ldots, P_n$ are matrices specifying the information about the decisions x_j, given in advance. $X_j(\cdot)$ are *decision rules* to be chosen. These rules can be arbitrary functions. Note that only continuous variables are allowed to be adjustable. Having integer adjustable variables causes problems with the linear decision rule.

At time t, it is fully reasonable to assume that the information, say contents of the packs of coffee, of the previous time steps $t-1$ is known. In this case, $P_t\zeta = \zeta^{t-1} := [\zeta_1; \ldots; \zeta_{t-1}]$. The corresponding optimization problem is called the *Adjustable Robust Counterpart* (ARC):

$$\min_{X(\cdot),t} \left\{ t : A \begin{bmatrix} X_1(P_1(\zeta)) \\ \vdots \\ X_n(P_n(\zeta)) \end{bmatrix} - b \leq 0 \right\} \forall \zeta \in \mathcal{Z}. \tag{5.5}$$

Unfortunately, the ARC is very difficult to solve. In fact, it is typically severely computationally intractable [4]. This is why in practice the problem is restricted by only allowing for affine decision rules, which means that an adjustable variable will be expressed as an affine combination of the uncertain parameters on which it depends. This restricted version of the ARC is called the *Affinely Adjustable Robust Counterpart* (AARC).

Before we formulate the AARC, let us first introduce the concept of *fixed recourse*. A problem has fixed recourse if the coefficients of every adjustable variable do not depend on uncertain parameters, i.e. are certain. In other words, it is not allowed to have a product of an uncertain parameter and an adjustable variable in our problem. It is not possible to adjust the variable to the uncertainty when it is multiplied by an uncertain parameter.

In the AARC, the decision rules are restricted to affine ones:

$$x_j = X_j(P_j\,\zeta) = p_j + q_j^T\,P_j\,\zeta,\; j = 1, ..., n, \tag{5.6}$$

which leads us to the same situation as in Equation 5.3:

$$\min_{t, y=\{p_j, q_j\}} \left\{ t : \begin{array}{ll} \hat{c}_\zeta^T y + d_\zeta & \leq t \\ \hat{A}_\zeta y - b_\zeta & \leq 0 \end{array} \right\} \quad \forall \zeta \in \mathcal{Z}, \tag{5.7}$$

with $\hat{c}_\zeta^T$,d_ζ,$\hat{A}_\zeta$,b_ζ affine in ζ:

$$\begin{array}{ll} \hat{c}_\zeta^T y & = \sum_j c_\zeta^j [p_j + q_j^T P_j \zeta] \\ \hat{A}_\zeta y & = \sum_j A_\zeta^j [p_j + q_j^T P_j \zeta], \end{array} \qquad with\,[\,y = \{[p_j, q_j]\}_{j=1}^n].$$

which is as tractable as a RC [4]. The problem will become an LP or a conic quadratic problem. In (5.7), with affine decision rules, t, p and q are the variables of the problem, which in case of fixed recourse are linear in x and linear in the uncertainty. The actual decisions x are defined by these coefficients and the corresponding portions $P_j\zeta$ of the true data once these portions become known. Instead of affine decision rules, it is also possible to allow for quadratic decision rules. In that case the problem becomes a semidefinite programming problem (SDP), which is much harder to solve [21].

Note that both the ARC and the AARC are generalizations of the normal RC. The normal RC is a trivial case of an (A)ARC with all matrices P_j equal to zero. The robust optimal decision rules replace the constant decisions that are given by the RC. Now that we have introduced all the necessary robust optimization techniques, we will apply them to our problem. This will be discussed in the next section.

5.3 Application to our problem

In our problem, the demand is uncertain. In Section 4.8 we have seen that a change in demand can have a big impact on the objective function value. The demand we are considering cannot be predicted perfectly. For instance, weather factors influence the amount of heat needed. Therefore, we wish to make our model robust with respect to demand. We have chosen for a box-uncertainty of 20%, which means that the demand is uncertain in a range of [-20%, +20%] around the nominal values. We have also used an uncertainty of 20% in the Monte Carlo analysis in Section 4.8. Note that we are in the case of fixed recourse.

We have taken the transport as an adjustable variable in our model. It is also very logical to let the amount of commodities transported depend on the demand which can help to make a better prediction. The binary variables, as well as the production and intensity, are not allowed to be adjustable because they are not continuous. This has a practical reason as well, as CHPs are inflexible and the CHP decision has to be taken earlier. Next to transport, the only other continuous variable is the storage, which will also be taken into account. The portion of data on which the transportation depends will be demand in the previous time steps. Taking more steps into account increases the quality of the solution, but also the complexity and the running time of the model increase a lot. This is a tradeoff and only numerical results will show us the right amount of time steps to consider. If we do not take any time steps into account, then all matrices P_j are zero and the normal RC is obtained.

We have just elaborated on the application of the robustness theory to our problem. In the next section, we will discuss how this is implemented in our model.

5.4 Implementation

We have implemented this approach in AIMMS. This was done by adapting the existing model to its AARC. We started by making the demand uncertain, with a box-uncertainty of 20% around the nominal values of (5, 15, 80). The transport variable $T_{i,j,t,c}$ is made adjustable and dependent on the demand at time t. This gives the linear decision rule:

$$T_{j_1,j_2,t,c} = p_{j_1,j_2,t,c} + \sum_{i,t_2,c} (q_{i,t_2,c_2,j_1,j_2,t,c} \cdot d_{i,t_2,c_2}) \tag{5.8}$$

We can enter the portion of dependence by defining the relation between t_2 and t. Logical choices for this dependence are for instance one previous time period, or all previous time periods. This can be achieved by respectively $t_2+1 = t$ and $t_2 < t$.

Instead of making the storage adjustable, we have decided to eliminate the storage variable from the model by substitution. This is because we wish to avoid equalities in robust optimization. The storage equation (3.2) is a recursive one. Moreover it has to satisfy constraint (3.10), bounding the maximum amount of storage. Given that $S_{i,t=0,c} = 0$ we have the following equations:

$$\begin{aligned}
S_{i,t=1,c} &= & l_{i,S,c} * T_{i,S,t=1,c} - T_{S,i,t=1,c} &\leq ms_{i,t=1,c} \\
.S_{i,t=2,c} &= l_{i,i,c} * S_{i,t=1,c} & + l_{i,S,c} * T_{i,S,t=2,c} - T_{S,i,t=2,c} &\leq ms_{i,t=2,c} \\
&\vdots \\
S_{i,t=T,c} &= l_{i,i,c} * S_{i,t=(T-1),c} & + l_{i,S,c} * T_{i,S,t=T,c} - T_{S,i,t=T,c} &\leq ms_{i,t=T,c} \, .
\end{aligned}$$

These can be written as follows:

$$S_{i,t,c} = \sum_{t_2=1}^{t} l_{i,i,c}^{t-t_2} \cdot (l_{i,S,c} \cdot T_{i,S,t_2,c} - T_{S,i,t_2,c}) \leq ms_{i,t,c} \, . \tag{5.9}$$

We can substitute this equation for (3.2) and (3.10), eliminating the storage variable. We have now taken all the continuous variables into account. Before the model is ready to run, some more adaptations have to be done.

The demand equation (3.1) was adapted. This is done because an equality constraint containing an uncertain parameter (in this case, the demand) can only be satisfied if a variable is multiplied by this uncertain parameter. We can simply change the equation into:

$$d_{i,t,c} \leq P_{i,t,c} + \sum_j l_{j,i,c} \cdot T_{j,i,t,c} - \sum_j T_{i,j,t,c} \,\forall\, i,t,c \tag{5.10}$$

because in this case the demand is still satisfied, while buying more or selling less commodities than necessary is allowed. However, this will not be optimal. Hence, equality holds at the optimum.

The objective function uses the nominal values of the uncertain parameter d to optimize. The objective function value can be computed by substituting the decision rules, using the nominal value for the demand in this case.

5.5 Results

In this section, we will discuss the results of the AARC-approach. The computation time is dependent on the size of the model and on the portion of previous demands taken into account. We have analyzed the computation time for different sizes and different portions. The results are in Table 5.2.

Size (I/T)		10/12	15/12	10/18
	1	65.0	277.3	208.0
	2	185.4	920.6	631.6
Portion (#time steps)	3	268.1	>1000	>1000
	4	437.0		
	5	507.3		

Table 5.2: Computation time of the AARC-approach for different portion sizes.

Only in small cases, with 10 companies and 12 time steps, the solver can take a few portions into account. This is hardly surprising given the huge amount of variables. For instance, $(I, T) = (10, 12)$ with a portion of 1 time step has 106,000 variables. With a portion of 2 time steps this amount becomes 196,000. When considering slightly larger cases, of the size $(I, T) = (10, 18)$ or $(15, 12)$, only one or two time steps can be considered. If we consider smaller cases, of for instance 6 companies or 6 time steps, it is possible to analyze all previous time steps. If we would consider more time steps, it would lead to a better result, as the adjustable variable will have more freedom. The drawback is the increasing complexity.

We have run a few instances of different sizes. The question is if the AARC-solution improves on the worst case. Apart from allowing the transportation to be adjustable on demand of the previous time periods, it is also interesting to allow it for the same time period. In this case, a decision rule is created for all possible demands. The realization of the demand will lead to a transport decision made.

Size (I/T)	10/6	10/12	12/12
Adapted nominal solution	-211,582	-423,680	-47,648
RC solution	-21,178	-1,408	111,760
AARC with previous time step	-21,178	-1,408	111,760
AARC with current time step	-19,848	3,091	119,963
AARC with previous and current time step	-18,280	3,630	120,249

Table 5.3: OF value of the AARC-approach for different portion sizes, compared to its nominal and robust solution.

Table 5.3 shows some computational results. The nominal solution is infeasible because the demand is uncertain. In the worst case, demand is (6, 32, 96) and we can calculate the cost of fixing this infeasibility in the same way as we did in Section 4.8. By subtracting these costs from the nominal value, the adapted nominal solution is calculated. We see that the RC is much better than the adapted nominal solution. Moreover, the AARC indeed improves on the RC, but not always with a significant improvement. It seems that the AARC with one previous time step gives the same results as the RC. We do not know the cause for this. Our research indicates that the most effective improvement is given when taking the current time step as the portion of data. This gives a significantly better solution than the RC. The outcomes are very volatile with respect to the data and it would be very interesting to see how well the AARC-approach does on an actual case.

We can conclude that the AARC-approach indeed improves on the RC solution, but the size of the improvement is dependent on which portion of data is used. This approach is suitable for small models of a maximal size of about $(I, T) = (10, 12)$. When the number of companies or time steps increases, the computation time becomes too large.

Chapter 6

Conclusions and further research

In this chapter, we will conclude this thesis and give recommendations for further research.

6.1 Conclusions

This study was concerned with the meta-EMS problem. We have introduced the problem and posed our research questions and goals. Next, we considered the relevant literature for the current subject. The literature is mainly focused on households, whereas we are considering horticulture companies. Some of the ideas can be used, but these have to be adapted to our case of companies. We have motivated our assumptions and given a detailed problem definition, followed by the corresponding mathematical model. We formulated the problem as an MIP-model, we have discusses the complexity of the model and shown it to be NP-hard.

We have described the test problems and data generation. Then we analyzed the solution. The performance measure was minimal costs, and we have shown our cooperative model to improve on the current, non-cooperative situation. The computation time is exponential with the size of the instance. We have solved given test cases to optimality and improved on the heuristic that was made for these cases. Next, we have constructed our own, more complicated test cases. We have tried to improve the solving speed in several ways, but the default options by AIMMS perform well. We have seen that the model runs fast enough for models of smaller size, but when the size increases some heuristics might be required. We have proposed two heuristics, one based on aborting the solver and obtaining a very good bound, the other based on an LP-relaxation. A sensitivity analysis was performed and it appeared that the model is insensitive with respect to changes in demand. However, the model is not robust with respect to changes in demand, which is shown by a Monte Carlo analysis.

We have given an introduction to robustness and introduced the concept of (A)ARC. Finally, we have made the model robust against changes in demand by implementing the Affinely Adjustable Robust Counterpart (AARC). This improves on the worst case RC solution, but the size of the improvement is dependent on which portion of data is used. Our research indicates that the most effective improvement is given when taking the current time step as the portion of data. This approach is suitable for small models of a maximal size of about $(I, T) = (10, 12)$. When the number of companies or time steps increases, the computation time becomes too large.

6.2 Recommendations for further research

Directions for further research include building a good heuristic to tackle the large-size problems. Better data would lead to more accurate models, which do not have to be as generic as this one and could be faster, or at least more adapted to the real world. Moreover, it would be very interesting to apply the AARC-approach on a model with real data and analyze the results.

It would be interesting to build different heuristics for the larger models and see how well these perform. The LP-generation based heuristics are interesting, but also heuristics based on column generation might be effective for this type of problems.

An interesting spin-off subject is game theory in relation to our model. In Section 4.2 we compare the cooperative model with the non-cooperative model. This brings cooperative game theory to mind, and it would be interesting to research whether there exists a solution where every company is better off. Another topic in this direction is to calculate the amount of financial compensation that each company should receive for their efforts and commodities used.

Bibliography

[1] AIMMS (Advanced Integrated Multidimensional Modeling Software), www.aimms.com

[2] Akkermans, H., Schreinemakers, J. and Kok, K., 2004. Microeconomic distributed control: theory and application of multi-agent electronic markets, *Proceedings of CRIS 2004 - 2nd International Conference on Critical Infrastructures*, 2004.

[3] Alanne K, Saari A., 2004. Sustainable small-scale CHP technologies for buildings: the basis for multi-perspective decision-making. *Renewable and Sustainable Energy Reviews*, 8(5), pp.401-31.

[4] Ben-Tal, A., El Ghaoui, L. and Nemirovski, A., 2009. *Robust Optimization.* Princeton: Princeton University Press.

[5] Bertsekas, D.P. and Tseng, P., 1988. Relaxation methods for minimum cost ordinary and generalized network flow problems, *Operations Research*, 36, pp.93-114.

[6] Bosman, M.G.C., Bakker, V., Molderink, A., Hurink, J.L. and Smit, G.J.M., 2010a. Planning the production of a fleet of domestic combined heat and power generators, *European Journal of Operational Research*, January 2012, 216(1), pp.140–151.

[7] Bosman, M.G.C., Bakker, V., Molderink, A., Hurink, J.L. and Smit, G.J.M., 2010b. On the microchp scheduling problem, *Proceedings of the 3rd Global Conference on Power Control and Optimization (PCO)*, February 2010, Gold Coast, Australia.

[8] Bosman, M.G.C., Bakker, V., Molderink, A., Hurink, J.L. and Smit, G.J.M., 2011. Controlling a group of microCHPs: planning and realization, *First International Conference on Smart Grids, Green Communications and IT Energy-aware Technologies, ENERGY 2011*, 22-27 May 2011, Venice, Italy, pp.179-184. International Academy, Research, and Industry Association.

[9] Caldon, R., Patria, A. and Turri, R., 2004. Optimization algorithm for a virtual power plant operation, *39th International Universities Power Engineering Conference (UPEC)*, 3, pp.1058-1062.

[10] El Bakari, K., Myrzik, J.M.A. and Kling, W.L., 2009. Prospects of a virtual power plant to control a cluster of distributed generation and renewable energy sources, *Proceedings of the 44th International Universities Power Engineering Conference (UPEC)*, 1-4 Sept. 2009, pp.1-5.

[11] Even, S., Itai, A. and Shamir, A., 1976. On the complexity of timetable and multicommodity flow problems, *SIAM Journal on Computing*, 5(4), pp.691–703.

[12] Frank Phillipson, Senior Research Scientist at TNO, supervisor of this thesis.

[13] Houwing, M., Heijnen, P.W. and Bouwmans, I., 2006. Deciding on micro-CHP; A multi-level decision-making approach, *Proceedings of the 2006 IEEE International Conference on Networking, Sensing and Control, 2006 (ICNSC '06)*, pp.302-307.

[14] Kok, K., 2010. Multi-agent coordination in the electricity grid, from concept towards market introduction, *Proceedings of the 9th int. joint conf. on Autonomous Agents and Multiagent Systems*, industry track, 2010.

[15] Kok, K., 2011. Dynamic pricing as control mechanism, *Power and Energy Society General Meeting, 2011 IEEE*, pp.1-8.

[16] Kok, K., Scheepers, M. and Kamphuis, R., 2010. Intelligence in electricity networks for embedding renewables and distributed generation, *Intelligent Infrastructures*, Springer, Intelligent Systems, Control and Automation: Science and Engineering Series.

[17] Kok, K., Venekamp, G. and Macdougall, P., 2010. Market-based control in decentralized electrical power systems -extended version, *Electronic Markets*, 1(4), pp.61-66.

[18] LTO Noord Glaskracht (Land- en Tuinbouw Organisatie), www.ltonoordglaskracht.nl

[19] Molderink, A., 2011. *On the three-step methodology for Smart Grids.* Ph.D. thesis, University of Twente, The Netherlands.

[20] OCAP (Organic Carbondioxide for Assimilation of Plants), www.ocap.nl

[21] Vandenberghe, L. and Boyd, S., 1996. Semidefinite programming. *SIAM Review*, 38, pp.49-95.

[22] Wille-Haussmann, B., Erge, T. and Wittwer, C., 2009. Decentralised optimisation of cogeneration in virtual power plants. *Solar Energy*, 84(4), pp.604-611.

[23] Wright, A., Firth, S., 2007. The nature of domestic electricity loads and effects of time averaging on statistics and on-site generation calculations, *Applied Energy*, 84, pp.389-403.

List of Figures

List of Tables

Appendix A

Test cases

These are the test cases of Section 4.6. The possible starting time steps were randomly generated. It was allowed to generate the same numbers, which sometimes results in only three different possible starting time steps. Next to this, with 32 time steps and a duration of 8 time steps, it is not a possibility to start later than the 24th time step. The random generation did not account for this either. We have still shown the possibility, but selecting it will not lead to a feasible solution.

	Company	CHP capacity	d(E)	d(H)	Duration	Starting time steps
Supplying companies	1	280	0	0	8	1, 2, 7, 28
	2	290	0	0	8	6, 9, 11, 22
	3	300	0	0	8	3, 12, 14, 16
	4	310	0	0	8	2, 3, 10, 27
	5	320	0	0	8	11, 12, 25, 30
	6	330	0	0	8	10, 23, 28
Demanding companies	7	0	326	495	15	8
	8	0	299	1312	10	23
	9	0	1655	31	5	5
	10	0	20	1162	21	12
	11	0	1363	890	31	2
	12	0	592	129	25	8

Table A.1: Test case 1

	Company	CHP capacity	d(E)	d(H)	Duration	Starting time steps
Supplying companies	1	275	0	0	8	7, 21, 26, 31
	2	285	0	0	8	8, 9, 19, 28
	3	295	0	0	8	17, 18, 29, 30
	4	305	0	0	8	5, 9, 29, 31
	5	315	0	0	8	17, 20, 27
	6	325	0	0	8	16, 17, 26, 28
	7	335	0	0	8	4, 9, 17, 29
Demanding companies	8	0	167	306	2	2
	9	0	601	220	5	15
	10	0	180	451	26	7
	11	0	818	1227	16	17
	12	0	826	685	23	2
	13	0	1846	529	30	3
	14	0	1637	1062	32	1
	15	0	1386	389	21	6

Table A.2: Test case 2

	Company	CHP capacity	d(E)	d(H)	Duration	Starting time steps
Supplying companies	1	265	0	0	8	1, 2, 15, 20
	2	275	0	0	8	6, 14, 17, 21
	3	285	0	0	8	1, 5, 7, 15
	4	295	0	0	8	15, 16, 26
	5	305	0	0	8	18, 20, 24, 27
	6	315	0	0	8	11, 15, 20, 21
	7	325	0	0	8	8, 10, 18, 24
	8	335	0	0	8	5, 20, 25, 27
	9	345	0	0	8	3, 5, 11, 24
Demanding companies	10	0	903	1173	8	17
	11	0	742	146	14	10
	12	0	31	1171	1	13
	13	0	763	1027	31	2
	14	0	898	1180	29	4
	15	0	44	1134	6	13
	16	0	90	916	7	20
	17	0	279	1164	16	1
	18	0	867	454	16	17

Table A.3: Test case 3

	Company	CHP capacity	d(E)	d(H)	Duration	Starting time steps
Supplying companies	1	260	0	0	8	23, 31, 32
	2	270	0	0	8	4, 13, 16, 25
	3	280	0	0	8	6, 17, 18, 30
	4	290	0	0	8	5, 12, 21, 23
	5	300	0	0	8	5, 6, 10, 23
	6	310	0	0	8	2, 7, 10, 15
	7	320	0	0	8	20, 23, 27, 32
	8	330	0	0	8	1, 26, 29
	9	340	0	0	8	2, 6, 22, 32
	10	350	0	0	8	17, 19, 23
Demanding companies	11	0	1451	805	14	19
	12	0	471	1432	23	3
	13	0	772	1065	4	3
	14	0	618	63	23	10
	15	0	399	909	11	12
	16	0	798	864	19	6
	17	0	267	738	15	4
	18	0	4	840	23	1
	19	0	202	106	29	4
	20	0	616	308	16	3
	21	0	1154	1364	26	3

Table A.4: Test case 4

	Company	CHP capacity	d(E)	d(H)	Duration	Starting time steps
Supplying companies	1	250	0	0	8	21, 22, 28
	2	260	0	0	8	14, 25, 26, 30
	3	270	0	0	8	3, 18, 27
	4	280	0	0	8	16, 17, 23, 31
	5	290	0	0	8	1, 11, 16, 17
	6	300	0	0	8	5, 7, 10, 26
	7	310	0	0	8	4, 5, 26, 27
	8	320	0	0	8	4, 22, 30, 32
	9	330	0	0	8	2, 16, 19, 24
	10	340	0	0	8	10, 22, 27, 29
	11	350	0	0	8	7, 23, 31
	12	360	0	0	8	5, 22, 25, 28
Demanding companies	13	0	1454	579	30	3
	14	0	1803	15	3	30
	15	0	220	950	13	20
	16	0	213	263	3	29
	17	0	1024	185	9	2
	18	0	1719	668	1	6
	19	0	520	73	12	21
	20	0	1854	1092	6	14
	21	0	588	1388	18	15
	22	0	106	903	14	19
	23	0	1054	773	8	8
	24	0	1978	568	32	1

Table A.5: Test case 5