DEGREE PROJECT IN MATHEMATICS,
SECOND CYCLE, 30 CREDITS
STOCKHOLM, SWEDEN 2019

Optimization of Collateral allocation for Securities Lending

An Integer Linear Programming Approach

MAGNUS ORRSVEDEN

EMIL TARUKOSKI

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ENGINEERING SCIENCES

Optimization of Collateral allocation for Securities Lending

An Integer Linear Programming Approach

MAGNUS ORRSVEDEN

EMIL TARUKOSKI

Degree Projects in Optimization and Systems Theory (30 ECTS credits)
Engineering Programme – Industrial Engineering and Management, 300 hp
Master's Programme in Industrial Engineering and Management, 120 hp
KTH Royal Institute of Technology year 2019
Supervisor at SEB: Fredrik Niveman
Supervisor at KTH: Anders Forsgren
Examiner at KTH: Anders Forsgren

TRITA-SCI-GRU 2019:105
MAT-E 2019:60

Royal Institute of Technology
School of Engineering Sciences
KTH SCI
SE-100 44 Stockholm, Sweden
URL: www.kth.se/sci

ABSTRACT

Collateral management has, during the most recent years, been an increasingly important part of a bank's operation. The bank is facing an allocation problem of how to post collateral to all its counterparties in order to mitigate the credit risk and the number of transactions that requires collateralization is increasing. This master thesis has investigated if it is possible to effectively solve this allocation problem and hence reduce the cost of collateral management by using numerical optimization. Four mathematical linear optimization models of different structure and characteristics have been developed which aims to reflect the complex nature of the problem.

These models have been solved with real-life data and it can be concluded that optimization can be used in order to reduce the cost of the collateral allocation and that the problem can be efficiently solved in an acceptable amount of time. The solution showed a reduced cost of almost 15% for the selected business day compared to the cost of current collateral allocations.

Keywords: *Collateral Management, Collateral optimization, Operations Research, Linear Programming*

SAMMANFATTNING

Säkerhetshantering har under de senaste åren varit en allt viktigare del av en banks verksamhet. Banken står inför ett allokeringsproblem kring vilka säkerheter som ska ställas till dess motparter för att minska den kreditrisk som uppkommer i samband med lånehandeln samt att antalet transaktioner som kräver ett säkerställande ökar. Denna uppsats har undersökt möjligheten till att effektivt lösa detta allokeringsproblem och därigenom minska kostnaden för säkerhetshantering genom att använda numerisk optimering. Fyra matematiska linjära optimeringsmodeller av olika strukturer och egenskaper har utvecklats som syftar till att återspegla problemets komplexa natur.

Dessa modeller har lösts med verklig data och slutsatsen är att optimering kan användas för att minska kostnaden för en banks säkerställande och att problemet effektivt kan lösas inom en acceptabel tidsram. Lösningen visade en reducerad kostnad på nästan 15% för den valda arbetsdagen jämfört med kostnaden för nuvarande allokering av säkerheter.

Nyckelord: *Collateral Management, Collateral optimization, Operations Research, Linear Programming*

ACKNOWLEDGEMENTS

First and foremost, we would like to express our deep gratitude to the Market Finance and Collateral Management teams at Scadinaviska Enskilda Banken (SEB) for the opportunity to conduct this project within collateral optimization as our master thesis. We would also like to give a special thanks to our supervisor Fredrik Niveman, for his guidance and professional insights during our project.

Furthermore, we would like to thank Anders Forsgren, professor at KTH Royal Institute of Technology, for sharing his valuable knowledge and for the support that he has provided during this thesis.

Stockholm, May 2019

Magnus Orrsveden & *Emil Tarukoski*

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Part I

INTRODUCTION TO THE ANALYSIS

INTRODUCTION 1

1.1 BACKGROUND

The financial crisis of 2007-2008 shook the world and resulted in collapsed stock markets and banks that were forced into default. One of the underlying causes for the financial crisis was that the collateralization of high-risk financial instruments was insufficient and that the risk aversion of countless banks and financial institutions was low. Financial instruments that were based on low-credit-worthy mortgages were sold and traded and the downturn of the American housing market resulted in what is known as the worst financial crisis since the Great Depression of the 1930s. Ever since the crisis, the field of collateral management, i.e. the process on how to secure financial transactions through collateralization, has gone through several regulatory changes. Additionally, the academic world has seen a significant increase of scientific papers regarding collateral management and the subject has been widely studied in recent years. It all boils down to risk management where a bank or a financial institution wants to minimize its credit risks of financial transactions to avoid a new financial crisis. [1]

The receivables that a bank has towards its counterparties generally generates a credit risk for the bank. These counterparties could, for example, be other commercial banks, hedge funds, central banks, state agencies and asset managers and the receivables could be stocks, repos or other securities that are lent out to the counterparty as a part of a financial transaction. If the counterparty should go in default, there is an implied risk that the counterparty will not be able to fulfill its commitment to the lender, i.e. the bank. In cases where the bank is in debt to the counterparty, the counterparty poses a risk that the bank may not complete its part of the agreement.

In order to reduce this mutual credit risk, it is common to agree on a process for securing the receivables and liabilities by mitigating the risk. This usually means that the counterparty transfers either bonds, equity, cash or other assets to the bank of a value that corresponds to, or exceeds, the receivable. This set of assets is transferred as securities to the counterparty, known as *collateral*. If the bank's receivable instead were to be a liability, it is the bank that transfers assets to the counterparty. The figure below illustrates if the bank has a liability towards the counterparty.

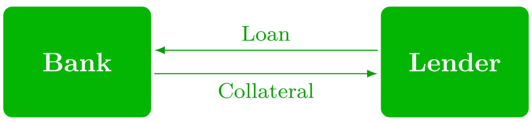

Figure 1: Illustration of a transaction where a bank post securities as collateral to a lender

When a financial transaction, that includes collateral, is planned, the terms of the collateral are defined in a bilateral agreement between the two parties. In the agreement, it is stated which type of security that is accepted as collateral, how the assets are to be valued and which terms that has been agreed upon, among other regulations. In accordance with each respective agreements, the bank has an opportunity to choose which assets are to be transferred to the respective counterparty. Each asset has an opportunity cost for the bank, i.e. a cost related to how the assets may have been used in other ways, and a cost related to the credit risk of the counterparty. Additionally there is a transaction cost related to the administrative handling of the transaction. By placing the assets as collateral in an optimal way, the sum of these costs may be reduced.

The differences in these agreements and the size of the required amount of collateral makes the decision regarding which asset to allocate to which counterparty a very complex and important decision. If the process is handled poorly, the allocation of collateral will be both expensive and time-consuming. Both the problem and the solution is to some extent intuitive for a situation where only a few available assets are to be distributed to 2-3 counterparties, but as the number of possible allocations grows, it soon becomes complex and hard to manually overlook. This then requires some system or algorithm to handle this process. Collateral optimization is about determining which allocation of collateral that results in the lowest total cost while simultaneously fulfilling the requirements and terms of the agreements.

1.2 PROBLEM STATEMENT

The complex structure and the large number of decisions that needs to be done, makes the allocation of collateral a suitable area to apply the methods and models of mathematical optimization. An effective model would, potentially, have a significant impact on the cost of the bank and how it handles its collateral management. To be able to test if this is possible, the following research question aims to be answered and discussed:

- *Is it possible for a bank to efficiently optimize its collateral allocation towards its counterparties?*

Furthermore, this gives rise to several questions that need to be answered before the research question can be evaluated.

- Which previous advancements have been made within the field of collateral optimization?
- What is the best-suited model formulation to use for this kind of problem?
- Will the optimization generate realistic cost-reductions compared to actual collateral postings?
- How to best address time-complexity when computationally solving the model and which solver is the most efficient?

1.3 AIM AND DEMARCATION

The aim of this master thesis is to solve an optimization problem of how to allocate collateral in a cost-optimal way. The optimization model aims to be as realistic as possible with multiple constraints that reflect the real-life situation and the challenges of the collateral management teams at Skandinaviska Enskilda Banken AB, further denoted as SEB. Furthermore, the results of the optimization are aimed to be actionable for SEB through a list of recommended transactions that correspond to the most optimal solution or a similar user-friendly and actionable output.

Regarding the scope and the demarcation of the thesis, the main subject that will be covered and analysed is stock loans within securities lending. Collateral are posted in other markets and in other financial transaction types as well, but to get a reasonable size and structure of the problem, this will be a delimitation. Additional limitations will involve the direction of these transactions where only the outgoing collateral positions will be handled, while decisions regarding how contracts for incoming collateral are handled, will be left out. More details regarding assumptions and additional delimitations will be further covered in the methodology section.

1.4 THESIS DISPOSITION

This thesis is divided into three main chapters; *Introduction to the analysis*, *Understanding of the principles* and *Results and discussion*. The introduction is followed by a literature review where previous research on collateral management and collateral optimization are covered. The second chapter starts with a theoretical background section in which the fundamental ideas of both collateral management and linear programming are covered. The chapter continues with the methodology section where the structure of the analysis and the methods used to find the results are described.

The final chapter of the thesis includes the results and discussion sections. In the results section, the most important results are presented in graphs followed by a discussion section where the results and implications are discussed. Lastly, the chapter is summed up by a conclusion section which includes the conclusion of the thesis and suggested areas for further research. The conclusion is followed by a list of references and an appendix where additional information and data tables are presented.

LITERATURE REVIEW 2

In this section, previous studies and papers regarding the subjects of relevance for this thesis are presented. The field of collateral management has been widely studied during the most recent years while the field of collateral optimization only offers a small number of relevant work, at least from a mathematical perspective. Due to this, this section will be divided into two areas. The review begins by focusing on the field of collateral management to provide a background on the subject followed by a summary of the existing research on collateral optimization.

2.1 COLLATERAL MANAGEMENT

Research of collateral management has been increasingly studied the recent years. This can be shown by using databases like *Web of Science*. For example, when searching for topics including *collateral management* and *risk* in Web of Science and filtering on relevant research fields, one gets 78 results of which 65 is from 2008 or later. In one of the papers published before 2008, the authors explicitly state that collaterals are an important part in both commercial and industrial loans but that the "academic literature addressing its role is small".[2]

As mentioned in the introduction, one of the key background factors to this is the financial crisis in 2008. This is further stated by the bank BNY Mellon in their informative white-paper where they conclude that: "The reformation and restructuring of the financial markets in the aftermath of the global financial crisis has had a profound effect on how collateral is used and managed".[1] Similar to BNY Mellon, several banks and financial institutions has published own articles describing the financial system, how they work with collateral and how the industry has changed.[1, 3, 4, 5] Furthermore, a few academic studies that have the crisis in focus have been published as well and therefore adds to the literature on collaterals and collateral management.[6, 7]

Most academic work on collateral management is focusing on risk management and the effects of collateral on the financial world, both when it comes to risk, capital structures and regulations. For example, Lorenzoni et al. (2007) and Rampini & Viswanathan (2010) both provide different dynamic models of collateralized financing in which collateral work as a constraint to the model.[8, 9] Additional research questions related to collateral are, among others, how collateral determines the capital structure [10], the role of collateral in defaulted bank loans to SMEs (small and medium-sized enterprises) [11], how collateral can reduce the credit risk of commercial banks [12], how collateral has played an important role in how financial landscape has changed the recent years and how it may shape the global demand/supply for collateral.[13]

These academic papers add value to the field of collateral management but do, as mentioned above, mainly focus on purely financial theories and the role of collateral from a financial perspective. The key takeaways and conclusions from previous reports on collateral management, mainly address the importance of collateral in the world of finance, its increasing demand due to regulations and its potential reduction of credit risks, etc. Since our thesis will focus on the mathematical implementations, rather than the financial theory, the results and the conclusions of the studies are of secondary interest while a solid understanding of the subject, including the research field, is primal. Even though this thesis will not be focusing on the financial theories of collateral management, an underlying understanding of the subject is fundamental to be able to set up the problem and to validate the results.

2.2 COLLATERAL OPTIMIZATION

One of the most crucial part of collateral management is to develop and find an automated way of allocating collateral. Since the algorithms often are the most valuable tangible assets for a company, studies can be difficult to find regarding the mathematical approach. Instead, the information published is more focused on the model set-up and characteristics. For example, the collateral is usually traded in an integer-valued number of assets which is why most research on this area implies the method of an integer linear programming approach but also that a non-linear programming approach could be worth considering.[14]

According to white papers, written by employees of industry actors, a vital part of the optimization is to formulate the optimization model and furthermore, its objective function. Generally, there are four main objectives that are mentioned when optimizing collateral allocation; to minimize the cost of collateral, to minimize the funding cost, to maximize the liquidity or funding capacity of the retained inventory (remained assets) or to minimize the operational cost by automating the total process. Funding costs do, in this case, represent the use of a specific asset or collateral and can be evaluated using external or internal benchmarks against different available positions.[15, 16]

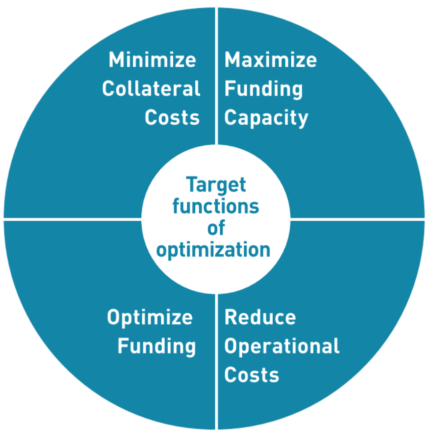

Figure 2: Four target functions of collateral optimization [16]

Furthermore, recently published articles concludes that it is especially important to maximize the remaining liquidity and funding. This due to new regulations as the act of Liquidity Coverage Ratio and the Net Stable Funding Ratio which forces banks to set aside more assets.[17]

The cost model is the foundation of how the optimization infrastructure will perform and drive the allocation process. There are currently a variety of ways on how to formulate a cost or allocation model. The main ones that has been mentioned in publications are Ranking Cost Model, Market Based Rankings, Economic Cost Model, Waterfall Allocation and Allocation using numerical optimizations.[14] Although a solution will contain market behaviors (credit quality, liquidity profiles, maturity, rating, etc.), Ranking cost models will not be considered since it is a sequential approach and not an automated method. An alternative approach is the economic cost model of using an asset as collateral, meaning that it calculates the cost of placing each position while also considering the risk- and operational cost of moving that collateral.

The two remaining models are known as allocation models. Waterfall allocation is not seen as a pure optimization model since it is based on rankings of collateral requirements. It first allocates the lowest ranked assets (the least desirable positions) to the highest ranked agreements by a defined cost model. This method simply needs a set of rules to iterate through the requirements rather than an optimization algorithm.

Now, the most complex, detailed and automated solution will come from a numerical optimization approach using linear programming. The target is to optimize the allocation of the total collateral inventory in a single process. When using a mathematical optimization algorithm, the authors indicates that it is very important that the right algorithm is chosen along with an accurate interpretation on how to translate the business problem into a mathematical formulation including both the objective function and its constraints. An additional key to gaining the ability to use optimization is to have a clear view of the inventory across the bank and to have standardized data inputs. If an optimization algorithm is implemented, the allocation process can be automated through optimal allocation suggestions based on the current allocations and inventory. Additionally, white papers that are discussing collateral optimization explains how time-complexity can be a big problem if the calibration and data handling is poorly executed. This is why knowledge in collateral management is as important as mathematical knowledge.[14, 18]

Although it exists industry actors that have been publishing white papers that include some discussion on collateral optimization, a sparse number of studies on collateral management uses mathematical theories related to optimization. Instead, the literature offers a broad database with information about collaterals and which role it has in the financial system after, and before, the financial crisis and which parameters it affects with a low focus on the mathematical theories behind the finance. Two of the few academic reports covering the mathematics of collateral optimization is a master thesis in Engineering Physics from Umeå Universitet and a dissertation from the North-West University in computer science. The first one solves a simplified version of the collateral allocation problem and the thesis does not entirely give a view of the real-life problem that a bank is facing due to the size and structure of the problem. Additionally, Bylund's problem is solved with central counterparties in focus which is one type of deal that a bank can use collaterals for.[19]

The second mentioned paper attempts to solve a more life-like problem by the usage of integer linear programming. However, the main focus of the thesis is to measure and investigate computation time based on different models rather than on the actual cost-savings that the model implies. Furthermore, the data sets that are used when testing the models are randomly generated and are not based on any actual organization. Additionally, the author mentions that the final model of his dissertation does not include constraints that could be considered as well as that it would be a good further research subject to apply a similar model on "real-world" data.[20].

With the existing academic publications on the field of study in mind, our master thesis aims to both solve a real-life-like problem for the bank while focusing both on applicability and the eventual cost-savings. This aims to fill the gap in the literature regarding mathematical collateral optimization by using existing and well-proven optimization methods and models applied to collateral management.

Part II

UNDERSTANDING OF PRINCIPLES

THEORETICAL BACKGROUND 3

In order to fully understand the analysis of this thesis, the theories and frameworks used will be explained more thoroughly in this section. Firstly, key concepts of collateral management will be introduced in order to give the reader an understanding of the dynamics of collaterals within the financial industry. This includes a broad description of markets where collateral typically is used, how a bank mitigates counterparty credit risk and which restrictions of bilateral agreements between the different settlement parties that exist. Secondly, the field of operations research, some fundamental concepts and methods of mathematical optimization will be presented and described.

3.1 COLLATERAL MANAGEMENT

Due to new regulations, more competition and lower margins, the demand for collateral has been increasing the recent years. Collateral management is nowadays a more business-critical function of a bank with links to multiple other functions like trading, risk management, balance sheet optimization and liquidity management.[21] The use of collateral is common in multiple different financial markets where the requirements and the role of collateral vary depending on the market. One well-known market where collateral is used is in the repurchasing agreements (Repo) market where a security is sold accompanied by the seller's commitment to repurchase it a given time later. This can often be used to give the seller of the security short-term financing since they receive cash. [3]

Another area where collaterals are used, is within securities lending. Here, a financial transaction is initiated when a security is borrowed and another security is pledged as collateral of the borrowed asset. The key difference between repos and securities lending is mainly regarding which type of asset that is borrowed by the borrower. A third market where collateral is used is within the over-the-counter derivatives market as well as in exchange-traded derivatives. In trades with derivatives, central counterparties (CCP:s) are used as a middle man between the counterparties of a bilateral agreement. By doing so, the CCP takes on the counterparties settlement risk and can reduce the risk through multilateral netting of multiple bilateral agreements and by receiving collateral. Key factors of collateral settlements in this type of market are initial margin (IM) and variation margin (VM) which is posted by both counterparties. The usage of collateral in these markets has increased since the financial crisis of 2008 due to new regulations, for example, EMIR which is a European regulation that requires a reporting structure and that liquid derivative contracts have to be cleared on a CCP. [1] The paragraphs above are summarized in Table 1.

Market/product area	Collateral usage
Repurchasing agreemenst (Repo)	Collateral to secure cash loans. Often used for short-term financing.
Securities lending	Collateral is pledged to secure a loaned security.
Cleared OTC derivatives & exchange-traded derivatives	Collateral including IM and VM that is exchanged between counterparties of a bilateral agreement and a CCP.

Table 1: Collateral markets and usage

As mentioned in the background section of the introduction, the simplest form of collateral agreement, a bilateral agreement, is easy to understand and easy to handle for the involved parties. Although, the real-life case of a bank, is often complex and difficult to overlook. In a real-life situation, each bank or party has several counterparties with different agreements and requirements to fulfill and handle. Each underlying security-loan that is to be secured with collateral is known as a *margin-set*. The details of a margin-set with one lending counterparty may not match the details of a borrower. This may require an additional transaction with another counterparty and a chain of transactions is initiated to be able to fulfill one initial deal. A large bank or financial institutions does often take on the role of intermediary (market maker) and offers to lend out a certain asset by lending it from a third party. This requires at least two bilateral agreements where collateral is to be posted to the lender and to be received from the borrower. The purpose of this trade could be for the borrower to get an exposure to the market to gain leverage from, or to hedge, a position.

The assets that are used as collateral has to be something that can be easily priced and will serve to reduce a loss in case of default of a counterparty. In other words, the assets have to be a liquid asset that can be used as *cash equivalent*. The type of assets that are accepted as collateral may differ per contract and the most common forms are for example cash, equities and different type of bonds, described in Table 2.

Each bilateral agreement between two counterparties describes how much collateral that should be posted, which type of asset that is accepted as collateral and may have different concentration limits for different attributes of the assets. The details of a bilateral agreement for collaterals are further described in section 3.1.2.[1] To be able to introduce a more detailed description of the collateral agreement, the fundamentals of risk management in collateral management will first be introduced in the section below.

Collateral Asset	Description
Cash	Cash in different currencies, for example, SEK, USD and EUR.
Corporate bond	A debt security issued by a company, backed by the payment ability of the company.
Covered Bond	A debt security issued by a bank or financial institution which is collateralized against a separate pool of assets.
Equity	The equity, or capital stock, is the shares of ownership of which a company is divided. Equity is defined as the difference between the value of a companies assets and its liabilities.
Financial Institution Bond	A bond used as insurance for banks and financial institutions against crime exposure.
Government Bond - Sovereign	Bonds issued by a national government to support government spending. Often considered to be risk-free since a government is unlikely to default.
Government Bond - Municipal	Bonds issued by local authorities as municipalities or a county to finance local capital expenditures.

Table 2: Common collateral assets

3.1.1 *Risk Management*

The main idea of collateral management is to reduce the risk of a financial transaction. The areas of risk management related to collateral management are mainly credit risk and market risk. Credit risk is the risk of a loss due to a borrower's failure to repay on a loan or to meet contractual obligations. Market risk is, as implied by the name, risk related to the dynamics of financial markets. To be able to manage the credit risk, the risk has to be quantified. There are several methods based on statistical analysis on historical data such as Probability of Default (PD), Loss-given-Default (LGD), Expected Loss (EL) and Value-at-Risk (VaR). Besides these analytical methods, which will not be explained more in detail due to the scope of this thesis, it is common that banks use the credit rate assessment of other institutes.[22]

Credit Ratings

A Credit Rating can best be described as a measure of one's ability to pay back its debt (creditworthiness). The measure is an evaluation of a debtor's credit risk and thereby a predicted forecast of the debtor's probability of defaulting and meet the legal obligations of its loan. A debtor, the one acquiring the loan, can be an individual, government, business, etc. The credit ratings are issued by a credit rating agency based on the available information acquired either officially by public information or non-public information provided directly by the debtor (in this case a corporation, financial institution or government, municipal and sovereign). The major credit rating agencies (CRAs) dominating this market is Standard&Poor's, Moody's and Fitch.[23]

Haircut

When collateral is being pledged to a counterparty, it is common that the collateral value is affected by a discount or a so-called haircut. This haircut factor reduces the collateral value compared to the theoretical market value of the securities posted as collateral. This allows the security lender to hold a small buffer of collateral to compensate for the market risk in terms of price volatility and liquidity. The size of the haircut is depending on the type of security pledged as collateral and may be specific per bilateral agreement. This is due to the fact that each counterparty may have a different risk assessment of a security. For example, equity from the OMXS30 index in Sweden may be considered a safer investment for a Swedish counterparty than for an American counterparty and a bond with a longer time to maturity is considered safer than a bond with a shorter time to maturity and will thereby yield a lower haircut. The haircut for each group of security to each counterparty is stated in the contract of the bilateral agreement. The less volatile, i.e. the lower the market risk, the lower the haircut. Additionally, the value of the haircut can be affected by the liquidity of the asset, where an asset that takes longer to liquidate probably will have a larger haircut. [1]

Modelling Risk through Cost

The risk management mentioned above is, as mentioned, one of the underlying cornerstones to collateral management. The risk management process results in the ability to model the cost of a certain collateral position in terms of opportunity costs, counterparty risk costs and transaction costs.

Opportunity Cost: The cost associated with a specific security corresponds to the attractiveness of that specific security. Consider two instruments, an equity and a bond. If the equity is more attractive than the bond in terms of alternative usage in another outgoing loan, or in any other area of the bank, then the equity will have a higher opportunity cost than the bond since more money could be generated from the equity than the bond. The value of that opportunity cost can be evaluated by calculating the value of a security in terms of SEK (swedish krona) and multiplied by how many base-points this security would otherwise have earned.

Counterparty Risk Cost: The cost associated with a specific counterparty corresponds to the risk of having bilateral contracts with that specific counterparty. This includes previously stated concept of the counterparty's credit rating, its PD, LGD, EL and other risk measures. By using these methods, a cost of capital can be estimated for having overexposed collateral to the counterpart. The overexposure corresponds to the difference between the market value of the posted collateral and the initial exposure (size of the initial loan). Hence, the overexposed collateral is exposed to a credit risk of the counterparty. This corresponds to a cost calculated as the difference between the market value of securities posted as collateral and the initial exposure multiplied by how many base-points it cost of having these overexposed securities at a specific counterpart.

Transaction Cost: The transaction cost of a financial transaction is often considered to be a fixed fee for making the transaction. A transaction cost could represent the commission, taxes, clearing, settlement- or ticket charges, etc., that is related to the trade.[24] These costs could be unique per counterparty of each trade and will in this thesis be considered as a fixed cost that is not depending on the size of the transaction.

3.1.2 *Collateral Agreements*

To summarize this section on collateral management, the key parameters and processes of collateral management all boil down to a collateral agreement that is to be fulfilled. This agreement is most often a bilateral agreement, where the specific details regarding a bilateral financial transaction are stated. The combination of several collateral agreements and cost functions derived from risk management for every margin-set and security forms the allocation problem that collateral management is facing and aims to solve efficiently.

The collateral agreements may include some, or all, of the following parameters [1]:

- *Exposure*
  The exposure of a collateral agreement is the value that is required to post to the counterparty of the agreement before a haircut and a margin. The amount is given in a specific currency and is equal to the value of the initial loan of the agreement.
- *Collateral margin*
  The lender of the agreement does often set a margin that requires the borrower to post additional collateral. The margins are used by the lender to cover the counterparty's credit risk. The margin implies that one counterparty should post at least the required collateral value, i.e. the value of the loan, increased by the margin to enter the agreement.
- *Haircut value*
  As previously described, the haircut of a specific asset may vary per contract depending on the risk assessment of the counterparties. The haircut may, therefore, be unique per security and counterparty.
- *Accepted collateral*
  The details regarding accepted collateral describe which type of assets from Table 2 above, that is accepted as collateral. Additionally, for each type of asset, attributes like the index of an equity, industry of a corporate bond and country of a government bond can be specified.
- *Concentration limits*
  In the concentration limits of an agreement, the accepted type of collateral is further detailed. If the agreement includes a concentration limit, it could be limited how much of a specific asset or type of asset that is accepted. For example, a contract may allow maximum $X\%$ of the collateral value in equities from a specific index or maximum $Y\%$ of government bonds. Here, concentration limits can be of many different shapes and may be in terms of collateral value or market value.

The key components that together make up the collateral value that has to be posted to each counterparty are illustrated in the figure below:

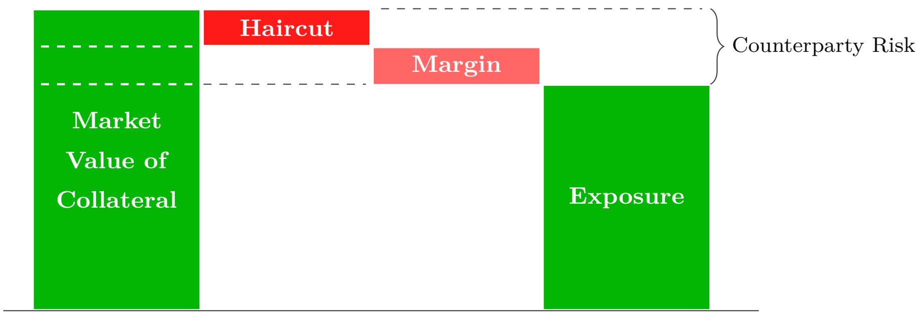

Figure 3: Illustration of the components that make up the market value of collateral

With a large number of collateral agreement schemes, costs associated with each counterparty, security and transaction, a bank faces a complex set of allocation problems which may be suitable for mathematical optimization. To give a simplified description of how this problem may look in the real world, consider the following example below.

EXAMPLE OF A COLLATERAL TRANSACTION Let's consider a simplified value-chain of an initiated stock-loan involving three parties, a borrower, a lender and a bank. The bank has a bilateral agreement with both the borrower and the lender. In this example, the borrower wants to initiate a stock-loan in order to short a stock. The borrower asks the bank for a loan of 10 equities in which case the bank finds a lender that has these 10 specific equities, as a market maker. Each loan requires the borrowing part to put out securities as collateral in case of default. In this example, only equities and government bonds are available and allowed as collateral. Meaning that the bank must give the lender collateral in exchange for the 10 equities and the borrower must give the bank collateral in exchange for the gathered 10 equities.[1] The two bilateral agreements have each an individual scheme for what securities that are accepted as collateral, in which amount and at what risk-level the counterparty values the different securities in exchange for the initial loan (consider a exposure amount of 100 SEK). In this case, the two following schemes are stated:

Agreement	**Bank-Borrower**		**Bank-Lender**	
Security	Equity	Government Bond	Equity	Government Bond
Margin	5%	5%	2%	2%
Haircut	20%	10%	15%	5%
Limit	50%	100%	30%	100%

Table 3: Simplified collateral scheme between the bank, borrower and lender

1 In reality, the stock-loan is in a much larger amount and the bank has several trades and exchanges between many counterparties in order to gather the required amount of equities.

In reality, this agreement schedule is much more complex and contains a larger set of securities and more specific concentration limits including country, industry sector, index etc. When a securities-lending transaction has been initiated, each party values their securities available as accepted collateral according to the risk-factors stated in their bilateral agreement. The market value of collateral is then valuated by:

$$Market\ value\ of\ Collateral = \frac{Exposure \cdot (1 + Margin)}{(1 - Haircut\ marginal)}$$

where each party can choose to either post equity or government bond as collateral according to the stated agreement. Here, the haircut has reduced the market value of the collateral which requires the poster to post additional units of each asset. This corresponds to:

Agreement	**Bank-Borrower**		**Bank-Lender**	
Security	Equity	Government Bond	Equity	Government Bond
incl. Margin	105	105	102	102
incl. Haircut	131.25	116.67	120.00	107.37
Limit	65.625	116.67	36.00	107.37

Table 4: Simplified collateral scheme stated in market value of collateral based on 100 SEK in exposure

If both counterparties of the bilateral agreements have an agreement (according to table 4) on the specific terms for the initial stock-loan an exchange can be executed (as seen in the simplified illustration in figure 4). The bank can initiate the stock-loan by gathering the 10 equities from the lender in exchange for either 36 SEK in equities (since there is a limit) and the rest in government bonds, 107.37 SEK in government bonds or a combination of both if that is acceptable in the agreement as collateral. When gathered, the bank can post the initial loan of 10 equities to the borrower while receiving collateral of either 65.625 SEK in equities anf the rest in government bonds, 116.67 SEK in government bonds or a combination of both. When posting collateral, each security, counterpart and transaction generates a cost that has to be taken into account which can make a specific security preferable to post in these two different trades. Finally, the borrower sells the loaned equity in hope that the stock will take a dive in the market in order for the borrower to buy back the same 10 equities at a lower price and then pay back its initial loan to the bank.

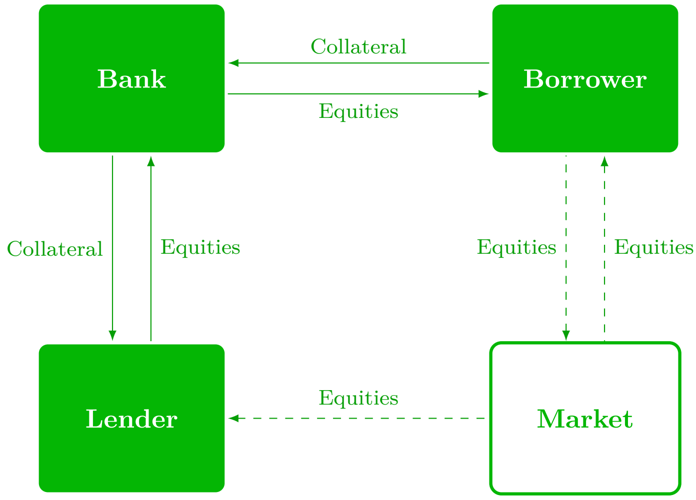

Figure 4: Illustration of the dynamics when a bank act as market maker

3.2 OPTIMIZATION

3.2.1 *Linear Programming*

The most basic form of an optimization problem is known as linear programming (LP). An LP-problem has an objective function that is to be optimized (minimized or maximized depending on the situation) subject to one or several linear constraints which sets the feasible region for the solution of the problem. Simple LP formulations can be applied to several areas across industries, for example, to help determine on which production mix a manufacturing company should have if the products have a known cost and the machines has a known capacity or how to in a cost-optimal way transport produced goods to customers while fulfilling every customer's demand. In these cases, the objective functions, as well as the constraints, will be linear and the results are continuous variables connected to the product of the route. The general form of an LP problem can be illustrated as:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & z = c^T x \\ \text{subject to} \quad & Ax \geq b, \\ & x \geq 0, \end{aligned} \tag{1}$$

where in this case, x is a non-zero variable vector, c is a cost vector associated with the variable x, b a vector containing the lower bound and matrix A constituting the constraint equation to the lower bound.

SIMPLEX method

One of the most common methods to solve an LP-problem is known as the simplex method and was first developed in the 1940s and is still today commonly used. The method effectively solves LP-problems and it may also be used as part of a solution algorithm for more advanced optimization problems like mixed-integer programming problems.

The main idea of the simplex method is to look for an optimal solution in adjacent vertices of the feasible set, i.e. the set of feasible point according to the constraints, which forms a polytope in linear optimization. The simplex method then begins at a starting extreme point and moves along edges of this polytope until it reaches the extreme point which is the optimum solution.

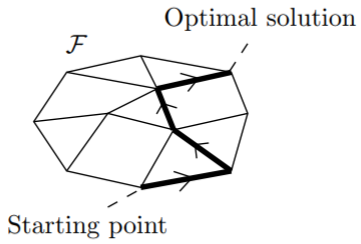

Figure 5: Illustration of the simplex method along with its formed polytope [25]

The objective function improves, or stays the same, for every iteration until it can no longer be improved and therefore is optimal. The method is often very efficient and the number of iterations that is required can be, with a few exceptions, around two to three times the number of constraints. The simplex method follows a well-defined algorithm that is described in detail in Appendix A.1.

Artificial Variables

When a linear problem cannot be solved due to infeasibility, it can be of interest to investigate why the problem is found to be infeasible.

There are therefore several techniques for investigating infeasibility. One popular set of methods is to find the source of infeasibility by using artificial variables. One common technique is called *the big-M method* which adds a penalty term to the objective function in order to find an initial basic feasible solution with the idea to push these artificial variables from the basis. An illustration of the big-M method applied to the standard linear programming approach is shown below.

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & z = c^T x \\ \text{subject to} \quad & Ax \geq b, \\ & x \geq 0 \end{aligned} \quad \rightarrow \quad \begin{aligned} \underset{x,a}{\text{minimize}} \quad & z = c^T x + \text{Ma} \\ \text{subject to} \quad & Ax + \text{a} \geq b, \\ & x \geq 0, a \geq 0, \end{aligned} \tag{2}$$

where M is a cost vector corresponding to the individual penalty term representing an arbitrarily large number for each artificial variable in the a vector. A good idea when computationally using M is to set a sufficiently large number in order to dominate the corresponding cost vector of its optimization variables. This means that a basis containing artificial variables will generate a large value for the linear program and if there exists a basic feasible solution to the original problem, the simplex algorithm will not have a corresponding basis including artificial variables in order to minimize its objective value.

A basic feasible solution to the problem generated in the big-M method where the artificial variables are defined as nonbasic variables is thereby also a basic feasible solution to the original problem and can be used as an initial basis for the original problem.[26]

3.2.2 *Integer Linear Programming*

When a problem involves variables that are restricted to be integers, it is defined as an integer problem. If the constraints are linear and all of the variables are restricted to be integers, it is an Integer Linear Programming problem (ILP), and if there are a mix of integer and continuous variables it is known as a Mixed Integer Linear Programming problem (MILP). The integer requirement makes the process of finding an optimal solution more complex since there is no convexity when there are integers involved. The integer requirement also reduces the number of feasible solutions and makes the ILP problem NP-hard to solve, which stands for non-deterministic polynomial-time, compared to LP-problems which can be solved efficiently using simplex or an interior point method. This indicates that it exists ILP-problems with a solving time that can increase exponentially or in other non-deterministic polynomial ways.

A large ILP problem is due to the integrality of one or several variables often hard to solve with an implicit enumeration approach that can be used for LP problems. Therefore, new methods that handle these kinds of problems have to be introduced.

Branch and Bound

The branch and bound method is one of the most used methods for solving general mixed, or pure, integer linear programming problems (MILP:s or ILP:s). The basic idea of the method, first introduced by A.H. Land and A.G. Doig, is to use a divide and conquer approach to break down the original ILP into subproblems of general linear programming problems (LP:s). These subproblems, the LP relaxations of the ILP, can be solved using conventional LP methods as described above, and be used to find the optimal solution of the ILP.

The method starts by solving an LP-relaxation of the original ILP, i.e. the same problem with the integer requirements relaxed. If the solution of the LP is not integer-valued for the variables with an integer requirement, the problem is branched into two new problems by adding a bound on the integer variables. For example, if y_j^R is a fractional solution of the LP for some j, two new branches can be restricted by $y_j \leq \lfloor y_j^R \rfloor$ and $y_j \geq \lceil y_j^R \rceil$ and the two new LP-relaxation will have disjoint feasible regions. Additionally, the solution of the first LP-relaxation will provide a lower (if minimization problem) bound for the ILP. As soon as an integer feasible solution is found in the branch and bound tree, it is saved as the best solution so far.[27, 28]

Each subproblem is either branched into two new subproblems or pruned and the search for the optimal solution continues until all branches are pruned. The branch and bound tree is pruned if a solution is integer feasible, optimal or worse than the best solution so far, or if the subproblem is infeasible. In a computational view, there are several options on how to branch the variables and how to search through the tree which may affect the speed of the method. [27, 28] A summary of the branch and bound algorithm can be found in Appendix A.2

Cutting Planes Methods

Another method to ensure that an ILP-problem can be solved is to use a cutting plane method. The main idea behind this type of method is to change the boundaries of the feasible region of the LP relaxation of the ILP-problem. By adding additional linear constraints, so-called cuts, to the LP relaxation, one can make sure that the optimal solution of the LP is integer-valued. The cuts slice the feasible region of the LP relaxation and make the original fractional solution infeasible. Such cuts are indefinitely many and the process of finding a suitable cut can be complex and time-consuming. [27]

To find the most suitable cutting plane, several cutting planes methods have been developed. The most common cutting planes methods are, amongst others, Gomory cuts, mixed integer rounding cuts and lift-and-project cuts. These cuts are more or less suitable for a different type of problems, for example, lift-and-project will perform best on a binary MILP. Modern cuts are often versions and improvements of the Gomory cuts or a mixed integer rounding cut (which implies a Gomory mixed integer cut for a certain type of problem).

Gomory's Cutting Plane Algorithm

One of the earliest techniques for generating cutting planes and thereby narrowing the optimization problem for a more efficient solution is Gomory cuts. This method was long considered an elegant theory but turned out to be a non-practical way of solving integer programs in practice. This mainly since it did not use its underlying structure, meaning that a large part of these cuts is frequently required for convergence. One of the biggest breakthroughs of this technique was when combing the Gomory's algorithm with the method of branch and bound, known as a branch and cut method that will be further explained below. It turned out, even regarding earlier remarks of its weaker cuts, that Gomory's cuts were both faster and more robust than a pure branch and bound. Gomory's mixed-integer cuts are still considered the most efficient way of solving mixed integer linear programs and used by most commercial software's within the field of Operations Research.[29]

The algorithm uses the simplex solution of the LP-relaxation of the problem to recursively add new constraints, cuts, to provide a feasible integer solution. These cuts will form a separating hyperplane which will separate the LP solution and the integer feasible region and the cuts are provided using the following algorithm:

Algorithm 1 GOMORY

Input: An LP problem in standard form **Output:** Cutting planes that will be used as additional constraints

(1) Start with an LP problem in a standard form in order to apply the simplex method.

(2) Iterate through the simplex method until it finds a solution. Select any non-integer constraint for a basic variable x_β, formulated as:

$$x_{\beta_i} + \sum_{j \in v} a_{i,j} x_{v_j} = [b_i] + f_i, \qquad \forall i, \quad x_v \in \{\text{non-basic variables}\}.$$

where f_i is b_is fractional part $(0 < f_i < 1)$.

(3) Partition all non-basic variables associated with $a_{i,j} \geq 0$ resp. $a_{i,j} < 0$ into two subsets P resp. N. Previous constraint is reformulated as:

$$x_{\beta_i} + \sum_{j \in P} a_{i,j} x_{v_j} + \sum_{j \in N} a_{i,j} x_{v_j} = [b_i] + f_i, \qquad \forall i.$$

(4) a) **If $[b_i] + f_i - x_{\beta_i} < 0$:** Then its integer part $[b_i] - x_{\beta_i}$ is negative, meaning that:

$$\begin{aligned} & \sum_{j \in P} a_{i,j} x_{v_j} + \sum_{j \in N} a_{i,j} x_{v_j} = ([b_i] - x_{\beta_i}) + f_i \geq f_i - 1, \\ \longrightarrow \quad & \sum_{j \in P} a_{i,j} x_{v_j} + \tfrac{f_i}{f_i - 1} \sum_{j \in N} a_{i,j} x_{v_j} \leq f_i \end{aligned}$$

b) **If $[b_i] + f_i - x_{\beta_i} > 0$:** Then its integer part $[b_i] - x_{\beta_i}$ is positive, meaning that:

$$\cdots \longrightarrow \quad -\sum_{j \in P} a_{i,j} x_{v_j} - \tfrac{f_i}{f_i - 1} \sum_{j \in N} a_{i,j} x_{v_j} \leq -f_i$$

Since this case implies same inequality, a slack is added, meaning that the new constraint is:

$$-\sum_{j \in P} a_{i,j} x_{v_j} - \tfrac{f_i}{f_i - 1} \sum_{j \in N} a_{i,j} x_{v_j} + s = -f_i$$

(5) Step (2) to (4) is repeated until all rhs for basic integer variables x_{β_i} is integers.

Figure 6: Gomory cuts for mixed integer linear problems [30]

Heuristics and Metaheuristics

To improve the performance of the branch and bound method explained above, a heuristic can be used to find a primal bound. If an upper primal bound exists, the number of nodes that will be pruned due to a less optimal objective value will increase. This will, therefore, result in a smaller and more narrow search tree which yields a more efficient implicit enumeration. Since ILP-problems are NP-hard, heuristics can sometimes be used alone to find a feasible solution as well. There are several heuristics (problem-specific) and metaheuristics (problem-independent) to use to find a primal bound and available optimization solvers often test and combine different methods. [31]

To be able to find a tight primal bound for the branch and bound tree, a first feasible solution has to be found through some start heuristic. To further tighten this bound, additional heuristics and metaheuristics can be used. The first step is to go from an initial LP-solution to a feasible integer solution. This can be made with some *diving heuristic*. The idea of a diving heuristic is to bound fractional variables to their closest integers in some direction, i.e. set the lower bound of a fractional x_j is $\lceil x_j \rceil$ or its upper bound to $\lfloor x_j \rfloor$. By iterating this process, the new LP solution will either be infeasible or eventually find an integer solution. There are multiple strategies to choose which variable to be bounded first. [31]

Another primal heuristic is called *The Feasibility Pump* and the basic idea is to construct one LP-feasible, but possibly not integer feasible, sequence of points and another sequence with integer feasible points. If a feasible integer solution exists, these two series will converge towards a feasible solution. The main algorithm is described below:

Algorithm 2 Feasibility Pump ALgorithm

Input: An LP-feasible solution $x^* := argmin\{c^T x : Ax \geq b\}$
Output: An integer feasible solution.

(1) Round x to nearest integer: $\tilde{x} = [x^*]$
(2) Solve new LP problem and update x^*:
$x^* := argmin\{\Delta(x, \tilde{x}) : Ax \geq b\}$
(3) a) **If** x^* is integer: Return x^* as an integer solution
b) **Else**: set $\tilde{x} = [x^*]$ and go to step (2).

Figure 7: Feasibility Pump algorithm to find a primal bound [32]

Additionally, there are primal heuristics based on rounding techniques. For example, the *RENS* (relaxation enforces neighborhood search) heuristic creates a sub-MIP of the original problem where each fractional variable of the optimal LP-solution is bounded to be either the lower rounding integer or the upper bounding integer. This results in additional constraints on the form $\lfloor x_j^* \rfloor \leq x_j \leq \lceil x_j^* \rceil$ for all j with fractional solutions and where x_j^* is the optimal LP-solution. Furthermore, heuristics based on local search methods and metaheuristics as tabu search and Simulated Annealing can be used to further refine the integer solution. This could result in an improvement from an initial non-optimal feasible integer solution or a local optimum to a solution closer to the global optimum. Due to the NP-hardness nature of the MILP or ILP problems, heuristics can sometimes be used to get a "good enough"-solution for large problems and can sometimes converge to the global optimum, but without any guarantees. [27]

Branch and Cut

When applying a cutting-plane to a MILP-problem one may struggle with slow convergence. The recursive process of adding cuts as new constraints may create large LP-problems, hence increase the time to find an optimal solution. Therefore, cutting plane methods are often integrated into the branch and bound algorithm. By combining the cutting plane idea into the branch and bound algorithm, one may end up with stronger linear relaxations for the subproblems which yield tighter bounds and thereby improved the efficiency of the branch and bound method. The branch and cut method do therefore require fewer iterations compared to the branch and bound method without cutting planes.

Modern solvers do often use a branch and cut process to solve a MILP in an efficient way by preprocessing the problem to reduce the size of the problem and by using several different primal heuristics to provide a tight primal bound. Additionally, the cuts of the nodes in the branch and bound tree can be performed by using several different cutting plane methods. Moreover, the solvers can often be adjusted to, for example, use a specific cutting plane method or to specify the allowed tolerance. [33]

3.2.3 *Sensitivity Analysis*

Sensitivity analysis is a set of postoptimum analysis tools used to determine how the solution of a problem changes when input data are changed. One of the most common changes that are to be analyzed is when the right-hand-side of the constraints are changed. Other input changes could be regarding the cost function in the objective or that new constraints are added to the model. The aim of sensitivity analysis is to answer the question on how the objective value would change if these changes were made, a "what-if"-analysis, and to do so without having to do the changes. Generally, the key idea is to analyze how optimality and feasibility are affected by an input change. [26]

Sensitivity analysis is well developed in linear programming and general rules can be formulated by using the duality theory of the simplex method. Since the solution of the LP-problem is computed through the simplex method, ranges of values, for example, a range of values for each b which yields an optimal solution, can be computed without any additional steps of the simplex method. These types of computations are often made automatically in LP solvers and the results of the sensitivity analysis can be called for after a solution has been given. [26]

However, for integer programming problems, the situation is much more complicated and non-trivial. Due to the NP-hardness nature of the MILP-problem, the computations of a value function, i.e. the objective value depending on the input data, is very hard and time-consuming. Although, it has been proven that the relationship between the dual and primal problem that is valid for an LP can be generalized for a MILP as well. This results in that sensitivity analysis can be carried out as long as a dual function can be constructed. Generally, the characteristics of, and the ease to find, a dual function to a MILP-problem is dependent on the algorithm used to solve the MILP and it is in some cases not computationally feasible to determine a dual function. These non-general forms and the unreliable computation times have made it difficult for solvers to implement an efficient sensitivity analysis of a MILP problem.[34, 35]

4 METHODOLOGY

This section contains a documentation of how data is gathered and used as input to the optimization models. The mathematical model formulations are explained by introducing the thoughts behind the structure and characteristics, the notations used in the model formulations, the objective functions and a thorough brief of the constraints. In total, 4 models will be introduced and described.

4.1 DATA COLLECTION

All data used in this paper has been provided by SEB and are, with some exceptions, real-life data on all contracts regarding stock-loans between the bank and its counterparties. The data has been gathered with help from supervisors at SEB and collected through spreadsheets and databases.

The data contains all relevant information regarding the margin sets with all counterparties and available securities for allocation as collateral. The input data files are structured to contain key information needed to perform the optimization model in a realistic way. In the following sections, the main input data needed for the models will be described.

The data that has been used for the analysis presented in the results section are data from one day in February 2019.

4.1.1 *Securities and Inventory*

Data regarding the available securities that can be posted as collateral were collected from different sources. Here, a lot of securities, each with a unique International Securities Identification Number (ISIN), were listed. The attributes of interest, independent on the type of security was mainly, ISIN-code, the currency of the ISIN with related exchange rate, the type of asset, index (mainly for equities), country of origin, cost associated with the ISIN as described in section 3.1.1 and the market price of the security.

Additionally, the number of available units that can be posted had to be an input parameter for each ISIN. This number is the combined value of previously posted units and units available in stock at the bank. This due to the fact that as long as a change is feasible and all positions are filled, a certain ISIN posted as collateral can be brought home and then posted to another collateral position if it is favourable.

4.1.2 *Counterparty*

Data on the counterparties and their margin sets that were required for the models were mainly the cost related to each counterparty. These costs were given to us and had been calculated according to risk management processes as broadly described in section 3.1.1 above.

4.1.3 *Agreements and Concentration Limits*

The most important data that makes the problem complex is regarding the bilateral agreements, i.e. the margin-sets, with every counterparty. As described above, each margin-set describes what type of collateral that can be posted, how much that has to be posted and if there are more detailed concentration limits for that specific contract. Therefore, data had to be collected that describes all of these points.

The most essential input data here id the required amount, initial margin, accepted collateral in terms of asset type, origin and index belonging, a haircut as well as a specification of the concentration limits that each contract may have that overrules the general acceptance of a security. Additionally, the opportunity exist to set a specific transaction cost, for each counterparty and security, to model a real-life scenario where specific counterparties are easier to trade with than others. Although, this has not been done for the models of this thesis, it can easily be added. This in due to the fact that no exact data on the transaction costs toward the counterparties were available. Instead, the number of transactions was reduced with the usage of a fixed limit on the transactions. By combining these data points each bilateral agreement could be described in terms of data and be used as input parameters for the optimization.

4.1.4 *Previous Collateral Posted*

In order for the model do detect which transactions and re-locations that has to be made, it is essential to know previous collateral postings. This means that data on all securities (ISIN) that is posted as collateral and to which margin-set it is posted, is required as an input.

4.2 ASSUMPTIONS

To be able to use the collected data for the optimization models, some assumptions had to made during the process of finding a solution.

- Relocating a specific asset that is posted as collateral is always possible if collateral agreement allows it. If nothing is stated in the concentration limits or agreement data, all counterparties are considered to be equal and for example, tri-party solutions and specific restrictions, are initially disregarded in the models.

- All inventory that has ISIN-data as price, exchange rate and ISIN-cost, is seen as available inventory. If no ISIN-data exists, collateral posted for that ISIN will be removed from the previous postings since necessary information is missing.
- It is not possible to execute an infinite amount of transactions due to human involvement in the transaction process.
- If previous collateral postings are infeasible according to the model setup, the constraints are adjusted to make this an initial feasible solution. This mainly applies to the requirement constraints.
- If the previous allocations are to be fractional, it must be rounded to the nearest integer in order for the algorithm to find a feasible solution (this only applies to the models with integer constraints).
- All financial data are gathered from the same time period and minor data errors have been adjusted.
- The transaction costs related to each bilateral agreement are disregarded, i.e. set to be zero, and the number of transactions is instead limited by an upper limit of transactions allowed.

4.3 OPTIMIZATION MODELS

Multiple mathematical optimization models have been used to answer the objectives of this thesis. These models differ in terms of the objective function and on the characteristics of the variables. These models have further been analyzed and discussed based on their solutions and solution times. To be able to find an optimal model for the problem, four models, that differs in terms of constraints and objective functions, have been used and validated.

Model 1 - LP-problem

The first model is a pure LP-problem where the variables describing how much of one ISIN that is allocated to each margin-set are allowed to take continuous values greater than, or equal to, zero where transactions is ignored. This model gives the theoretical lower bound but is less applicable in the real world due to the integer nature of the variables.

Model 2 - MILP-problem with transaction limit

The second model is a MILP-problem with an limit of transactions that are allowed compared to a previous allocations. This increases the applicability of the solution since the presence of a transaction limit in the constraints, and the possibility to add a transaction cost in the objective function, will inhibit the number transactions which are more lifelike due to a human factor in the execution of transactions. This is a MILP-problem since the variable representing the transactions is a binary variable (integer with value 0 or 1).

Model 3 - ILP-problem

The third model is a pure ILP-problem without any transactions. This model improves the applicability of the solution due to the integer constraints of the variables, i.e. it is only possible to post a quantity of a security that corresponds to an integer.

Model 4 - ILP-problem with transaction limit

The fourth, and final, model is an ILP-problem with transaction costs and a transaction limit. This model is the most realistic and life-like.

4.3.1 *Mathematical Notations - Sets and Parameters*

In order to formulate these four models, definitions and notations had to be stated. The following indexes and sets are used for the models stated above:

- Let i refer to a specific security (asset), $i \in S$, where $S = \{1, 2, \dots\}$.
- Let j refer to a specific margin-set, $j \in C$, where $C = \{1, 2, \dots\}$.
  - Let $S^{(t)}$ refer to a subset of S with securities of a specific type, bond, equity, country, industry sector etc, $t \in \{1, 2, ..., T\}$. $\bigcup_{t=1}^{T} S^{(t)} = S$.

The following notations of the parameters have been used:

mp_i	the market price of security i.
FX_i	the exchange rate to SEK of security i.
$h_{i,j}$	the associated risk of a specific security i for margin-set j called *haircut*.
R_j	the required amount of securities for margin-set j.
IM_j	the initial margin of margin-set j.
I_i	the available inventory of security i in number of units.
c_i^o	the opportunity-cost associated with security i.
c_j^r	the cost associated with the risk of margin-set j.
$c_{i,j}^t$	the transaction cost associated with allocation of security i to margin-set j. (All set to 0 in this thesis).
$x_{i,j}^0$	previous allocation of security i to margin-set j.
M	maximal number of transactions $y_{i,j}$.

$Pm_{i,j}^{(t)}$ maximal percentage market value of security i for margin-set j and type t.
$Pc_{i,j}^{(t)}$ maximal percentage collateral value of security i for margin-set j and type t.
$Cm_{i,j}^{(t)}$ maximal nominal currency value of security i for margin-set j and type t.
$Cc_{i,j}^{(t)}$ maximal nominal currency value of security i for margin-set j and type t.
$Pm_{j}^{(t)}$ maximal percentage market value of type t for margin-set j.
$Pc_{j}^{(t)}$ maximal percentage collateral value of type t for margin-set j.
$Cm_{j}^{(t)}$ maximal nominal currency value of type t for margin-set j.
$Cc_{j}^{(t)}$ maximal nominal currency value of type t for margin-set j.

4.3.2 *Optimization Variables*

Since the problem seeks a solution for how to allocate securities as collateral in the most cost-efficient way, both counterparty (meaning the specific stock-loan, here denoted margin-set) and which security to choose needs to be considered for a readable output. The primary optimization variable is thereby defined as:

$$x_{i,j} = \text{Number of units of security } i \text{ allocated to margin-set } j.$$

where $x_{i,j} \geq 0$. This variable is bound to be an integer for model 3-4, while it for model 1-2 is continuous. For models 2 and 4, there is an additional factor that has to be taken into account, namely if a transaction has to be made or not. This corresponds to when collateral in a margin-set is changed compared to a previous allocation. This allows the model to consider both a transaction cost and how many transactions that have to be made which allowed the output not only to be readable but also practical and executable. The additional variable introduced is a binary variable, defined as:

$$y_{i,j} = \begin{cases} 1, & \text{if security } i \text{ is relocated to or from margin-set } j, \text{i.e if } |x_{i,j}^0 - x_{i,j}| > 0. \\ 0, & \text{otherwise.} \end{cases}$$

4.3.3 *Objective Functions*

The objective function z is corresponding to *minimize* the costs of all securities in relation to each individual counterparty and margin-set. Here, the cost is related to the value of the position. The objective function is mainly divided into opportunity cost and counterparty risk. The opportunity cost relates to the potential earning of a specific security i if not posted as collateral. The counterparty risk is corresponding to the overexposed value of the posted collateral to margin-set j and the risk of having that extra margin of exposure towards that counterparty. Therefore, each unit of security i placed to margin-set j is multiplied by its market price i and exchange rate i in order to generate the market value (in SEK) which is multiplied by the cost (in base-points). Thereby, the models without a transaction cost (model 1 and 3) has the objective function according to:

$$
\begin{aligned}
z &= \overbrace{\sum_{j \in C} \sum_{i \in S} mp_i FX_i x_{i,j} c_i^o}^{\textit{Opportunity Cost}} + \overbrace{\sum_{j \in C} \sum_{i \in S} (mp_i FX_i x_{i,j} - R_j) c_j^r}^{\textit{Counterparty Risk}} \\
&= \sum_{j \in C} \sum_{i \in S} mp_i FX_i x_{i,j} (c_i^o + c_j^r) - R_j c_j^r
\end{aligned}
\tag{3}
$$

For the remaining models (model 2 and 4), a transaction cost was added for each posted security i to margin-set j. The transaction cost is only initiated if the binary variable $y_{i,j} = 1$ which adds an extra term in the objective function according to the following:

$$
\begin{aligned}
z &= \sum_{j \in C} \sum_{i \in S} mp_i FX_i x_{i,j} (c_i^o + c_j^r) - R_j c_j^r + \overbrace{\sum_{j \in C} \sum_{i \in S} c_{i,j}^t y_{i,j}}^{\textit{Transaction Cost}} \\
&= \sum_{j \in C} \sum_{i \in S} mp_i FX_i x_{i,j} (c_i^o + c_j^r) - R_j c_j^r + c_{i,j}^t y_{i,j}
\end{aligned}
\tag{4}
$$

Although, as previously mentioned, the transaction cost has been set to zero for all transactions in this thesis but the opportunity to add a transaction cost specified for each set $\{i, j\}$ (mainly for margin-set j since most transaction cost cover overhead cost related to the transactions) exists.

4.3.4 *Constraints*

To make sure that all requirements according to the collateral agreements and internal processes are fulfilled, multiple constraints had to be added to the objective functions in order to create sustainable model formulations.

Requirement Fulfillment

The required market value of collateral posted to each margin-set j, can at the lowest have the value of the initial stock-loan R_j with an included initial margin IM_j.

$$
\sum_{i \in S} mp_i FX_i h_{i,j} x_{i,j} \geq (1 + IM_j) R_j, \qquad \forall j \in C. \tag{5}
$$

Inventory

The total amount of posted collateral of security i cannot exceed the available amount in the inventory of security i.[1]

$$
\sum_{j \in C} x_{i,j} \leq I_i, \qquad \forall i \in S. \tag{6}
$$

1 This corresponds to all stock-loan with outstanding collateral and the number of securities available in the bank's possession.

Concentration Limits

The constraints regarding the concentration limits could appear in multiple combinations depending on the contract. Attributes such as type of security, bond-ratings, country of origin, industry sector, index, etc. are stated as parameters in the collateral agreement for specific concentration limits that the counterparty requires. The most common concentration limits (upper-limits) are total share of a single security as collateral, total percentage of a security type or from a specific country. There are four combinations of concentration-type stated in the agreements, namely a percentage market value or collateral value and a currency based market value or collateral value (nominal amount). These different combinations can be formulated as constraints starting with the concentration limit stating that the allocated collateral to margin-set j can at the most take a market value of P_j^t percent of the initial loan, where t specifies the attributes. The same might apply for every single security i if stated in the agreement (which is defined first below).

$$mp_i FX_i x_{i,j} \leq P_{i,j}^{(t)} R_j, \qquad \forall j \in C, \forall i \in S^{(t)}, \forall t \in \{1,2,...,T\}, \tag{7}$$

$$\sum_{i \in S^{(t)}} mp_i FX_i x_{i,j} \leq P_j^{(t)} R_j, \qquad \forall j \in C, \forall t \in \{1,2,...,T\}. \tag{8}$$

If the concentration limit states that the allocated collateral to margin-set j can at the most take a collateral value of Co_j^t percent of the initial loan, a haircut factor per security i and margin-set j has to be taken into account:

$$mp_i FX_i h_{i,j} x_{i,j} \leq Co_{i,j}^{(t)} R_j, \qquad \forall j \in C, \forall i \in S^{(t)}, \forall t \in \{1,2,...,T\}, \tag{9}$$

$$\sum_{i \in S^{(t)}} mp_i FX_i h_{i,j} x_{i,j}, \leq Co_j^{(t)} R_j \qquad \forall j \in C, \forall t \in \{1,2,...,T\}. \tag{10}$$

If the concentration limit states that the allocated collateral to margin-set j can at the most take a market value of Cm_j^t in nominal amount and $Cm_{i,j}^t$ for a single security it defines as:

$$mp_i FX_i x_{i,j} \leq Cm_{i,j}^{(t)}, \qquad \forall j \in C, \forall i \in S^{(t)}, \forall t \in \{1,2,...,T\}, \tag{11}$$

$$\sum_{i \in S^{(t)}} mp_i FX_i x_{i,j} \leq Cm_j^{(t)}, \qquad \forall j \in C, \forall t \in \{1,2,...,T\}, \tag{12}$$

and if a maximum collateral value Cc_j^t is in nominal amount, a haircut factor per security i and margin-set j has to be taken into account, and a nominal amount of $Cc_{i,j}^t$ for a single security:

$$mp_i FX_i h_{i,j} x_{i,j} \leq Cc_{i,j}^{(t)}, \qquad \forall j \in C, \forall i \in S^{(t)}, \forall t \in \{1,2,...,T\}, \tag{13}$$

$$\sum_{i \in S^{(t)}} mp_i FX_i h_{i,j} x_{i,j} \leq Cc_j^{(t)}, \qquad \forall j \in C, \forall t \in \{1,2,...,T\}. \tag{14}$$

4.3.5 *Transaction Constraints*

The constraints introduced so far are valid for all models while the following constraints are added to the models with the binary $y_{i,j}$ in its objective, i.e. model 2 and 4, initiating transaction costs and limit.

Initiate Transaction

To initiate a transaction when a re-allocation has been done, a link has to be set between the variable $y_{i,j}$ and $x_{i,j}$, stating that if there is a difference between the old allocation and the new, a transaction has been made. This link could be initiated by adding the following constraint:

$$|x^0_{i,j} - x_{i,j}| - I_i y_{i,j} \leq 0, \qquad \forall j \in C, \forall i \in S \tag{15}$$

where the difference in allocations cannot succeed the available inventory. These constraints results in that $y_{i,j} = 1$ if the position of $x_{i,j}$ has changed. Furthermore, to be able to solve the problem with this constraint, the absolute value had to be removed and the constraint was divided into two new constraints:[2]

$$\text{-}x_{i,j} - I_i y_{i,j} \leq \text{-}x^{0,}_{i,j} \qquad \forall j \in C, \forall i \in S \tag{16}$$

$$x_{i,j} - I_i y_{i,j} \leq x^{0,}_{i,j} \qquad \forall j \in C, \forall i \in S \tag{17}$$

Transaction Limitation

To make the model and output more realistic and executable, a constraint limiting the number of transactions $y_{i,j}$ is needed. This constraint limits the number of transactions since only a maximum number of transactions are possible in reality. Therefore, the sum $y_{i,j}$ were bounded to be at most a fixed number of transactions M:

$$\sum_{j \in C} \sum_{i \in S} y_{i,j} \leq M. \tag{18}$$

4.4 MODEL DEFINITION

By combining the objective functions and all of the constraints, including the characteristics of the variables, mathematical formulations of the models were created. For example, the complete mathematical model formulation of the optimization problem is defined and formulated as the following for model 4:

2 This particular constraint can be seen in problems defined as *fixed-charge problems* where a similar approach is used to accomplish whenever $x_i > 0$, $y_i = 1$ must hold.[36]

$$\underset{x_{i,j}}{\text{minimize}} \quad z = \sum_{j \in C} \sum_{i \in S} mp_i FX_i x_{i,j} (c_i^o + c_j^r) - R_j c_j^r + c_{i,j}^t y_{i,j}$$

$$\begin{aligned}
\text{subject to} \quad \sum_{i \in S} mp_i FX_i h_{i,j} x_{i,j} &\geq (1 + IM_j) R_j && \forall j \in C, \\
\sum_{j \in C} x_{i,j} &\leq I_i && \forall i \in S, \\
mp_i FX_i x_{i,j} &\leq Pm_{i,j}^{(t)} R_j && \forall j \in C, \forall i \in S^{(t)}, \forall t \in \{1, 2, ..., T\}, \\
mp_i FX_i h_{i,j} x_{i,j} &\leq Pc_{i,j}^{(t)} R_j && \forall j \in C, \forall i \in S^{(t)}, \forall t \in \{1, 2, ..., T\}, \\
mp_i FX_i x_{i,j} &\leq Cm_{i,j}^{(t)} && \forall j \in C, \forall i \in S^{(t)}, \forall t \in \{1, 2, ..., T\}, \\
mp_i FX_i h_{i,j} x_{i,j} &\leq Cc_{i,j}^{(t)} && \forall j \in C, \forall i \in S^{(t)}, \forall t \in \{1, 2, ..., T\}, \\
\sum_{i \in S^{(t)}} mp_i FX_i x_{i,j} &\leq Pm_j^{(t)} R_j && \forall j \in C, \forall t \in \{1, 2, ..., T\}, \\
\sum_{i \in S^{(t)}} mp_i FX_i h_{i,j} x_{i,j} &\leq Pc_j^{(t)} R_j && \forall j \in C, \forall t \in \{1, 2, ..., T\}, \\
\sum_{i \in S^{(t)}} mp_i FX_i x_{i,j} &\leq Cm_j^{(t)} && \forall j \in C, \forall t \in \{1, 2, ..., T\}, \\
\sum_{i \in S^{(t)}} mp_i FX_i h_{i,j} x_{i,j} &\leq Cc_j^{(t)} && \forall j \in C, \forall t \in \{1, 2, ..., T\}, \\
\text{-}x_{i,j} - I_i y_{i,j} &\leq \text{-}x_{i,j}^0 && \forall j \in C, \forall i \in S, \\
x_{i,j} - I_i y_{i,j} &\leq x_{i,j}^0 && \forall j \in C, \forall i \in S, \\
\sum_{j \in C} \sum_{i \in S} y_{i,j} &\leq M && \forall j \in C, \forall i \in S, \\
x_{i,j} &\geq 0 && \forall x_{i,j} \in \mathbb{Z}, \\
y_{i,j} &= 0 \text{ or } 1
\end{aligned}$$

4.5 MODELLING

To be able to solve these models with the provided data, powerful computer software and optimization solvers had to be used. The fundamental programming language used has been Python, version 3.7. Python was chosen since it is an object-oriented programming language that offers a high-level data structure and several third-party modules.

To solve the optimization models, a software engine or interface (library) for Python had to be chosen. There are several options of API:s or Python interfaces available, both commercial and open-source. To ensure replicability and a user-friendly output combined with a modeler that can access several solvers, a free open-source library called PuLP has been used. PuLP is built to leverage the power of the Python language and allow the user to call on third-party solvers in a pythonic way without too much special and specific syntax.[37]

Furthermore, a few solvers were chosen to be used to solve the models. PuLP supports multiple solvers, both commercial and open-source, from different distributors. Through a modular approach, PuLP was used to formulate the models and then expose the data to a solver class. The models were created based on data frames and series using the python data analysis toolkit Pandas as well as matrices and lists in python. The solvers that have been used is, Coin-or Branch and Cut (CBC) which is a free open-source callable library [38] and the Gurobi optimizer solver which has a free students licence.[39]. More than one solver was chosen due to the fact that different solvers can produce different information as output as well as a discussion regarding open-source versus a commercial solver.

The results were obtained using a HP laptop with an Intel Core i5-8350U processor and 8 gb installed RAM-memory.

4.5.1 *General Modelling Approach*

These mentioned solvers use mathematical methods described in section 3.2 above to solve the optimization problems as efficiently as possible. Additionally, the solvers often allow an optimality gap for integer problems which allows the solver to accept a feasible solution as optimal if it is below a given value or a fraction from the best possible solution. The solvers follow an approach according to the flowchart below where most of the steps are allowed to be altered and changed according to what was suitable for the specific type of problem:

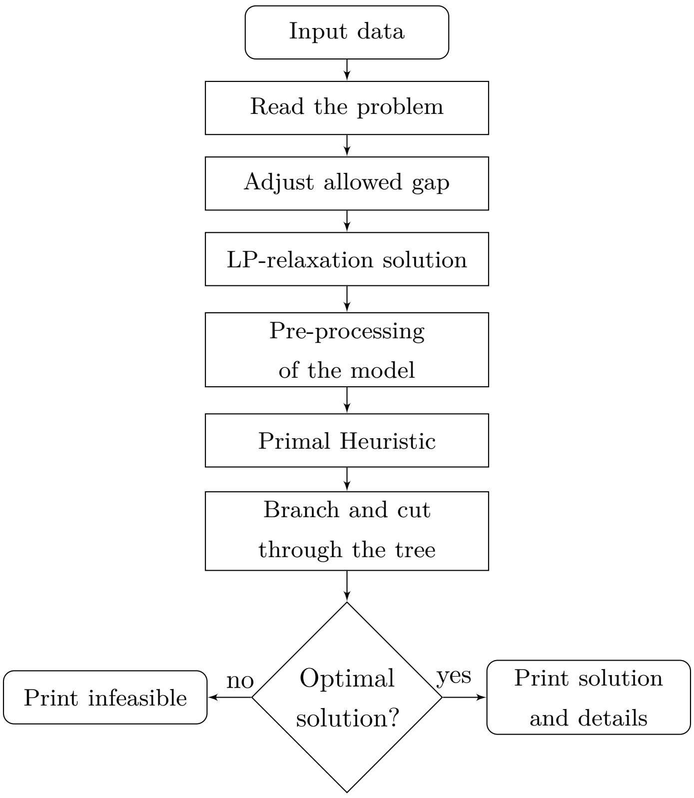

Figure 8: Flowchart of Strategy for Model Building

4.5.2 *Adjusted Modelling Approach*

To be able to compare the results of each solver, an initial solution had to be given. Therefore, the *real* allocation of collateral that was posted at the end of the business day were given as an input. This input had to be tested in the model, with the constraints, to make sure that it truly was a feasible solution. If something differs, for example, if the sum of posted collateral to a counterparty were less than the requirement, this had to be adjusted. From that, an initial feasible cost could be computed that worked as a baseline of comparison with the output of the solvers.

Therefore, in this case, the modelling was slightly adjusted or rather extended. Additionally, from having a feasible input and a result comparison, an important extension for the usefulness was to add error-handling. One important part of this was to provide details of an infeasible solution so that the problem could be adjusted. For example, if the input data has important factors missing which would normally result in infeasibility, the option to automatically change that data-error in the program were provided.

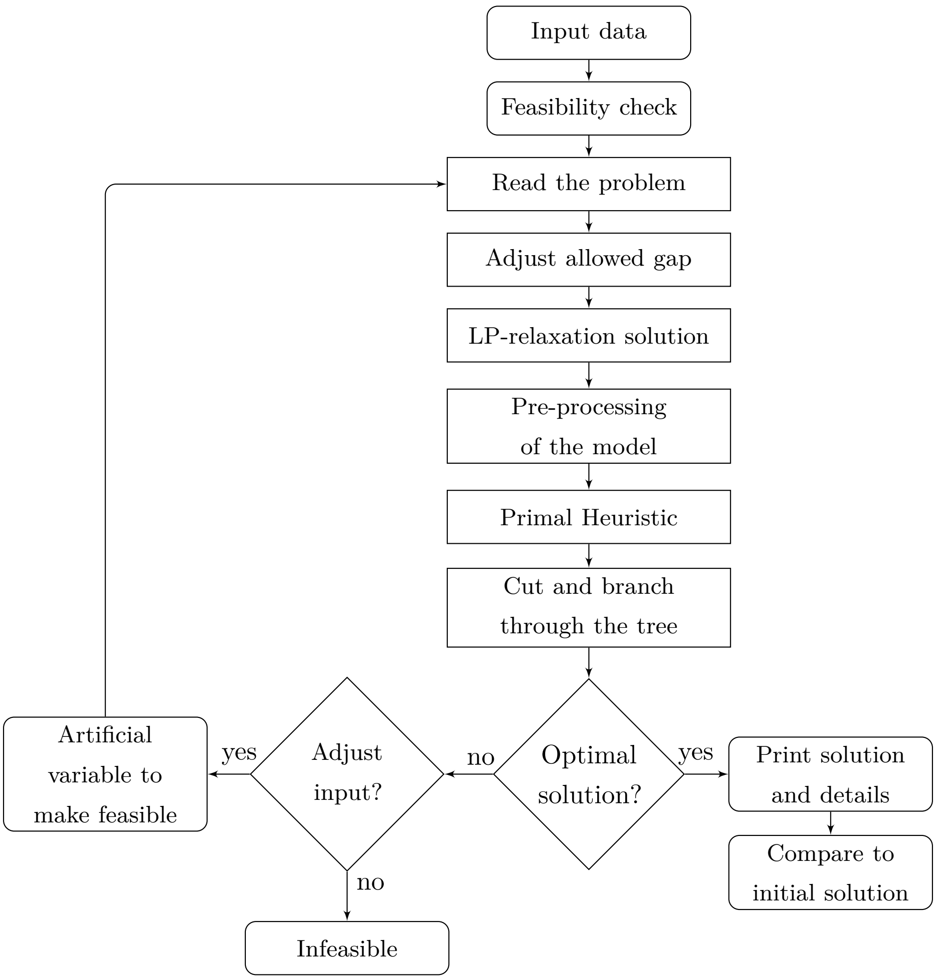

Figure 9: Flowchart of Strategy for Model Building

4.5.3 *Key Factors for Result Evaluation*

Since four different models have been analysed with the purpose of answering the research question on how to efficiently use optimization to determine the collateral allocation, these models had to be evaluated and compared. Efficiency has been considered both in terms of time to solve the problem as well as the reliability of the results. The key factors to measure and compare were, therefore, the average total solution time depending on the size of the problem and the total savings per model compared to the initial feasible solution along with the number of allowed transactions. All these factors have been compared based on the data of a specific day.

In order to choose a solver, all models were compared regarding their time-complexity for each solver. The efficiency level of the solvers could thereby determine which solver to best use in further evaluation of results.

Part III

RESULTS AND DISCUSSION

RESULTS 5

As previously mentioned, the following results are based on input data from one business day. The results presented below show the difference between the initial, previous, posting of collateral during that business day compared to the suggested postings of collateral from the different optimization models in this thesis.

5.1 EVALUATION OF SOLVERS

The result analysis begins by comparing the two different solvers used by evaluating the different models time-complexity. Figure 10 and Figure 11 show how the solution time increases along with the size of variables used in the different models. Here, the size of the problem, i.e. the number of variables used in the models, were randomized from the available margin-set.[1] The total number of variables used, i.e. 100 %, were in the order of $2 \cdot 10^5$. To obtain an acceptable and reliable solution time, both an optimality gap of 1% and a re-sampling of each variable-set were used. To get a reliable result with equivalent prerequisites the maximal number of transactions for model 2 and 4 were set to be unbounded.

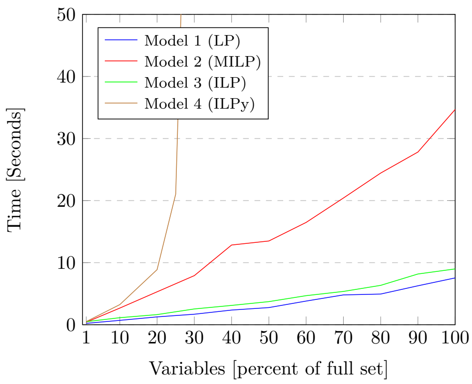

Figure 10: Solver from COIN-OR

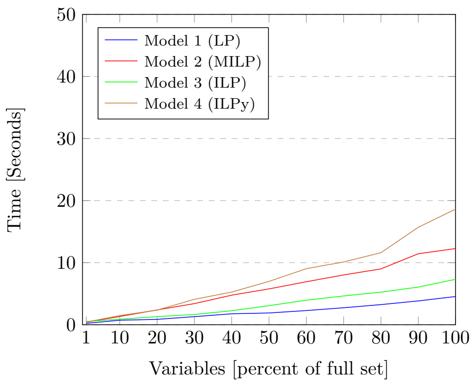

Figure 11: Solver from Gurobi

1 To be able to always get a feasible solution when testing different sizes of variable sets, only the margin-set were sampled since a security that otherwise would have made it feasible could have been falling out of the input data-set.

As expected, the time-complexity becomes larger for the models that are of a more complex nature. As one can see in the figures above, the solving time increases in the following pattern for the models (Model 1 (LP) → Model 3 (ILP) → Model 2 (MILP) → Model 4 (ILP with y)). [2] Despite the fact that each solver could efficiently solve the models, except for model 4 on the Coin-or solver, and that they both revel the same objective value for the full size models, a clear computational advantage could be seen for the Gurobi optimization solver. Since the margin-sets were randomly selected, the objective values for the smaller problems were not worth comparing. The further evaluation of results were thereby chosen to be continued with the usage of the Gurobi solver. A complete table of the different solution times can be found in Appendix A.3.

5.2 RESULTS OF MODELS

When examining the results, the linear models, i.e. model 1 (LP) and model 3 (ILP), will create a lower bound for model 3 (MILP) and model 4 (ILP with transaction) respectively. This means that the lower bound will be the lowest possible objective value which model 2 and model 4 cannot surpass. This since the solution of model 2 and 4 are the same as the solution of model 1 and 3 if the number of transactions are large enough. The results that were chosen to be illustrated below is therefore the examination of model 2 resp. model 4 in relation to model 1 resp. model 3, as well as for the previous collateral postings. The optimality gap allowed in the solvers were set to be a maximum of 1% depending on the problem and solution time.[3] Although, when the optimal solution of model 1 and 3 were computed, the optimality gap were set to zero.

5.2.1 *Model 2 - Mixed Integer Linear Programming*

Four graphs are presented below, representing the trade-off between total cost reduction (objective function) and the maximum number of transactions allowed. Figure 12 shows how the share of total cost differs when the optimization is applied with different number of transactions allowed. The lower bound (green dashed line) is the minimum cost of the LP-model (Model 1) and the upper bound (red dashed line) is the cost of the previous collateral postings which is set to be 100% of the total cost. In

2 In the Coin-or solver, model 4 surpasses more than 1000 seconds after just 30 % of the data-set which is considered large enough (exponential) development to not be documented in the results.

3 To ensure enough data points to the results, the fraction gap was adjusted if the solution required more than 600 seconds. Added gap between 0.1 % and 1 %.

figure 13, the total cost is divided into the different cost components of the objective function (opportunity cost and counterparty risk) and shown as percentage of the total cost. For instance, the opportunity cost accounts for approximately 85% of the initial total cost and the counterparty risk cost accounts for approximately 15% of the initial total cost.

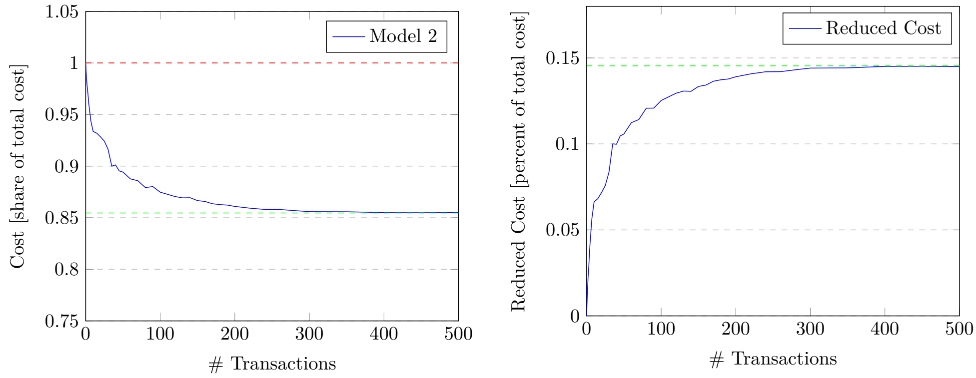

Figure 12: Results illustrating the objective function for model 2. *Left* – The total cost representing the objective function of the model. *Right* – The reduced cost compared to the previous allocations.

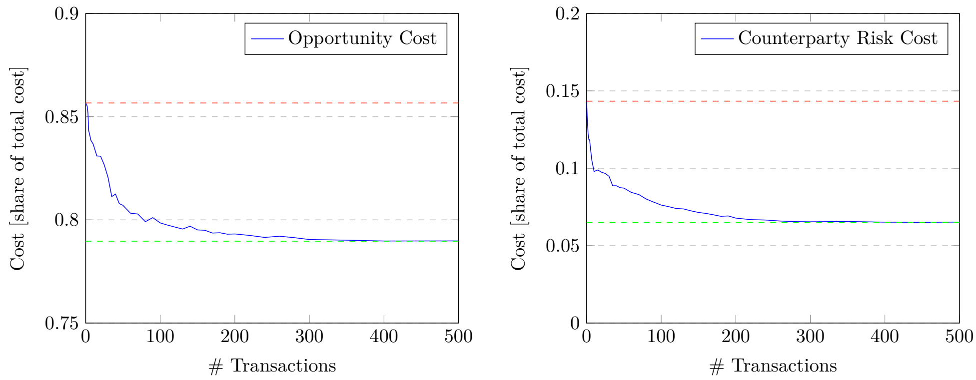

Figure 13: Results illustrating the cost components for model 2. *Left* – Part opportunity cost in the model objective. *Right* – Part counterparty risk cost in the model objective.

The results shows that a potential savings, reduced cost compared to the previous collateral postings the same day, of almost 15% could be reached by using models 1 and 2. When the number of transactions increases, the potential cost reduction converges to the maximal amount of savings, i.e. the lower bound given from the LP-model (15%). More than half of the potential cost reduction are enabled by allowing at least 25 transactions. A large part of the efficient cost reduction, during a few set of transactions, is the fact that the solution is primarily reducing the counterparty risk cost, i.e. the overexposured collateral postings. In figure 13 (right) it is shown that after just a few transactions, the cost related to the counterparty risk is reduced by around 30%. A table of the results can be found in the Appendix A.4.

5.2.2 *Model 4 - Integer Linear Programming with Transactions*

The following results are presented in the same structure as above. Two graphs are presented below representing the trade-off between total cost reduction (objective function) and maximal number of transactions allowed. Figure 14 shows how the share of total cost differs when the optimization is applied with different number of transactions allowed. The lower bound (green dashed line) is the minimum cost of the ILP-model (Model 3) and the upper bound (red dashed line) is the cost of the previous collateral postings and is set to 100% of the total cost. Similar to figure 13 above, figure 15 shows the share the total initial cost for the different cost components (opportunity cost and counterparty risk).

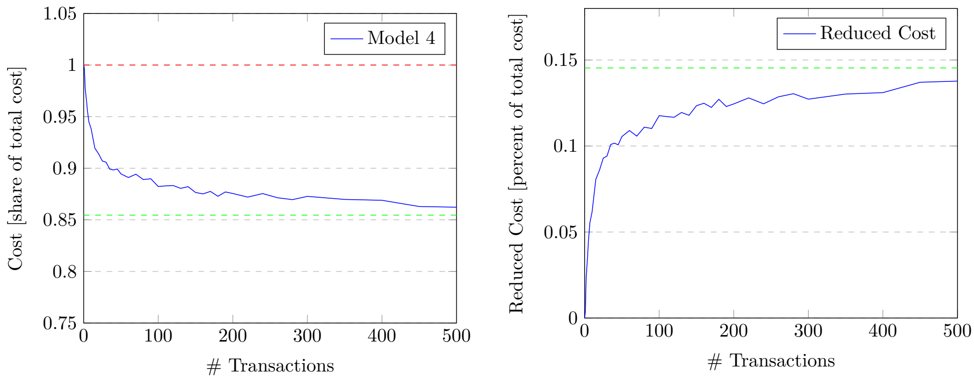

Figure 14: Results illustrating the objective function for model 4. *Left* – The total cost representing the objective function of the model. *Right* – The reduced cost compared to the previous allocations.

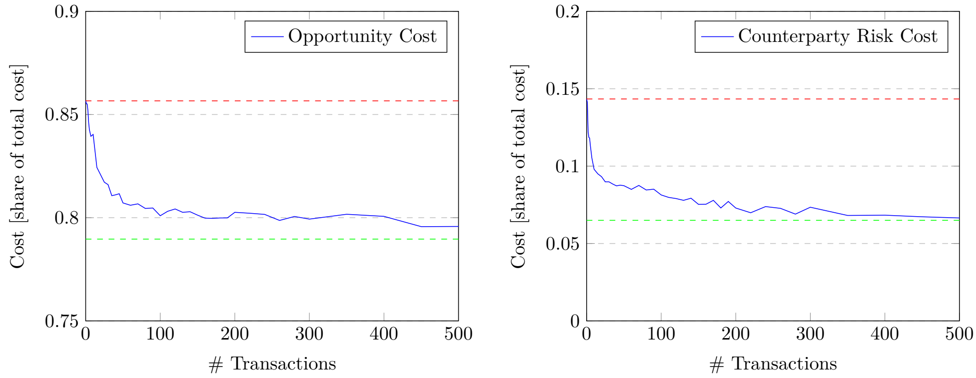

Figure 15: Results illustrating the cost components for model 4. *Left* – Part opportunity cost in the model objective. *Right* – Part counterparty risk cost in the model objective.

Similar to model 1 and 2, model 3 and 4 show that it is possible to reduce the costs of collateral by almost 15%. Compared to model 1, model 3 have a slightly higher minimum due to the integer constraint of the allocation. Moreover, as for model 2, the results of model 4 shows that most of the reduced cost could be realized from a few number of transactions but in this case the overall convergence towards the lower bound is slower and it is still a gap after 500 transactions. A table of the results can be found in Appendix A.4.

DISCUSSION 6

6.1 MODEL COMPLEXITY

As seen in the results presented in section 5.1 the different models have different time complexity. As expected, the more complex the structure of the model, the longer the solution time. The results of the solution time of the LP problem and the LP problem with transaction limits, i.e. model 1 and model 3, have a tendency of increasing linearly depending on the size of the problem. This is a positive finding due to the fact that the increasing rate of the solution time is predictable for larger problems as well.

On the other hand, the time complexity of the two integer problems is not as predictable. This is an expected consequence of the integrality of the models due to the previously mentioned NP-hardness of integer-problems. Although the time complexity seems to grow almost linearly for the fourth model when using the Gurobi solver, one has to bear in mind that each point in the graph is the average of several runs. Despite that it may be a non-predictable development of the solution time when the model is increased in scale, the Gurobi solver does so far seem to handle all types of models and sizes introduced in acceptable running time. Contrarily, the solution time of the CBC solver grows rapidly and already at 30% of the full sized problem, the model takes too long time to further be considered as efficient.

The results of the time-comparison show how the commercial Gurobi solver clearly outperforms the open-source CBC solver. This fact has previously been shown in other studies and have been focusing on purely the time complexity of different solvers. The open source solvers often handle small pure LP-problems well but as the complexity and the size of the problem increases, the commercial solvers outcompete the open source solvers.[40] This fact is unfortunate for companies that want to use mathematical optimization in their operations without heavy investments in an expensive optimization software. The resources and development abilities of large commercial companies with revenues from licences and other products, makes it almost impossible for a non-profit organization as Coin-or to be able to compete. The commercial solvers are being continuously developed and updated to guarantee that the most efficient combination of solution methods are in place while an open-source solver from a scientific foundation, like Coin-or, are depending on grants and partnerships.

The conclusions that can be drawn from the evaluation of these two solvers are that when the size and complexity of the model increases, it might be worth considering an investment in a commercial software for optimization. Although, in an early stage of adapting optimization, and uppermost regarding linear programming as the main problem form, an open source solver could be sufficient.

6.2 DISCUSSION OF RESULTS

When examining the results of section 5.2, it is clear that numerical optimization may have a large impact on the total costs of collateral management. The lower bounds, i.e. the objective value from model 1 and model 3, shows that for the particular day which the model has been tested on, could reduce costs by up to almost 15%. Model 1 shows a slightly higher cost reduction than model 3 but the difference is insignificantly small.

When applying the limitations in the number of transactions, the tradeoff between the maximum number of transactions and the cost reduction is appearing. Meaning that the extra transactions may not be worth considering in due to the increased cost of executing all trades on an administrative level. This is an interesting finding due to the real-life application of the results where a collateral management unit only can execute a certain number of transactions per business day. The fact that a large share of the total potential cost reduction is realized after a few transactions is therefore a positive aspect of the results. One can notice that a large share of the initial cost reduction from a few set of transactions are from a counterparty risk cost reduction. This is an expected finding since a small number of transactions mainly allows the adjustment of over-exposed margin sets.

If one compares the results of model 2 and 4, the similarities are striking. Both graphs behave in a similar way and the main difference is, as previously mentioned, the convergence rate towards the lower bound. This means that the integer-constraint of model 4 results in that more transactions are required to reduce the costs.

6.3 OPTIMALITY CONDITIONS

An important question to answer and discuss when solving an optimization problem is that if one can guarantee that the solution is optimal. Depending on the characteristics of the objective function and the constraints of the model, there are different techniques to verify the optimality of the model. In this thesis, mixed integer linear programming problems and linear programming problems have been solved and the focus will, therefore, be on these types of problems.

Since all results are feasible solutions for each of the models, one can question if the solver has found a global or a local minimum. This is often not intuitive based on the answer of the optimization and additional analysis has to be made. To test this, one method is to look at the convexity of the problem. If the objective function is a convex function and the feasible region obtained from the constraints form a convex set, then any local solution is also a global solution. Since convexity is inherent with a linear objective function and linear constraints, the LP-problem in model 1 is a convex problem. In that case, it could, therefore, be concluded that the solution of model 1 is a global minimum.[26]

For the rest of the models, things get more complicated. The involvement of binary and integer variables makes the remaining problems non-convex since the integrality of the feasible region makes it non-convex according to the definition of convexity. Even though the problem is non-convex, the solution can be guaranteed to be globally optimal if the solution of the LP-relaxation of the initial MILP-problem is also feasible for the MILP-problem. Although, this is most likely not the case but the fundamentals of that theorem are the motivation behind the cutting-plane method.[27]

Both solvers used in this thesis first solves the linear relaxation of the original MILP-problem which provides a lower bound for the problem. By then using the branch-and-cut method, feasible solutions will be computed and infeasible solutions from the LP-relaxations will be cut off, providing a tighter lower limit for what is a possible solution. If an allowed gap is given to the solver, the tree-search will be terminated when a good-enough solution is found. This allowed gap is often given as a percentage of the best possible solution. Even though that the solution is not the same as the result of the relaxation, the solution might be the global optimal solution but in many cases, it could also be a local minimum that is close enough to the objective of the optimal solution.

Additional analyses, and eventually additional methods, could be carried out and used in order to ensure that a global optimal solution is found for every model. The trade-off, in this case, is rather solution time versus the impact of the result. For this thesis, as well as for many other practical situations, a solution that is feasible and within a reasonable gap from the global solution, is preferred over a true global optimum which in worst-case-scenarios requires an unreasonably long time to solve.[26] With this in mind, the results of this thesis are within a 0,1-1% (depending on the problem) optimality gap compared to what might be the global optimum. This is due to the fact that solution times increased severally when a transaction limit was introduced.

To get enough data points, the optimality gap was introduced if a solution took more than 600 seconds.

6.4 IMPLICATIONS - OPERATIONAL OUTPUT

To fill the identified gap in existing literature, one important aim of this thesis was to apply the optimization models on real-life-data and investigate how useful the models output could be. Therefore, it is an interesting discussion on whether or not the output is actionable and how well the model reflects the conditions and challenges that the bank faces in their day-to-day operations. Since the difference in reduced cost between model 2 with output according to $x_{i,j} = \lceil x_{i,j} \rceil$ (meaning that the non-integer solution is rounded up to the nearest integer), and model 4 is insignificantly small, it is highly recommended to use model 2 in the daily operations in due to the efficient solving time and little requirement to post collateral in whole units.

The output of the optimization may be formed into a list of suggesting transactions and how much each transaction will save in terms of basis points. From this, the collateral management unit can implement some of these changes. Although, since the inventory is the sum of the posted collateral, the solution often creates a chain of transactions. This indicates that if one should be able to make one transaction, two others might have to be made to fulfill the requirement constraints. The size of this chain can be reduced with the limitations of the number of transactions but might create a situation where the bank has to do all of the suggested transactions to be able to fulfill every requirement.

Additionally, some of the assumptions in section 4.2 make the models ignore parts of the complexity of the real-life allocations. For instance, in real-life, tri-party contracts differ from normal bilateral agreements which results in that trades cannot be executed with these counterparties in the same way. Moreover, the collateral transactions are often required to be of at least a certain size, an increment which the value of the collateral may not fall short of. This increment amount could, for example, be that collateral can only be transferred if the value exceeds 10 000 SEK. Furthermore, during each business day, multiple input parameters might change value which requires the ability to run the model in an efficient way several times per day. For instance, stock prices, exchange rates and counterparty costs may be subject to changes. Additionally, new margin-sets may be initiated or changed during the day. This requires additional standards regarding how data is managed and how the input data should be structured to avoid data handling problems.

6.5 MODEL IMPROVEMENTS AND SOURCES OF ERROR

The previous discussion leads into the possible model improvements and sources of error for the models of this thesis. In addition to the previously mentioned notes on the tri-party integration and the inclusion of increments, there are a few more possible improvements that might further strengthen the results. The opportunity to add specific transaction costs per counterparty and ISIN exists but has not been used due to lack of data. By adjusting the transaction costs, one can control the attractiveness of making transactions with a certain counterparty. Transaction costs may also be used to prevent trades with tri-party margin-sets.

When it comes to improvements of the performance of the program, additional commercial solvers could have been tested to make sure that the best possible solver was chosen. Furthermore, the program has only been tested on normal PC:s and possible performance enhancing actions, like running on a more powerful computer or accessing more RAM-memory by using a external server, has not been tested. This might increase the solution time and make it possible to get a solution from both solvers.

Additional improvements are mainly related to the output and the linking between other software and data sources. To have a similar model, up and running in the operational activities of the collateral management unit, people with more knowledge in computer science would probably have to be involved. Today, the model is lacking in terms of user interface and the pedagogy of the code is limited, due to specific problems that have been solved during the work with these models.

CONCLUSION 7

7.1 CONCLUSION

The objective of this thesis has been to investigate the research questions stated in section 1.2, and foremost if it is possible for a bank to efficiently optimize its collateral allocation from an mathematical approach.

Based on the results of section 5 it can be concluded that it is possible. The four optimization models that has been tested are all solvable through a commercial solver in an acceptable time. The results are satisfying in terms of the potential cost reduction that could be realized. Furthermore, the dynamics of the results confirm the expected results in terms of how the costs are reduced and how it shows an exponential decrease along with the number of transactions. Additionally, the initial reduced cost can be withdrawn from a primarily reduction in counterparty risk cost. Meaning that the model execute the overexposed collateral positions before relocation collateral. The most efficient model, in terms of both solution time, applicability and reduced cost, is model 2 while model 4 is the most realistic interpretation.

This thesis has proven that it is possible to use operations research to numerically optimize the collateral allocation of a bank and that it could have a substantial impact on the banks day-to-day operations.

7.2 FURTHER RESEARCH

Further work on this subject is highly recommended. A recommendation for future research is to look at the sources of errors presented in this thesis. Here follow some suggestions of central factors that would help further research:

- Allocation including the characteristics of tri-party and agency margin-sets
- Allocation including a minimum increment for collateral postings
- Better and more extensive time-complexity analysis for a higher variety of solver and change in model parameters

- Extend the mathematical approach with different methods and their specifics to compare other solvers
- The ability to test models on a longer time period repeatedly during several business days
- Include the ability to plan for long term positions and thereby taking the time for each positions collateral postings into account
- Sourcing analysis: Create a slack analysis to see which type of securities that gives the most amount of potential savings when rerunning the program and thereby generate suggestions on which assets to source (development of a sensitivity analysis)

Part IV

APPENDIX

APPENDIX A

A.1 SIMPLEX METHOD

The simplex method follows a well-defined algorithm that can be described with the following formulation of the LP problem:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & z = \bar{z} + r_v^T x_v \\ \text{subject to} \quad & A_\beta x_\beta + A_v x_v = b, \\ & x_\beta \geq 0, \quad x_v \geq 0, \end{aligned} \tag{19}$$

where x_β constitutes the basic feasible variables and x_v the non-basic feasible variables, $x \in \{x_\beta, x_v\}$. The $\bar{z}$ and r_v are expressions of the dual variable y (simplex multiplier), thus

$$\bar{z} = c_\beta^T \bar{b} = y^T A_\beta \bar{b} = y^T b, \qquad r_v^T = c_v^T - y^T A_v. \tag{20}$$

In a basic feasible solution, $x_v = 0$, meaning that x_β and $z = \bar{z}$, the following theorem applies.

Theorem *Suppose that $\bar{b} \geq 0$ and $r_v \geq 0$. Then the basic feasible solution x with $x_\beta = \bar{b}$ and $x_v = 0$ is an optimal solution to the linear programming problem (P).*

Algorithm 3 SIMPLEX

INPUT: A complete linear programming formulation of the optimization problem.

OUTPUT: Solution-vector x.

(1) Start with the partition of variables corresponding to β, represented by the basic feasible solution x_β. Its given that $\overline{b} \geq 0$ since x_β is a basic feasible solution. The vectors $\overline{b}, y, r_v$ are calculated according to,

$$A_\beta \overline{b} = b, \quad A_\beta^T y = c_\beta,$$
$$r_v = c_v - A_v^T y.$$

(2) a) **If $r_v \geq 0$:** The algorithm has found that the basic feasible solution x_β ($x_\beta = \overline{b}, x_v = 0$) is an optimal solution to the problem and thereby terminates.

b) **If $r_v < 0$:** Choose q and $\overline{a}_{v_q}$ such that r_{v_q} is the m,ost negative component of r_v and $A_\beta \overline{a}_{v_q} = a_{v_q}$.

(3) a) **If $\overline{a}_{v_q} \leq 0$:** The algorithm has not found an optimal solution to the problem and thereby terminates.

b) **If $\overline{a}_{v_q} > 0$:** Calculate t_{max} and p such that

$$t_{max} = min\{\frac{\overline{b}_i}{\overline{a}_{i,v_q}} : \overline{a}_{i,v_q} > 0\},$$
$$t_{max} = \frac{\overline{b_p}}{\overline{a}_{p,v_q}}, \quad \overline{a}_{p,v_q} > 0.$$

Interchange v_q and β_p in order to update the indexes β and v. Proceed to step (1).

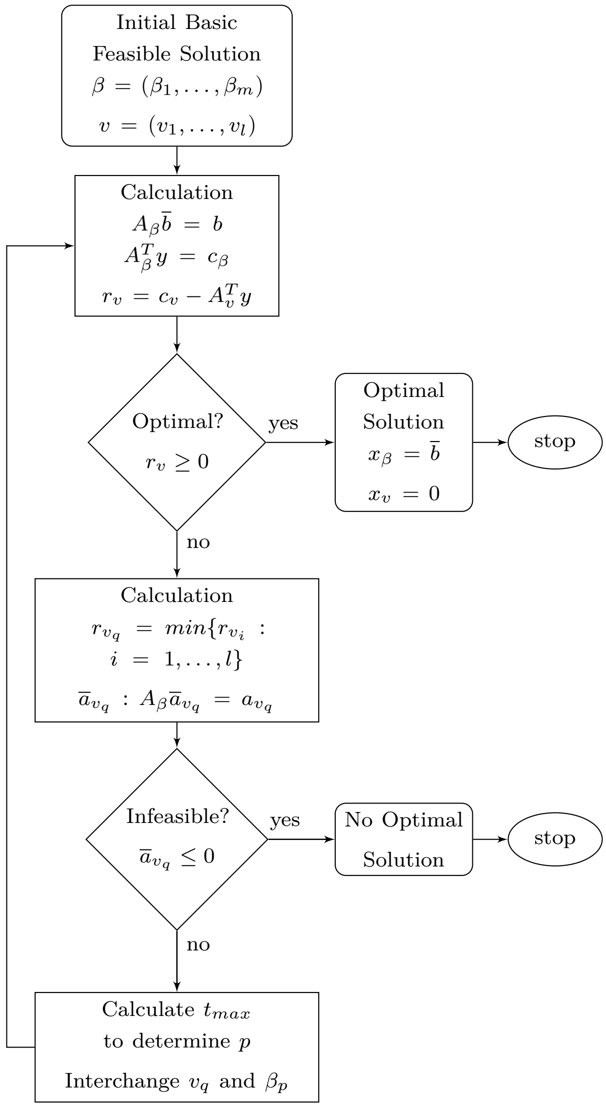

Figure 16: The simplex Algorithm [25]

A.2 BRANCH AND BOUND

The algorithm of the branch and bound method are summarized below:

Algorithm 4 Branch and Bound

INPUT: An ILP or MILP problem P to minimize.

OUTPUT: Optimal LP solution.

(1) Solve LP relaxation of P.
   a) **If** optimal solution is integer-valued:**Stop**.
   b) **Else**: set "best so far"-objective value to infinity, $Z_{Best} = \infty$. Go to step 2.

(2) Let y_j be a non-integer variable solution of the LP relaxation of P. Branch P into two subproblems by adding constraints of the form $y_j \leq \lfloor y_j^R \rfloor$ and $y_j \geq \lceil y_j^R \rceil$. Then solve the new LP relaxations.
   a) **If** subproblem is infeasible: prune that node,
   b) **Else**: go to step 3

(3) Set objective value to Z_{new}.
   a) **If** subproblem has an integer solution: prune that node. Update Z_{Best} if $Z_{new} \leq Z_{Best}$
   b) **Else**:
      i. **If** $Z_{new} > Z_{Best}$: Prune node
      ii. **Else**: Go to step 4

(4) a) **If** all subproblems are pruned: **Stop**; optimal integer solution found with objective Z_{Best}.
   b) **Else**: Go to step 5.

(5) Replace subproblem P with the remaining subproblem with the smallest Z_{new} and go to step 2.

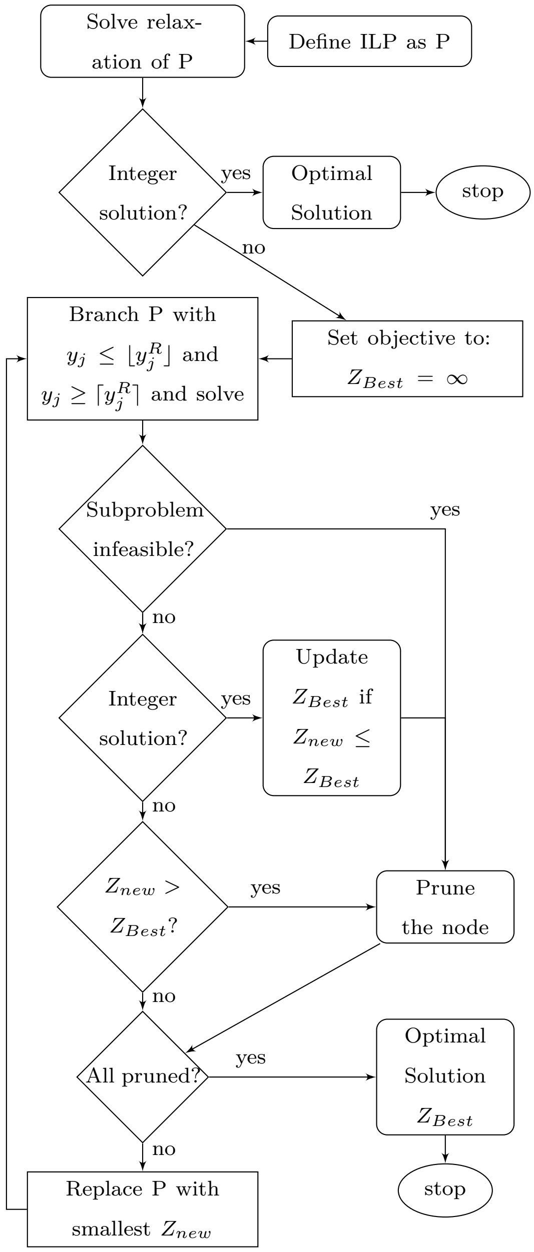

Figure 17: Flowchart of the Branch and Bound Model [27]

A.3 TIME-COMPLEXITY

Gurobi- and Coin-or Solver for all Models

	Solver	**COIN-OR**				**GUROBI**			
Margin-sets	**Model**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**
1		0.22	0.41	0.49	0.51	0.22	0.40	0.52	0.42
14		0.70	2.68	1.13	3.24	0.72	1.33	0.88	1.47
28		1.26	5.30	1.63	8.87	0.86	2.38	1.30	2.35
41		1.69	7.91	2.54	123.71	1.30	3.40	1.65	4.09
55		2.36	12.84	3.11	1000.00<	1.76	4.76	2.25	5.26
69		2.76	13.49	3.72	1000.00<	1.89	5.77	3.07	6.98
83		3.81	16.48	4.68	1000.00<	2.28	6.92	3.95	9.03
97		4.80	20.41	5.36	1000.00<	2.73	8.02	4.63	10.11
110		4.94	24.43	6.34	1000.00<	3.24	8.99	5.25	11.60
124		6.27	27.82	8.17	1000.00<	3.83	11.43	6.06	15.69
137		7.54	34.68	8.99	1000.00<	4.54	12.27	7.31	18.62

Table 5: Solver Statistics [in seconds with tolerance level of 1%]

A.4 OBJECTIVE STATISTICS

Gurobi Solver for Model 2

# Transactions	Opportunity Cost	Counterparty Risk	Objective	Savings
0	0.8566	0.1434	1	0
1	0.856	0.1303	0.9863	0.0137
2	0.8551	0.1222	0.9773	0.0227
3	0.852	0.1186	0.9705	0.0295
4	0.8433	0.1186	0.9618	0.0382
5	0.8418	0.1136	0.9555	0.0445
7	0.8385	0.1051	0.9436	0.0564
10	0.8368	0.0969	0.9337	0.0663
15	0.831	0.1009	0.9319	0.0681
20	0.8309	0.0974	0.9283	0.0717
25	0.8266	0.0977	0.9242	0.0758
30	0.8205	0.0958	0.9164	0.0836
35	0.8093	0.0887	0.898	0.102
40	0.8125	0.0887	0.9012	0.0988
45	0.8079	0.0875	0.8954	0.1046
50	0.807	0.0872	0.8942	0.1058
60	0.8032	0.0844	0.8877	0.1123
70	0.8028	0.083	0.8858	0.1142
80	0.7991	0.0801	0.8792	0.1208
90	0.8031	0.0781	0.8812	0.1188
100	0.7985	0.0762	0.8747	0.1253
110	0.7974	0.0752	0.8726	0.1274
120	0.7965	0.074	0.8705	0.1295
130	0.7955	0.0737	0.8693	0.1307
140	0.7969	0.0725	0.8694	0.1306
150	0.7951	0.0715	0.8666	0.1334
160	0.7949	0.0708	0.8658	0.1342
170	0.7936	0.0699	0.8636	0.1364
180	0.7938	0.069	0.8627	0.1373
190	0.7931	0.0692	0.8623	0.1377
200	0.7932	0.0678	0.861	0.139
220	0.7925	0.0668	0.8593	0.1407
240	0.7915	0.0666	0.858	0.142
260	0.7921	0.0659	0.858	0.142
280	0.7914	0.0655	0.8569	0.1431
300	0.7905	0.0654	0.8559	0.1441
350	0.7902	0.0656	0.8558	0.1442
400	0.7898	0.0652	0.8549	0.1451
450	0.7898	0.0651	0.8548	0.1452
500	0.7898	0.0652	0.855	0.145

Table 6: Solver Statistics for Model 2 using Gurobi

Gurobi Solver for Model 4

# Transactions	Opportunity Cost	Counterparty Risk	Objective	Savings
0	0.8566	0.1434	1	0
1	0.8552	0.1413	0.9965	0.0035
2	0.8551	0.1222	0.9773	0.0227
3	0.852	0.1186	0.9705	0.0295
4	0.846	0.1182	0.9642	0.0358
5	0.8426	0.1133	0.9559	0.0441
7	0.8395	0.1053	0.9448	0.0552
10	0.8403	0.098	0.9383	0.0617
15	0.8244	0.095	0.9194	0.0806
20	0.8208	0.0932	0.914	0.086
25	0.8172	0.0898	0.907	0.093
30	0.816	0.0898	0.9059	0.0941
35	0.8107	0.0884	0.8991	0.1009
40	0.8111	0.0872	0.8983	0.1017
45	0.8116	0.0877	0.8993	0.1007
50	0.8072	0.0874	0.8945	0.1055
60	0.806	0.085	0.891	0.109
70	0.8067	0.0875	0.8942	0.1058
80	0.8045	0.0846	0.889	0.111
90	0.8047	0.0851	0.8898	0.1102
100	0.8009	0.0814	0.8823	0.1177
110	0.8031	0.0798	0.8829	0.1171
120	0.8042	0.079	0.8833	0.1167
130	0.8026	0.0779	0.8805	0.1195
140	0.8029	0.0793	0.8821	0.1179
150	0.8012	0.0753	0.8765	0.1235
160	0.7998	0.0753	0.8751	0.1249
170	0.7997	0.0779	0.8775	0.1225
180	0.7999	0.073	0.8728	0.1272
190	0.7999	0.0771	0.877	0.123
200	0.8026	0.0729	0.8755	0.1245
220	0.8022	0.0699	0.872	0.128
240	0.8016	0.0739	0.8755	0.1245
260	0.7987	0.0727	0.8714	0.1286
280	0.8006	0.069	0.8696	0.1304
300	0.7993	0.0734	0.8727	0.1273
350	0.8017	0.0681	0.8698	0.1302
400	0.8007	0.0683	0.8689	0.1311
450	0.7957	0.0672	0.8629	0.1371
500	0.7958	0.0665	0.8622	0.1378

Table 7: Solver Statistics for Model 4 using Gurobi

[1] Bank of New York Mellon & The Field Effect. *Collateral Management: A Review of Market Issues.* [Online; accessed 15-feb-2019]. 2015. URL: https://www.bnymellon.com/_global-assets/pdf/our-thinking/collateral-management-a-review-of-market-challenges.pdf.

[2] John K, Lynch A.W, and Puri M. "Credit Ratings, Collateral, and Loan Characteristics: Implications for Yield." In: *Journal of Business* 76.3 (2003), pp. 371–409.

[3] Euroclear. *Understanding Repos and The Repo Markets.* [Online; accessed 12-jan-2019]. 2009. URL: https://www.theotcspace.com/sites/default/files/2011/11/003-the-repo-market.pdf.

[4] Fender I and Lewrick U. "Mind the gap? Sources and implications of supply-demand imbalances in collateral asset markets." In: *BIS Quarterly Review* (2013).

[5] Rule G and Bank of England. *Collateral Management in Central Bank Policy Operations.* [Online; accessed 20-jan-2019]. 2012. URL: https://www.bankofengland.co.uk/-/media/boe/files/ccbs/resources/collateral-management-in-central-bank-policy-operations.

[6] Brunnermeier M.K. "Deciphering the liquidity and credit crunch 2007-2008." In: *Journal of Economic Perspectives—Volume 23* 1.1 (2009), pp. 77–100.

[7] Greenspan A. "The Crisis." In: *Brookings Papers on Economic Activity* 41.1 (Spring) (2010), pp. 201–261.

[8] Lorenzoni G and Walentin K. "Financial frictions, investment and Tobin's q." In: *Sveriges Riksbank Working Paper Series* 208 (2007).

[9] Rampini A.A and Viswanathan S. "Collateral, risk management, and the distribution of debt capacity." In: *The Journal of Finance* 65.6 (2010), pp. 2293–2322.

[10] Rampini A.A and Viswanathan S. "Collateral and capital structure." In: *Journal of Financial Economics* 109.2 (2013), pp. 466–492.

[11] Duarte F.D, Gama A.P.M, and Gulamhussen M.A. "Defaults in bank loans to SMEs during the financial crisis." In: *Small Business Economics* 51.3 (2018), pp. 591–608.

[12] Cipovová E and Jaroslav B. "Impacts of selected methods of credit risk management on bank performance." In: *Proceedings of the 8th European Conference on Management, Leadership and Governance* 30 (2012).

[13] Singh M. "The Changing Collateral Space." In: *IMF Working Paper* (2013). URL: https://www.imf.org/en/Publications/WP/Issues/2016/12/31/The-Changing-Collateral-Space-40280.

[14] SUNGARD. *Collateral Optimization – How it really works*. White Paper. [Online; accessed 18-December-2018]. 2013. URL: http://finance.flemingeurope.com/webdata/4201/WP_CollateralOptimization_HowItReallyWorks_2013.pdf.

[15] Schiebe T et al. *Techniques for Post-Trade Collateral Optimization*. [Online; accessed 7-jan-2019]. 2016. URL: https://www.fisglobal.com/solutions/institutional-and-wholesale/asset-management/-/media/fisglobal/files/report/techniques-for-post-trade-collateral-optimization.pdf.

[16] Schiebe T et al. *Collateral Optimization: The Next Generation of Collateral Management*. [Online; accessed 22-jan-2019]. 2015. URL: https://www.sapient.com/content/dam/sapient/sapientglobalmarkets/pdf/thought-leadership/Collateral_Optimization_Whitepaper.pdf.

[17] Pricewaterhousecoopers. *First Take: Basel's Final NSFR*. [Online; accessed 10-jan-2019]. 2014. URL: https://www.pwc.com/us/en/financial-services/regulatory-services/publications/assets/net-stable-funding-ratio-basel-iii.pdf.

[18] 4sight and Rule Financial. *Whitepaper - Buyside Collateral Management: Challenges and Opportunities*. [Online; accessed 29-April-2019]. 2014. URL: https://blog.gft.com/wp-content/uploads/2015/03/Whitepaper1.pdf.

[19] Bylund J. *Collateral Optimization*. Master Thesis. Umeå University, Department of Physics. 2017.

[20] PG Reynolds. "Computational Studies of Collateral Optimisation Problems." PhD thesis. North-West University, 2018.

[21] Eber P et al. *Collateral Management*. Deloitte. [Online; accessed 13-mars-2019]. URL: https://www2.deloitte.com/content/dam/Deloitte/lu/Documents/risk/lu-collateral-management.pdf.

[22] Baker H.K and Filbeck G. *Investment Risk Management*. First Edition. Oxford University Press, 2014. ISBN: 978-0-1993-3196-3.

[23] White L.J. "The Credit Rating Agencies and Their Role in the Financial System." In: *Forthcoming in E. Brousseau* (2018).

[24] Kissell R. “The Expanded Implementation Shortfall: Understanding Transaction Cost Components.” In: *The Journal of Trading* 1.3 (2006), pp. 6–16.

[25] Sasane A and Svanberg K. *Optimization (1811/1831/1841)*. First Edition. Department of Mathematics, Royal Institute of Technology, Stockholm, 2016.

[26] Griva I, Nash S.G, and Sofer A. *Linear and Nonlinear Optimization*. Second Edition. Society for Industrial and Applied Mathematics, 2009. ISBN: 978-0-8987-1661-0.

[27] Ravindran A.R. *Operations Research and Management Science Handbook*. First Edition. CRC Press, 2007. ISBN: 978-0-8493-9721-9.

[28] Nemhauser G and Wolsey L. *Integer and Combinatorial Optimization*. Firth Edition. John Wiley & Sons, Inc., 1999. ISBN: 9780471828198.

[29] Cornuéjols G. “Revival of the Gomory cuts in the 1990’s.” In: *Annals of Operations Research* 149.1 (2007), pp. 63–66.

[30] Marchand H et al. “Cutting planes in integer and mixed integer programming.” In: *Discrete Applied Mathematics* 123.1-3 (2002), pp. 397–446.

[31] Berthold T. “Primal Heuristics for Mixed Integer Programs.” In: (2006).

[32] Fischetti M, Glover F, and Lodi A. “The Feasibility Pump.” In: *Mathematical Programming* 104.1 (2005), pp. 91–104.

[33] L Caccetta and SP Hill. “Branch and cut methods for network optimization.” In: *Mathematical and computer modelling* 33.4-5 (2001), pp. 517–532.

[34] Pachkova E.V. “Duality in MIP by branch-and-cut: generating dual price functions using branch-and-cut.” In: *Advanced Modeling and Optimization* 7 (2005).

[35] Guzelsoy M and Ralphs T.K. “Integer Programming Duality.” In: 7 (2010), pp. 1–13.

[36] Winston W.L. *Operations Research: Applications and Algorithms*. Fourth Edition. Brooks/Cole, a division of Thomson Learning, Inc., 2004. ISBN: 978-0-5343-8058-8.

[37] Mitchell S, O’Sullivan M, and Dunning I. “PuLP: A Linear Programming Toolkit for Python.” In: *Department of Engineering Science* (2011).

[38] Lougee-Heimer R. “The Common Optimization Interface for Operations Research: Promoting open-source software in the operations research community.” In: *IBM Journal of Research and Development* 47.1 (2003), pp. 57–66.

[39] IBM ILOG CPLEX. “V12. 1: User’s Manual for CPLEX.” In: *International Business Machines Corporation* 46.53 (2009), p. 157.

[40] Meindl B and Templ M. "Analysis of commercial and free and open source solvers for linear optimization problems." In: *Eurostat and Statistics Netherlands within the project ESSnet on common tools and harmonised methodology for SDC in the ESS* 20 (2012).

TRITA -SCI-GRU 2019:105