# Optimization of Order Quantities in the Presence of Quantity Discounts

**Mikko Närhi**

**School of Science**

Bachelor's thesis
Espoo 8.10.2021

**Supervisor**

Prof. Ahti Salo

**Advisor**

D.Sc. (Tech.) Markku Verkama

**Aalto University**
**School of Science**

**Aalto University**
**School of Science**

**Aalto University, P.O. BOX 11000, 00076 AALTO**
**www.aalto.fi**
**Abstract of the bachelor's thesis**

**Author** Mikko Närhi

**Title** Optimization of Order Quantities in the Presence of Quantity Discounts

**Degree programme** Engineering Physics and Mathematics

**Major** Mathematics and Systems Sciences **Code of major** SCI3029

**Teacher in charge** Prof. Ahti Salo

**Advisor** D.Sc. (Tech.) Markku Verkama

**Date** 8.10.2021 **Number of pages** 21 **Language** English

**Abstract**

Commonly the price of a purchase item depends on the purchase quantity. This is especially the case when the purchase items are manufactured to order, in which case fixed manufacturing costs are shared by a bigger quantity. On the other hand, as the order quantity increases, so does the warehouse cost. Optimizing order quantities is therefore an important part of the purchasing process. In the purchasing process, it is beneficial to consider multiple alternatives for the possible amount to be ordered.

The objective of this thesis is to find the minimum annual total cost function and the corresponding optimal order quantity for every purchase item under investigation. Towards this end, a mixed-integer linear programming model was formulated to find the optimal order quantities for each purchase item. A Python program was used to implement the optimization model and calculate the optimal order quantities.

The case company of this thesis is Normet, a Finnish company that specialises in products for tunnel construction and mining. All data used in this thesis was acquired from Normet's ERP-system.

The results of the model were consistent and the general recommendation for the purchase items was to order in larger quantities. This is because the benefit of lower prices usually outweigh the drawbacks of warehouse costs and capital costs. By increasing the order quantities of purchase items with price breaks, we can make optimal procurement decisions. Therefore, the model can help the company achieve significant savings. Inclusion of risk analysis and the removal of several assumptions are identified as topics for future research.

**Keywords** Quantity Optimization, All-Units Quantity Discount, Mixed-Integer Linear Programming, Multi-Product, Inventory Management

Aalto University
School of Science

Aalto-yliopisto, PL 11000, 00076 AALTO
www.aalto.fi
**Tekniikan kandidaatintyön tiivistelmä**

| | |
|---|---|
| **Tekijä** Mikko Närhi | |
| **Työn nimi** Tilauserän koon optimointi huomioiden hintaportaat | |
| **Koulutusohjelma** Teknillinen fysiikka ja matematiikka | |
| **Pääaine** Matematiikka ja systeemitieteet | **Pääaineen koodi** SCI3029 |
| **Vastuuopettaja** Prof. Ahti Salo | |
| **Työn ohjaaja** TkT Markku Verkama | |
| **Päivämäärä** 8.10.2021 **Sivumäärä** 21 | **Kieli** Englanti |

**Tiivistelmä**

Tuotteen hinta riippuu tilauserien koosta. Tämä pätee erityisesti, kun ostonimikkeet valmistetaan tilausta varten, jolloin valmistuskustannukset jakautuvat suuremmalle määrälle. Toisaalta kun tilausmäärä kasvaa, niin varastointikustannukset kasvavat. Tilausmäärien optimointi on siis tärkeä osa hankintaprosessia. Hankintaprosessin aikana on hyödyllistä pohtia useita eri vaihtoehtoja lopulliselle tilausmäärälle. Tämän kandidaatintyön tavoite on minimoida vuotuinen kokonaiskustannusfunktio ja määrittää sitä vastaava optimaalinen tilausmäärä jokaiselle ostonimikkeelle.

Ensimmäinen tehtävä oli muodostaa sekakokonaislukuoptimointimalli, jota käytetään jokaisen ostonimikkeen optimaalisen tilausmäärän löytämiseen. Tämän jälkeen työssä kehitettiin Python-ohjelma, jolle optimointimalli implementoitiin.

Tämän kandidaatintyön esimerkkitapaus on Normet, joka on suomalainen louhinta- ja kaivoskoneiden valmistukseen erikoistunut yritys. Kaikki aineisto, jota kandidaatintyössä käytetään, on saatu Normetin ERP-järjestelmästä.

Optimointimallin ja sen pohjalta kehitetyn Python -ohjelman tulokset olivat johdonmukaisia. Yleinen trendi tuotteiden optimaaliselle tilausmäärälle oli, että optimointimalli suosii suuria tilausmääriä. Tämä johtuu siitä, että hyöty suurten tilausmäärien halvemmista yksikkökustannuksista on usein mittava. Hintaportaita sisältävien ostonimikkeiden tilausmäärää kasvattamalla saadaan tehtyä optimaalisia hankintapäätöksiä. Tämän vuoksi työssä kehitetty optimointimalli pystyy auttamaan yrityksiä saavuttamaan merkittäviä säästöjä. Työssä ehdotetaan lisätutkimusta, joka tarkastelee riskejä ja mahdollisuuksia rajaavien oletusten poistamiseen.

**Avainsanat** Tilauserän koon optimointi, paljousalennus, useat tuotteet, varastohallinta

# Contents

# 1 Introduction

Usually the price of a purchase item depends, among other things, on the amount ordered. When a buyer purchases a greater number of items at once, the cost per unit for these items decreases, as can be seen from Figure 1. The discounts for different order quantities are called price breaks, and they are defined in price negotiations between the buyer and the supplier (seller). A price break schedule defines the unit price of a purchase item for every possible order quantity. If an order quantity falls into a certain price break interval, the corresponding unit price is chosen. Price break intervals are the order quantities that define the lower and upper boundaries where you can buy a purchase item for a specific unit price.

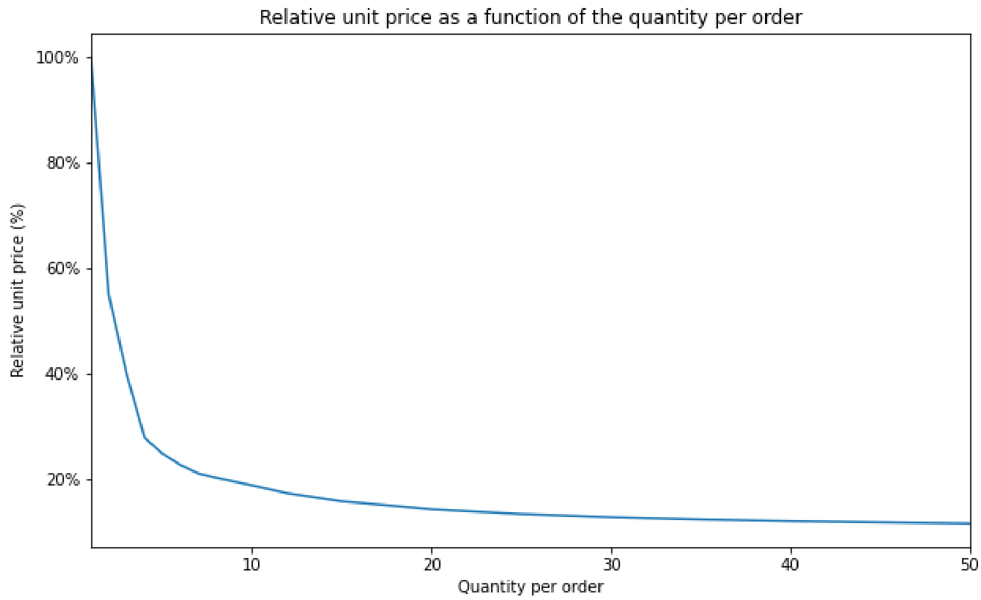


Figure 1: An example of price breaks for a purchase item.

Figure 1 is an example of how price breaks can behave for purchase items. The decline of the relative unit price is steep, the unit price for ten items per order is approximately 20% of the unit price for one item per order.

Price break opportunities are available to consumers as well. An example of a price break in business-to-consumer (B2C) transactions is the purchase of a cell phone plan for a fixed term. This means that the price per month (unit price) decreases when the fixed term is longer. The benefits of price breaks in business-to-business (B2B) transactions are magnified since companies require the same raw materials continuously and in greater quantities (Munson and Jackson, 2015).

Inventory management, when done properly, can lower the costs and increase the profits of a company significantly. The main challenge in this thesis is that each purchase item has unique price break intervals and because prices correspond to the intervals, it is necessary to find the optimal order quantity for each of the purchase items separately. In larger order quantities, fixed manufacturing costs are shared by a bigger quantity. However, as order quantity increases, so do warehouse costs.

The objective of this thesis is to find the minimum annual total cost function and the corresponding optimal order quantity for every purchase item under investigation. The proposed model is used on data provided by Normet, a Finnish company that specialises in making products for tunnel construction and mining. The paper presents computational results from 11 test instances with up to 100 purchase items and possible order quantities ranging from three to ten. The results were evaluated by comparing the total purchasing costs for the newly optimized order quantities with the reference total purchasing costs calculated from historical data. Lastly, conclusions are made and avenues for potential future research are outlined.

# 2 Literature review

The question "How much to order?" has been studied widely. Benton and Park (1996) review the literature on determining the lot size under price breaks and classifies the literature to a subcategory. The literature is first categorized to non time-phased demand and time-phased demand and then into all-units discounts and incremental discounts. Lastly, all papers are categorized to inspect either the buyer's perspective or buyer-supplier perspective. In this thesis, focus is on the literature that deals with non-time phased demand and all-units discounts from the buyer's perspective.

Rubin and Benton (1993) investigate a situation where the buyer is offered discounted price schedules from multiple suppliers, and the buyer has to make the decision under numerous constraints, such as limited storage space and restricted inventory budgets. The model in Rubin and Benton (1993) consists of primal and dual problems and solves these problems with the relaxation and Branch-and-Bound algorithms. The primal problem minimizes the sum of all annual total cost functions for all items. The total annual cost function is the sum of purchase, order placement and warehouse costs. The dual problem is formulated on a Lagrangian function that assigned a Lagrange multiplier to each constraint for each item, so that the augmented cost function is the total annual cost function from the primal problem and the Lagrange multipliers for each constraint. The model was estimated with data on 10 items with three price break intervals. This model deals with constraints and multiple possible suppliers for each item efficiently. It incorporates limited storage capacity and annual budgetary constraint and finds the optimal solution for every instance.

Moussourakis and Haksever (2008) present a zero-one mixed integer programming model to minimize the total cost function in multi-product multi-constraint inventory systems subject to the assumption of all-units price breaks. The model included a piecewise linearization of the number of orders function for each quantity interval and binary variable implementation that makes it possible to choose either an independent or a common cycle approach. The author computationally tested the model and formulated randomly generated data for the performance tests. They conclude that the model provides accurate optimal solutions and better results (up to 3%) than previous models. They found neither the independent nor the common cycle approach to be the optimal approach.

Goossens et al. (2007) give a mathematical formulation for the cost-minimization problem as the total price break problem. This formulation is a zero-one mixed integer programming model that minimizes the total costs for each purchase item. Each purchase item can be purchased from a number of suppliers that offer an arbitrary amount of quantity intervals. In addition, they present four variants of the basic form, each addressing different settings for an optimization problem. In the first variant, the market share for suppliers is constrained. In the second variant, the buyer is allowed to purchase larger quantities of purchase items than what is indicated by the demand, while in the third variant the number of suppliers to buy from is constrained. The fourth variant presents a multi-period model to answer the question: "When to order what goods?". The models were tested on randomly generated data. The author also describe three exact algorithms and conclude that each of these algorithms finds an exact solution in a reasonable amount of time. Each algorithm is best in different instances. For example, a linear programming based branch-and-bound algorithm works best in instances where the buyer is allowed to purchase more than is needed according to the demand and when the randomly generated dataset is large.

Jackson and Munson (2016) investigate the possibility of storage capacity expansion to fully utilize price breaks for multiple purchase items. If companies order multiple products that have price breaks, increasing storage capacity can help them take full advantage of price breaks and achieve savings. The author develop a model that minimizes the sum of total cost functions for all items simultaneously. This allows the model to include resource constraints. The model solves the capacity level and order quantity for each item. The model finds efficient solutions for multi-product lot-sizing problems that incorporate all-units and/or incremental price breaks and consider resource capacity as a decision variable. This study is the first to introduce resource capacity as a decision variable. The proposed algorithms can also solve resource capacity levels and order quantities for thousands of items in real time. Furthermore, the authors find that storage capacity expansion can lead to significant potential savings and that the expansion is usually profitable when suppliers offer all-units price breaks.

Lee et al. (2013) introduce a mixed integer programming model (MIP) to solve a lot-sizing problem with multiple suppliers, multiple periods and price breaks. The objective is to minimize the total cost function that is constructed of ordering costs, warehouse costs, purchase costs and transportation costs. The model takes the demand for each planning period as an input and finds the optimal lot-sizing quantities to fulfil the demand for each period.A genetic Algorithm is also introduced to solve the problem more efficiently. The results show that both the MIP and the Genetic Algorithm can find optimal or near optimal solutions to the problem in a relatively short amount of time but when the problem becomes NP-hard, only the Genetic Algorithm can solve the problem in a reasonable amount of time.

Zhang and Chen (2013) study a single period, single item procurement problem with stochastic demand and multiple suppliers. They present a nonlinear mixed integer programming model to minimize the total cost function consisting of purchase costs, supplier selection costs and holding-shortage costs. Stochastic demand is tackled by creating different test instances with different means, variances, probability density functions and cumulative functions. The results show that a larger variance in demand increases procurement costs. Additionally, the efficiency of the algorithm is tested on 100 randomly generated problems with a demand that follows a normal distribution. The algorithm solves these problems efficiently.

The minimization of a total cost function for purchase items, mixed-integer linear programming, all-units price breaks and the utilization of piecewise linear functions were relevant for the research of this thesis. Stochastic demand was not relevant because the demands for all purchase items have been received from Normet's demand forecasts. Also, incorporating multiple demand periods for the proposed model was unnecessary because the demand for each purchase item is given as an annual forecast.

# 3 Research data and model formulation

## 3.1 Research data

The research material for this thesis consisted of data from Normet's ERP system. The data consisted of four separate data files.

1. The price break intervals for all purchase items.
2. The annual demand of every purchase item.
3. The weight of every purchase item.
4. Historical purchase order data from the last 12 months.

The purchase order data was used to assess the performance of the proposed model by comparing the optimized annual total cost function value of each purchase item with the historical average annual total cost function value.

Preprocessing every data file was necessary, since each data file included relevant information for all purchase items, not just the purchase items with price breaks. Here the challenge was to find the purchase items with significant price breaks. Together, all data files consisted of 598 850 rows.

All data was originally received in Excel format and the data files were imported for data analysis and manipulation. Data analysis and manipulation were done with Python and run on Visual Studio Code integrated development environment.

## 3.2 Assumptions

The proposed model in this thesis is subject to the following assumptions:

1. Demand for all purchase items was taken from demand forecast provided by Normet, making it deterministic.
2. Purchase items have independent price breaks.
3. Suppliers confirm orders with any quantity in the given price break range.
4. The demand for each purchase item is an independent variable.
5. The time horizon is 12 months.
6. Warehouse costs are directly proportional to the volume of the purchase item.
7. There are no constraints in warehouse capacity.
8. Each purchase item has an independent cycle length. In an independent cycle, the order cycle for each item is different.
9. All purchase items have all-units price breaks. This means that the price break is applied for every purchase item purchased and all items are purchased for the same unit price.
10. Ordering cost per order for all purchase items is a fixed amount and it is independent of order size.
11. For a time horizon, there exists only one optimal order quantity and a corresponding unit price for each purchase item.

The assumptions of the proposed model are consistent with prior literature. Most assumptions are the same as in the models of Lee et al. (2013) and Jackson and Munson (2016) with modifications to fit the model of the thesis. All assumptions were made in cooperation with Normet.

## 3.3 Prerequisites

There was no precise data that contained the product volumes of all purchase items investigated in this thesis. Nevertheless, Normet's data management system, Sovelia included the height, length and width for some purchase items but they had to be investigated one by one manually making it virtually impossible to retrieve the exact product volumes of each purchase item in question. The data for the weights of all investigated purchase items was available. Plus, it was reasonable to assume that the volumes and weights of all purchase items are directly proportional. Then, the relation of the weight and volume of purchase item $j$ is

$$V_j = mw_j, \tag{1}$$

where $V_j$ is the volume of item $j$, $w_j$ is the weight of item $j$ and $m$ is a positive non-zero proportionality constant.

It was known that this assumption does not apply to purchase items such as fuel tanks. Nevertheless, it was also known that these purchase items do not contain any price breaks, making the assumption acceptable for the set of purchase items under investigation.

Only one value can be chosen to be the optimal order quantity and there are usually several price break intervals to choose from. That is why, in this thesis, we introduce a binary decision variable that is defined for purchase item $j$ from supplier $k$ in price break interval $i$ by

$$y_{ijk} = \begin{cases} 1 & l_{ijk} \leq x_j \leq u_{ijk} \\ 0 & \text{otherwise,} \end{cases}$$

where $y_{ijk}$ is a binary decision variable, $l_{ijk}$ and $u_{ijk}$ are the quantities that define the lower and upper boundary of each price break interval and $x_j$ is the optimal order quantity for purchase item $j$.


Figure 2: Price break schedule from two vendors for a purchase item $j$.

Figure 2 shows an example of the behavior of price breaks from two different vendors for one purchase item. When a price break interval ends, a lower unit price is chosen for the purchase item. For example, if we were to order two pieces of purchase item $j$ from vendor 1, the unit price would be three. The objective of this thesis is to optimize order quantities. In Figure 2, it is possible to acquire purchase items from two different vendors, however, it is not always feasible to consider all order quantities from all vendors for the optimal order quantity $x_j$. In the model of this thesis, we only consider the lowest unit price of every possible order quantity in the given price break range for the optimal order quantity.

In previous literature, Moussourakis and Haksever (2008) use a piecewise-linear function to approximate the nonlinear number of orders function. In their model, order quantities can have integer and fractional values and the objective is to minimize the error associated with the linear approximation. In our model, the linear approximation can have integer values for order quantities of any quantity in the given price break range.

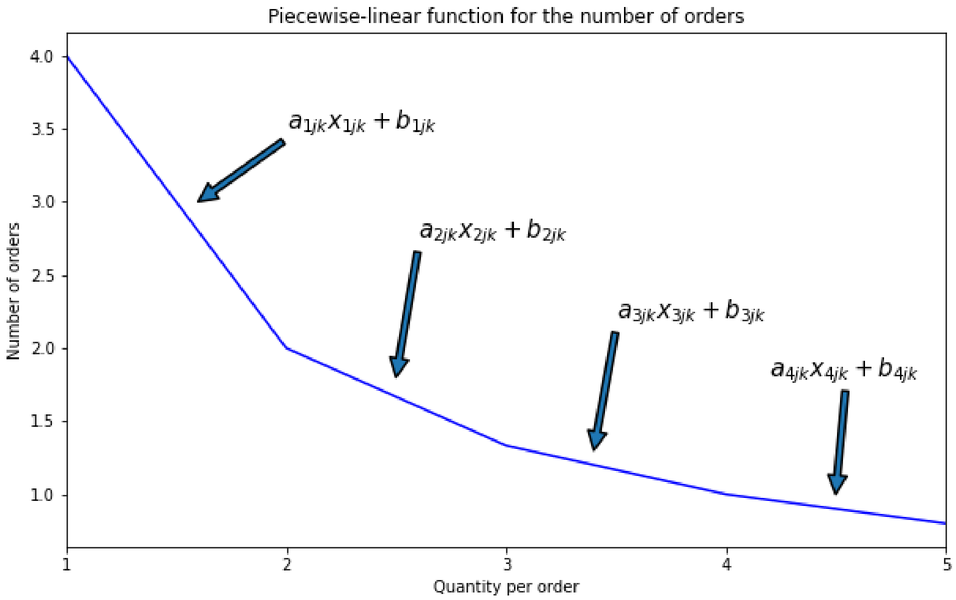


Figure 3: An example of the piecewise linear function for the number of orders.

The piecewise linear function used to find out the correct number of orders for each possible order quantity is plotted in Figure 3. The piecewise-linear function is introduced in this thesis to ensure a correct relation between the optimal order quantity and the number of orders for each purchase item $j$. The number of orders for purchase item $j$ is $n_j = \frac{D_j}{x_j} => n_j x_j = D_j$. This means that the piecewise linear function will, by definition, ensure that the demand $D_j$ for each time horizon (12 months), will be fulfilled. Because both $x_j$ and $D_j$ are always integers, the number of orders for a purchase item $j$, $n_j$ can have fractional and integer values in a time horizon.

The slopes $a_{ijk}$ and $y$-intercepts $b_{ijk}$ for each piecewise linear function $f_{ijk}$ are calculated as

$$a_{ijk} = \frac{\frac{D_j}{u_{ijk}} - \frac{D_j}{l_{ijk}}}{u_{ijk} - l_{ijk}} \tag{2}$$

$$b_{ijk} = \frac{D_j}{u_{ijk}} - u_{ijk}\left(\frac{\frac{D_j}{u_{ijk}} - \frac{D_j}{l_{ijk}}}{u_{ijk} - l_{ijk}}\right) = \frac{D_j}{u_{ijk}} - u_{ijk}a_{ijk}. \tag{3}$$

## 3.4 Notation

The following notations are used in the proposed model.

**Indices**

| | |
|---|---|
| $i$ | Index of each segment of the piecewise-linear approximation of the number of orders function, $i = 1, 2, ..., I$ |
| $j$ | Index of items, $j = 1, 2, ..., J$ |
| $k$ | Index of suppliers, $k = 1, 2, ..., K$ |

**Decision Variables**

| | |
|---|---|
| $x_j$ | Order quantity for purchase item $j$ |
| $x_{ijk}$ | Order quantity from supplier $k$ for purchase item $j$ in price break interval $i$ |
| $n_j$ | Number of orders for purchase item $j$ |
| $p_j$ | Price for purchase item $j$ |
| $p_{ijk}$ | Price from supplier $k$ for purchase item $j$ in price break interval $i$ |
| $f_{ijk}$ | Number of orders if the order size is in price break interval $i$ from supplier $k$ for purchase item $j$ |
| $y_{ijk}$ | A binary decision variable which is 1 if a price break interval $i$ is selected from supplier $k$ for purchase item $j$ and 0 otherwise |

**Parameters**

| | |
|---|---|
| $c_o$ | Ordering cost per order |
| $c_h$ | Annual warehouse cost for one $m^3$ |
| $V_j$ | Volume of purchase item $j$ |
| $r$ | Interest rate for capital costs |
| $s$ | Warehouse safety factor |
| $D_j$ | Annual demand for purchase item $j$ |
| $l_{ijk}$ | Lower bound of price break interval $i$ from supplier $k$ for purchase item $j$ |
| $u_{ijk}$ | Upper bound of price break interval $i$ from supplier $k$ for purchase item $j$ |
| $a_{ijk}$ | Slope of the line in the price break interval $i$ from supplier $k$ for purchase item $j$ |
| $b_{ijk}$ | $y$-intercept of the line in the price break interval $i$ from supplier $k$ for purchase item $j$ |

## 3.5 Optimization model

The proposed model of this thesis solves the optimal order quantity for each purchase item when the price breaks are known. The objective is to minimize the annual total cost function.

The final mixed-integer linear programming model from supplier $k = 1, 2, ..., K$ for product $j = 1, 2, ..., J$ is

$$\min_{x_j, n_j, p_j} \quad c_o n_j + p_j D_j + \frac{r}{2} p_j x_j + s V_j x_j c_h \tag{4}$$

$$\text{subject to} \quad x_j = \sum_{i=1}^{I} \sum_{k=1}^{K} x_{ijk} \tag{5}$$

$$p_j = \sum_{i=1}^{I} \sum_{k=1}^{K} p_{ijk} y_{ijk} \tag{6}$$

$$n_j = \sum_{i=1}^{I} \sum_{k=1}^{K} f_{ijk} \tag{7}$$

$$x_{ijk} - l_{ijk} y_{ijk} \geq 0, \quad i = 1, 2, ..., I, k = 1, 2, ..., K \tag{8}$$

$$x_{ijk} - u_{ijk} y_{ijk} \leq 0, \quad i = 1, 2, ..., I, k = 1, 2, ..., K \tag{9}$$

$$\sum_{i=1}^{I} \sum_{k=1}^{K} y_{ijk} \leq 1 \tag{10}$$

$$f_{ijk} = a_{ijk} x_{ijk} + b_{ijk} y_{ijk}, \quad i = 1, 2, ..., I, k = 1, 2, ..., K \tag{11}$$

$$\lim_{l_{(i+1)jk} \to u_{ijk}} f_{l_{(i+1)jk}jk} = f_{u_{ijk}jk}, \quad i = 1, 2, ..., I, k = 1, 2, ..., K \tag{12}$$

$$x_j, p_j, x_{ijk}, n_j, f_{ijk} \geq 0, \quad i = 1, 2, ..., I, k = 1, 2, ..., K \tag{13}$$

$$y_{ijk} \in 0, 1, \quad i = 1, 2, ..., I, k = 1, 2, ..., K. \tag{14}$$

The objective function (4) states that the annual total costs for item $j$ from supplier $k$ must be minimal. It consists of four components: annual ordering costs, annual purchasing costs, annual cost of capital and annual warehouse costs. Constraint (5) ensures that the order quantity $x_j$ is the sum of all $x_{ijk}$'s of which only one is nonzero and constraint (6) does the same for the price $p_j$ as only one interval of $y_{ijk}$ is nonzero. Constraint (7) determines the number of orders for each product $j$. Constraints (8) and (9) determine the price break intervals for $x_{ijk}$ and ensure that only one price break is chosen for product $j$ from supplier $k$, because only one value of $y_{ijk}$ is nonzero. If an interval is not selected, the constraints (8) and (9) ensure that the optimal order quantity does not belong to that interval. Constraint (10) makes sure that at most one price break interval in $y_{ijk}$ can be nonzero. Constraint (11) chooses the correct piecewise-linear function that determines the number of orders. Constraint (12) defines the continuity of the piecewise linear function at every breakpoint of every price break interval. Constraint (13) defines that every decision variable can only be a nonnegative amount while constraint (14) defines $y_{ijk}$ as a boolean variable.

The constraints in this proposed model are consistent with prior literature. Most constraints in the model are defined in the mathematical formulation of Goossens et al. (2007) and then modified to be suitable for the mathematical model of this thesis.

**Lemma 3.1.** There exists a unique optimal solution for purchase item $j$ from vendor $k$ such that
i) if $x_j > l_{ijk}$, then $x_j = u_{ijk}$ for all $x_j > 0$;
ii) if $x_j < u_{ijk}$, then $x_j = l_{ijk}$ for all $x_j > 0$.
**Proof.** A similar proof using an interchange argument is found in Zhang and Chen (2013). Suppose that there is a function $g_{jk}(x_j)$ for purchase item $j$ which is a piecewise-linear function that has $m_{jk}$ breakpoints such that $0 < l_{1jk} < ... < l_{Ijk} < u_{Ijk}$ and $u_{Ijk}$ is the maximum quantity of the given price break range. Also, $g_{jk}(x_{jk}) = p_{tjk}x_{jk}$ if $l_{tjk} \leq x_{jk} \leq l_{(t+1)jk}$ with $p_{tjk} > p_{(t+1)jk}$ for $t = 0, ..., m_{jk}$. Suppose that an optimal solution exists for a purchase item $j$ for two different suppliers $k$ and $u$ with optimal solutions $l_{tjk} \leq x_{jk} \leq l_{(t+1)jk}$ and $l_{hju} \leq x_{ju} \leq l_{(h+1)ju}$, respectively. We increase $x_{jk}$ by an arbitrarily small $\epsilon > 0$ while decreasing $x_{ju}$ by $\epsilon$. These changes to $x_{jk}$ and $x_{ju}$ are feasible and the change in the piecewise-linear function is $(p_{tjk} - p_{hju})\epsilon$. If $(p_{tjk} - p_{hju})\epsilon$ is negative, it contradicts the original optimal solution. If $(p_{tjk} - p_{hju})\epsilon$ is positive, a solution in which $x_{ju}$ is increased by $\epsilon$ and $x_{jk}$ is decreased by $\epsilon$, again contradicts the original optimal solution, since $(p_{tjk} - p_{hju})\epsilon$ would again be negative. If $(p_{tjk} - p_{hju})\epsilon$ is zero, then $p_{tjk} = p_{hju}$ and the optimal solution is either $x_{jk}$'s or $x_{ju}$'s nearest breakpoint. This argument can be applied repeatedly until an optimal solution at the breakpoints is reached.

# 4 Implementation

Moussourakis and Haksever (2008) state that obtaining solutions to zero-one mixed-integer programming models can be time consuming. In this thesis, an exact algorithm was developed to minimize the annual total cost function and its corresponding optimal order quantity for each purchase item. This means that the algorithm finds an optimal solution, unlike approximate algorithms which usually find near optimal solutions to problems that cannot be solved easily (Kokash, 2005).

For the computational experiments, we developed a computer program using Python's MIP package (Santos and Toffolo, 2020). A user of the developed software implementation must provide the item number, annual demand ($D_j$), weight ($w_j$), the ordering cost for one order ($c_o$), the annual warehouse cost for one $m^3$ ($c_h$), the prices of each price break interval ($p_{ijk}$) and the lower and upper points of each price break ($l_{ijk}$ and $u_{ijk}$) for each purchase item $j$. After this, the program returns the optimal annual total cost function and its corresponding optimal order quantity, the annual number of orders and the unit price of the chosen order quantity.

The average computation times for all test instances in this thesis are reasonable when using the Python's MIP library. This is supported by the computational results of this thesis (see Table 2). Reasonable computation times are important for the usability of the model. The software must be able to solve large test instances in a short amount of time to be useful in practice.

# 5 Computational experiments

## 5.1 Test instances

| Item number | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Annual demand ($D_j$) | 4 | 142 | 5 | 12 | 17 |
| Cost per order ($c_o$) | 50 | 50 | 50 | 50 | 50 |
| Annual warehouse cost ($c_h$) | 4.6 | 4.6 | 4.6 | 4.6 | 4.6 |
| Weight ($w_j$) | 23.15 | 0.132 | 57.4 | 20.1 | 17.31 |
| Price break intervals ($l_{ijk}$, $u_{ijk}$) | 0 - 1 | 0 - 9 | 0 - 1 | 0 - 1 | 0 - 1 |
| | 2 - 3 | 10 - 49 | 2 - 3 | 2 - 3 | 2 - 5 |
| | 4 - | 50 - | 4 - | 4 - | 6 - |
| Price for each interval ($p_{ijk}$) | 419.97 | 7.2 | 1 298.5 | 427.0 | 299.0 |
| | 335.98 | 6.0 | 1 038.5 | 366.5 | 196.0 |
| | 251.98 | 5.45 | 919.5 | 312.0 | 122.64 |

Table 1: Example of parameters for 5 purchase items with 3 price break intervals for each purchase item.

Table 1 shows the parameters for test instances. The test instance in Table 1 is the first test instance in Table 2. The test instances are all formed from historical data provided by Normet.

| Test instance | Nr. of purchase items | Nr. of price break intervals | Average CPU time (secs.) |
|---|---|---|---|
| 1 | 5 | 3 | 0.030 |
| 2 | 5 | 5 | 0.036 |
| 3 | 5 | 10 | 0.046 |
| 4 | 10 | 3 | 0.059 |
| 5 | 10 | 5 | 0.069 |
| 6 | 10 | 10 | 0.091 |
| 7 | 20 | 5 | 0.10 |
| 8 | 20 | 10 | 0.18 |
| 9 | 50 | 5 | 0.24 |
| 10 | 50 | 10 | 0.40 |
| 11 | 100 | 5 | 0.84 |

Table 2: Computational results of different problem sets containing different number of purchase items and different number of price break intervals.

Table 2 lists the properties for each test instance. Including the number of purchase items, the number of price break intervals and the average computational run time for each test instance. All test instances were solved using a Lenovo

ThinkPad laptop with an Intel i5 2.40GHz processor with 8GB RAM and Windows 10 operating system. The order quantities for all purchase items in each test instance were calculated exactly to optimality. A larger test instance takes, on average, more computation time to solve. The CPU times of all test instances were very reasonable thus enabling inventory managers to run optimization on their laptop in a short amount of time.

The computer implementation works up to par or even better than previous implementations for small test instances (Jackson and Munson, 2016) (Moussourakis and Haksever, 2008). However, for the largest test instance in Table 2, the average computation time is longer than the computation time for similar test instances in prior literature (Jackson and Munson, 2016).

## 5.2 Results

The proposed model of this thesis optimizes each purchase item individually. Hence, there are no resource constraints, such as budgetary or warehouse capacity in the model. The lack of resource constraints usually leads to larger order quantities to the higher end of the price break range (Table 3). For example, in the second last test instance in Table 2 with 50 purchase items, the largest possible order quantity per order is chosen for 24 purchase items. This means that the benefits of price breaks often outweigh the drawbacks of warehousing and capital costs.

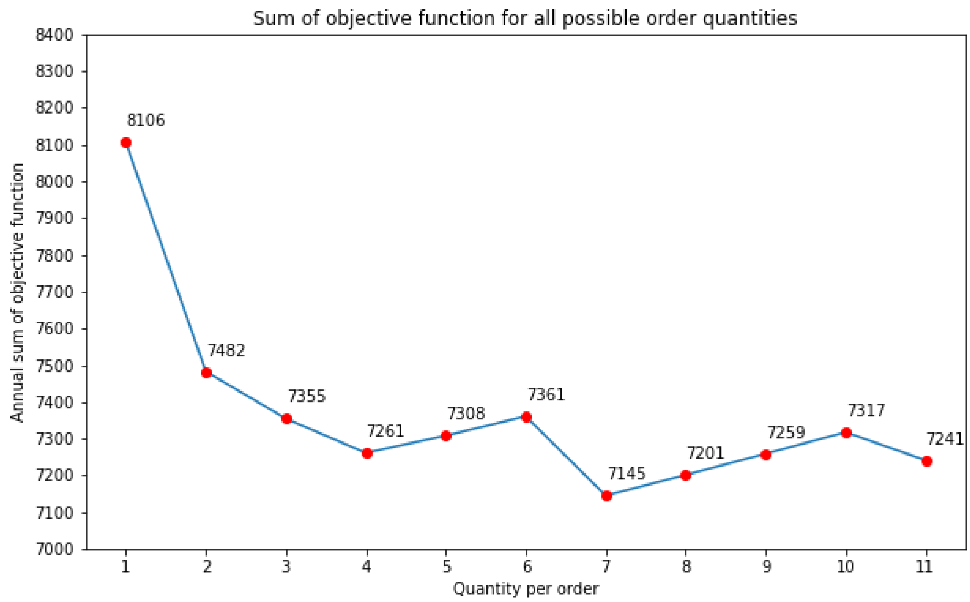


Figure 4: An example of the calculated objective function values for each possible order quantity for a purchase item.

Figure 4 gives an example of all possible quantities in the price break range and their annual total cost functions for a purchase item. The price break range here goes from one to 11. The optimal annual total cost function value for this purchase item is 7145 with an order quantity of seven.

| Item number | Optimized order quantity | Reference order quantity | Savings percentage |
|---|---|---|---|
| 1 | 4 | 2 | 24.4 % |
| 2 | 50 | 20 | 23.5 % |
| 3 | 4 | 1 | 28.5 % |
| 4 | 4 | 4 | 0 % |
| 5 | 7 | 2 | 40.1 % |
| Average | | | 23.3 % |

Table 3: The optimized order quantities, current order quantities and the savings percentages for each purchase item in test instance 1 of Table 2.

Table 3, lists the optimized order quantity, the reference order quantity from historical data and the savings percentage for each purchase item in test instance 1 of Table 2. In order to estimate whether the results are reasonable and/or statistically significant, we assess how the optimized objective function values compare with actual historical data of Normet's purchase order data where we find the average order quantity and the reference annual total cost function value for each purchase item under investigation. In this test instance, the model finds optimal solutions for four out of five purchase items.

The lack of resource constraints makes it possible for the proposed model to utilize larger order quantities to achieve savings. However, in reality, one has to verify that the implied warehousing and capital costs are acceptable for the optimized order quantities.

The savings percentage for almost all purchase items in Table 3 is significant. For the five inventory items in Table 3, the savings percentage for four purchase items is over 20%, the average savings percentage is 23.3 % and the highest savings percentage is 40.1%. On the scale of businesses, with possibly thousands of purchase items with price breaks, total savings can be substantial.

| Test instance | Average savings percentage |
|---|---|
| 1 | 23.3 % |
| 2 | 4.2 % |
| 3 | 14.1 % |
| 4 | 14.0 % |
| 5 | 34.1 % |
| 6 | 13.2 % |
| 7 | 19.7 % |
| 8 | 9.1 % |
| 9 | 26.8 % |
| 10 | 9.9 % |
| 11 | 14.2 % |

Table 4: The average savings percentage for each test instance.

Table 4 lists the average savings percentage for each test instance. Each test instance results in positive average savings percentages and some test instances have significant average savings percentages. In all test instances, the model finds optimal solutions consistently, which leads to savings.

The assumptions made in this thesis affect the results significantly. If we were to remove assumptions, such as the lack of resource constraints, the proposed model would not favor larger order quantities as much as it currently does. Depending on the resource constraints, every purchase item could not freely take any quantity in the given price break range.

The results of this thesis appear reasonable and reliable. The trend of preferring higher quantities is consistent with all test instances. The proposed model leads to higher order quantities than their historical values have been. This is in line with previous intuitive expectations of Normet. The results show that the proposed model can result in significant savings and optimal purchase decisions in practice.

# 6 Conclusions

In this thesis, we first used findings from prior literature to formulate a mixed-integer linear programming model for optimizing a multi-product inventory system with all-units price breaks offered by suppliers. Then, the model finds the optimal order quantity from the objective function which is the annual total cost function defined in this thesis, for each purchase item individually. Next, we compare the objective function values and optimized order quantities with the function values and reference order quantities from reference data provided by Normet. The model was tested on a Lenovo ThinkPad laptop by solving 11 different test instances, each containing a number of purchase items ranging from five to 100 and a number of price break intervals ranging from three to ten.

This thesis indicates that the proposed model can be expected to help Normet achieve significant savings. The difference of the current and optimized annual total costs for all purchase items can be as high as 25% (Table 4). The results of the model were consistent, every purchase item that had not been optimized resulted in savings after optimizing their order quantities. Since the model is unconstrained, it often favors larger order quantities.

The computational results of the implemented model in Python suggest that the model can be solved in a reasonable amount of time on almost any modern PC or laptop. The model can solve practical problems with thousands of purchase items and several price break intervals if needed, which makes it a viable potential tool for inventory managers. However, considering the assumptions of the model, it is not suitable for general use without modification. The model has several assumptions that make it usable for specific situations only. Several assumptions of the model make it usable for specific situations only. It can, however, be used in instances where every purchase item has all-units price breaks, demand is known and there are no resource constraints.

Future research can extend the thesis by removing assumptions such as deterministic demand and independent cycle length. A future study could address topics such as variable lead time, stochastic demand and safety stock. The addition of incremental price breaks and resource constraints and the possibility to choose between independent and fixed cycle lengths could be topics for future research, as could be the question of expanding capacity size to utilize the price breaks for all purchase items (see Jackson and Munson, 2016). Moreover, it would be fascinating to see which purchase items have such drastic price breaks that inventory capacity should be expanded to incorporate them.

# References

W.C. Benton and S Park. A classification of literature on determining the lot size under quantity discounts. *European Journal of Operational Research*, 92:219–238, 1996.

D. Goossens, A.J.T Maas, F. Spieksma, and J Van de Klundert. Exact algorithms for procurement under a total quantity discount problems structure. *European Journal of Operational Research*, 178:603–626, 2007.

J. Jackson and C. Munson. Shared resource capacity expansion decisions for multiple products with quantity discounts. *European Journal of Operational Research*, 253: 602–613, 2016.

N. Kokash. An introduction to heuristic algorithms. *Department of Informatics and Telecommunications*, 1-8, 2005.

Amy H.I. Lee, He-Yau Kang, Chun-Mei Lai, and Wan-Yu Hong. An integrated model for lot sizing with supplier selection and quantity discounts. *Applied Mathematical Modelling*, 37(7):4733–4746, 2013.

J. Moussourakis and C. Haksever. A practical model for ordering in multi-product multi-constraint inventory systems with all-units quantity discounts. *International Journal of Information and Management Sciences*, 19:263–283, 2008.

C. Munson and J. Jackson. Quantity discounts: An overview and practical guide for buyers and sellers. *Foundations and Trends in Technology Information and Operations Management*, 8:1–130, 2015.

P.A. Rubin and W.C. Benton. Jointly constrained order quantities with all-units discounts. *Naval Research Logistics*, 40:255–278, 1993.

H.G. Santos and T.A. Toffolo. Mixed integer linear programming with Python, 2020.

J. Zhang and J.J. Chen. Supplier selection and procurement decisions with uncertain demand, fixed selection costs and quantity discounts. *Computers Operations Research*, 40:2703–2710, 2013.