CHALMERS

Mathematical optimization of a returnable packaging system

- minimizing the total cost of distribution using MILP

Master of Science Thesis in Engineering Mathematics and Supply Chain Management

ANNELIE FORSLUND
FRIDA STOLT

Department of Mathematical Sciences
Division of Optimization
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2012

Abstract

Mathematical optimization is a methodology that can be applied when a quantitative approach for solving a problem is desired. The frequency at which mathematical optimization is used differ between different fields of applications and one, not too common field to study in this respect, is logistics. Logistical problems can be difficult to optimize since significant parts of the problems are typically non-quantitative by nature.

SKF Group is a leading global supplier of products, solutions, and services within rolling bearings, seals, mechatronics, services, and lubrication systems. It uses standard packaging for their products which can be reused through a Returnable Transport Packaging Program. Today SKF Logistics Services, which is a part of SKF Group, is managing the flow of empty returnable packaging in SKF's distribution network. It asked for a study, partly to verify whether of not the Returnable Transport Packaging Program is managed in a proper way, partly to investigate the possibilities for improvements in the handling of the packaging. The purpose of this project is, therefore, to identify the improvement potentials by creating a mathematical optimization model with the objective to minimize the total cost related to transportation and purchasing of SKF Group's returnable packaging world-wide.

Before creating a mathematical optimization model, the real world problem needs to be quantified. Most parts of the problem are described by using linear equations and inequalities, but some parts are described discretely. Hence the problem is described by a mixed integer linear program (MILP). This type of optimization model can be solved by using Branch and Bound (B&B), which is a method for exploring all possible solutions to the problem in an efficient way by using the nice properties that are associated with linear models. In the worse case B&B is, however, no faster than a complete enumeration method. Besides formulating the optimization model, a lot of data collection is needed to give parameters in the model values. This data was retrieved through information systems, questionnaires, and meetings. The problem consists of a huge amount of variables, which makes the model unsolvable for more than two time steps, due to lack of computer capacity. A relaxation of the integrality requirements on some of the variables was therefore necessary, which makes the problem faster to solve. The solution represents the real world problem reasonable well since a substantial part of the variables which had their integrality requirement removed represents relatively large flows. A further study to this project could be to investigate the possibilities to use decomposition algorithms instead of relaxing the integrality requirements on some of the variables.

The result was compared to SKF's current management of the distribution of the returnable packaging, and a conclusion can be made that the result is reasonable. Based on the result some recommendations are made, some of which are possible to implement. When creating an optimization model it is necessary to illuminate the system, which has been of great value for SKF since the questions have initiated important discussions concerning how the operations are managed today.

Keywords: Returnable packaging, Mixed integer linear program, Optimization modelling, SKF

Acknowledgements

We end our studies at Chalmers University of Technology with this master thesis project. In the project mathematical optimization theory and modeling has been applied to a practical problem, which has given us a great insight in how unpredictable, dynamic, and complex real world problems can be. The work has been exciting, fun, and sometimes frustrating which has made the last months pass very quickly.

We would like to thank our supervisor Mattias Axelsson at SKF Logistics Services for giving us the opportunity to work with such an interesting case. He has been a great support throughout the project and very responsive to our needs, due to his great ability to adapt to our way of thinking. He is also a quick learner and a good listener which has been very helpful. During the spring we have been located at SKF's headquarter in Gothenburg and would therefore like to thank the entire SKF Logistics Services for being so welcoming and for all the interesting discussions we have had during the lunch breaks. Special thanks to Magnus Wellsted for putting so much effort into our Excel sheets.

Last but not least, a big thanks to docent Ann-Brith Strömberg, our supervisor at the Department of Mathematical Sciences at Chalmers, for guiding us through this project.

Annelie and Frida, Gothenburg, June 2012

Contents

1

Introduction

Logistics is a crucial function in many companies and a successful logistics solution can result in lower costs, higher customer satisfaction and less environmental impact [1]. A single company's logistics network can be very large and the trend during the recent decades has been that the networks have grown in complexity due to an extended globalization [2]. Logistics can therefore be a very complicated topic. There are different ways to solve different types of logistic issues and one, not very common, approach is to use mathematical optimization modeling. Since mathematical optimization is often used when solving different types of network problems, it can be a useful tool when dealing with logistics networks as well.

The question whether producing companies shall deliver their products in returnable or one-way packaging has been studied for a long time within supply chain management. The trend since the early '90s has favored returnable packaging. This was adopted by the Swedish based company SKF Group when introducing a Returnable Transport Packaging Program, in which packaging is returned, cleaned, and reused.

1.1 Background

SKF Group (SKF) is a stable company that has been in the industry for more than 100 years. A reason why SKF has succeeded is that it is an innovative company that strives for continuous improvements in all parts of its business. The project that this report concerns has focused on an area at SKF which is considered to have a lot of improvement potentials.

1.1.1 Company Presentation: SKF Group

SKF was founded in 1907 in Gothenburg, where the headquarter is still located. The engineer Sven Wingquist invented the first self-aligning ball bearing and started the company in order to commercialize the idea. Since then, the company has extended its business and today SKF is a leading global supplier of products, solutions, and services within rolling bearings, seals, mechatronics, services, and lubrication systems. The products are produced all over the world and distributed to customers in all kinds of industries. In 2011 the company had a turnover of 66,216 million SEK and over 46,000 employees.

1.1.2 SKF Logistics Services

SKF Logistics Services (SLS) is a part of SKF who's business area is to independently sell and deliver logistics services worldwide. SLS is represented in 15 places around the world, employing over 1,200 people with the mission to realize the expectations of the customers concerning delivery precision. 90 % of SLS customers belong to SKF and the reaming are external companies. SLS serves as a third party logistics provider (3PL) which means that it does not own a freight hauling fleet of its own; instead, the demanded capacity are bought from other logistics providers specialized in these areas. The warehouses in SLS's network are owned by SLS and used for storage of both SKF's and external customers' products. SLS also provides other services than transportation and warehousing; among them packaging, labeling, inventory management, and information sharing. The idea is to provide an end-to-end logistics solution and give the customers the possibility to focus on their core competences.

To be able to deliver what is promised to customers, SLS has developed some solutions to facilitate the work. One of these is the usage of standardized returnable packaging, which are designed to minimize the handling, picking, and repacking operation in the whole chain from producer to consumer. The returnable packaging is used during both transport and storage and can be adapted to specific customers' needs. Even though SLS strives for increasing the use of returnable packaging, not all products are suitable to be transported and stored in these standardized items, for example due to size, product value, and weight. These products are instead shipped in other types of packaging, for example cardboard containers or specially designed boxes.

1.1.3 The issue posted by SKF Logistics Services

The returnable packaging used at SLS is shown on the front page of this report. The flow of returnable packaging is unified in the Returnable Transport Packaging Program at SLS. The objective of the program is to increase efficiencies and reduce costs associated with distribution. The forward flow of returnable packaging, when containing products, is determined by customer orders and can not be controlled by SLS. The backward flow, how the returnable packaging is handled and where it is transported when emptied of products is, however, controlled by SLS. Today, the management of the backward flow is centralized at SLS. Decisions are made on a monthly basis regarding the world-wide distribution of empty returnable packaging in order to meet the demand in different areas. This decision process is not standardized and is mainly dependent of qualitative analysis [I].

SLS wants to verify that the Returnable Transport Packaging Program is managed in a proper way and investigate whether any possibility for improvements in the handling of empty returnable packaging exist. A quantitative analysis of the situation can also be helpful when dealing with stakeholders. Generally, process as important and prioritized as the ones concerning the Returnable Transport Packaging Program are already handled in a fairly good way, but if it is possible to improve these processes any further, the capital released could be extensive. SLS has, because of its great importance and potential capital release, asked for a study of the situation by using mathematical optimization, which was considered to be a good approach since the data needed for creating a model was believed to be present.

1.2 Purpose

The purpose of the project is to identify improvement potentials of the distribution of SKF Logistics Services' returnable packaging by creating a mathematical optimization model with the objective to minimize the total cost related to transportation and purchasing world-wide.

1.3 Scope and Limitations

The model is created with respect to the situation at SKF in 2011 which would give an opportunity for SLS to compare the outcome of the model with the historical decisions. The optimization model focuses on the flows that SLS can control, namely the routes the empty returnable packaging can take, while all other flows in the model are considered fixed. The result of the optimization model will therefore be concerning the flows of the returnable packaging during 2011, defined over 12 time steps representing each month of the year.

An optimization model is a modification and a simplification of a real world problem since it is, in most cases, impossible to create a quantitative formulation of the problem concerning all properties involved. The result obtained from creating an optimization model is therefore just optimal given the model formulation, not with respect to the real world problem. This is important to understand when analyzing the results and when making decisions based on these.

The outcome of the project will not be a computer program that SLS can use for optimizing flows in the future. The result will be based on the data originating from the chosen time horizon only, and the project delivery will only include the result given by the optimization model and a compilation of the data retrieved when creating it.

The project focuses on the handling of returnable packaging for SKF. This means that no attention has been paid to the external customers of SLS or shipments performed in other packaging than the returnable packaging.

SKF has a lot of factories, suppliers, and customers around the world. In order to simplify the project, a decision was made to stop the degree of details on a country level. This means that if

SKF has more than one factory in a specific country these will be nested into one. Analogous for the suppliers and for the customers.

The project will not question whether of not the Returnable Transport Packaging Program should be used.

1.4 Thesis outline

The methods used when formulating the optimization model and retrieving the data are described in Chapter 2. This project is based on two main subjects, returnable packaging systems and mathematical optimization. It is important to understand the theories of these concepts to be able to create a desirable optimization model; and they are presented in Chapter 3. Before a mathematical model can be formulated, the real world problem must be captured and interpreted as properties of the problem. Chapter 4 therefore describes the nodes and the links of SKF's distribution network, together with the costs associated with the distribution. The optimization model is introduced in Chapter 5 where the properties of SKF's distribution system is defined mathematically as constraints and the purpose of the project as an objective function. The chapter also presents the complete optimization model. To verify the correctness of the model is necessary to test it, and this procedure is described in Chparter 6 which also contains the obtained result. This result, along with the data, is discussed in Chapter 7. The final chapter summarizes the conclusion drawn from the project. In order to distinguish between regular references and references from SKF and SLS are two bibliography present in this report. The regular references are found after Chapter 8 and is referred to with Arabic numerals. A list of people interviewed at SKF and SLS is found in Appendix A and is referred to with Roman numerals.

2

Method

THIS chapter describes how the process of creating an optimization model to find the improvements potential in SKF Logistics Services' distribution of returnable packaging progressed. Initially, the optimization model was formulated. This process is called model formation and comprises a simplification of the real world problem in order to be able to express it mathematically. Realistic sizes of real world optimization problems are almost always too large and complex to be solved by hand and therefore it is necessary to utilize an optimization software package. In order to obtain a result from the optimization model it must be fed with reasonable data. Data in this project originates from appointments with SLS, questionnaires, and SKF's data systems.

2.1 Model formulation

Formulating an optimization model can be very difficult if the real world problem in question is large and complex. It is therefore important to approach the problem in a structured way. The modeling process used in this project is schematically illustrated in Figure 2.1. This process has been described by both Andréasson et al. [3] and Lundgren et al. [4] – books in mathematical optimization.

2.1.1 Real Problem

A real world problem can often be very complex and difficult to define and this is the first part of the modeling process shown as a fluffy cloud in Figure 2.1. To be able to utilize mathematical optimization it is important that the problem can be quantitatively described. When the problem definition is made all relevant components within the problem needs to be identified. Most real world problems are non-quantitatively by nature which make some parts of the problem very difficult or impossible to include. Simplifications of the real world problem are then necessary.

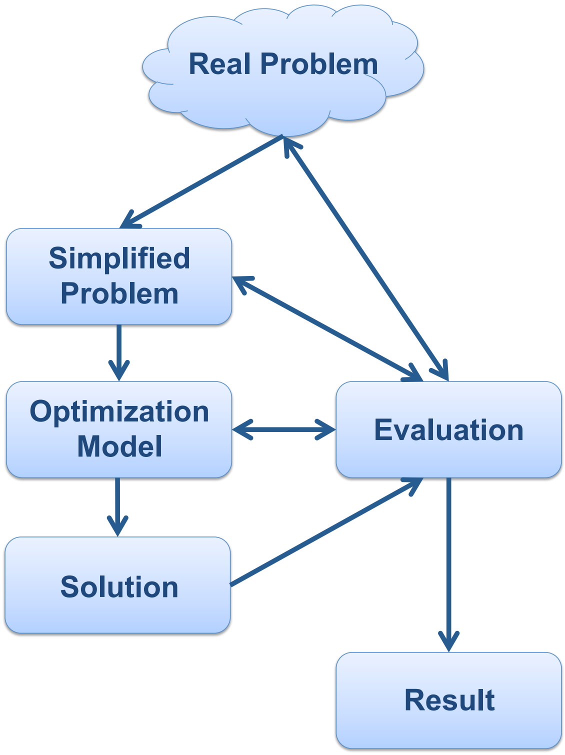


Figure 2.1: A schematic figure of the modeling process employed in this project.

2.1.2 Simplified Problem

Almost all real world problems need simplifications when formulating a mathematical optimization model, which is illustrated by the box Simplified Problem in Figure 2.1. As mentioned earlier some parts of the real world problem can be very difficult to quantify and if these parts are crucial it could be worth allocating extra time and effort to figure out whether and how to include them. If they are not worth the hazard, suitable simplifications are necessary in order to make the simplified problem reflecting reality satisfactory. When making simplifications it is important to understand how they might affect the final result. The simplified problem will not be good enough to represent the real problem if too many simplifications are made. How many and what simplifications that are acceptable to make is different for every problem and are needed to be evaluated from case to case. In this phase of the problem definition it is appropriate to ask whether or not mathematical optimization is the best approach.

To understand the problems SLS is facing, meetings with the management of SKF Logistics Services has been conducted to gain information and understanding of the problem [I]. The meetings have taken place continuously throughout the whole project and the simplified problem has been reformulated numerous times until a satisfying version was composed. This iterative process of redefining the problem definition is illustrated by arrows between Evaluation and Real

Problem in Figure 2.1.

2.1.3 Optimization model

When all relevant components are identified and all simplifications are made the optimization model can be created. This part of the modeling process is represented by the box Optimization Model in Figure 2.1. During this phase all information in the simplified problem is formulated mathematically and coded to create an optimization program. This phase can also be conducted in parallel to the second phase where the simplified problem is defined. The difficulties of the problem might not be fully understood until the mathematical formulation of the problem is created. Further simplifications might be needed to make the optimization model ready for the solution phase. This iterative process is illustrated by arrows between Optimization Model and Evaluation in Figure 2.1.

2.1.4 Solution and Result

When a solution is obtained from the solver it must be evaluated and compared to the real world problem. This is shown as arrows pointing from Solution to Evaluation and Real Problem to Evaluation in Figure 2.1. If the solution does not seem to be accurate the optimization model needs to be reformulated and resolved. This process is iterative and will result in a model that more and more reflects reality. The improvement of the model stops when solution obtained is considered to be accurate or good enough for the purpose. Eventually a final result is obtained which is illustrated by an arrow pointing from Evaluation to Result in Figure 2.1.

2.2 Optimization modeling language and solver

The mathematical formulation of the problem is programmed using a model language called A Mathematical Programming Language (AMPL) [5]. The program created using AMPL is solved by the optimization software package CPLEX [6].

2.2.1 AMPL

AMPL is structured in such way that the optimization model is divided into three different parts; a data part, a model part, and a command part. This is accomplished by the creation of three files with the extension `.dat`, `.mod`, and `.run`, respectively. In the model file the sets, parameters, variables, and constraints of the optimization model are defined. The data representing the sets and parameters is stated in the data file. For example, if a set $\mathcal{S}$ is defined in the model file, the data $\mathcal{S} = \{1, 2, 3\}$ is defined in the data file. This creates a good structure and a big and complex problem is then easier to handle. The command file expresses how to execute the model and data files. The command file can for example include logical operators, instruct the solver how to solve the problem, and state how to display the result. AMPL supports many different types of optimization models.

2.2.2 CPLEX

There are many commercial solvers available on the market, such as GLPK, FortMP, and Gurobi, and the one chosen for solving the optimization model crated for SKF is CPLEX. CPLEX is a software implemented in the C programming language and was initially named after the simplex method which is a method for solving integer, linear, and quadratic models. Linear and integer models are described in Sections 3.3 and 3.4.

2.3 Data collection

The result obtained from the solution of an optimization model is depending on the data fed into the model. The quality of the data is therefore crucial for the usability of the result from the model. This implies that not only the simplified version of the real world problem must be carefully evaluated; also the data collected from the real world problem must be evaluated to determine how consistent the data is with the real world problem.

To obtain satisfying results the data collection must be stringent. Different sources and methods were used for collecting the data depending on the type of the data. The two categories of data needed in this project are sets and parameters.

2.3.1 Appointments with SLS

The biggest source of information used in this project has been SLS. SLS had great influence during the Real Problem and Simplified Problem phase of the project, see Figure 2.1, when it comes to defining scope, limitations, and SKF's distribution network [I]. The meetings conducted with SLS have been of different kinds; interviews with structured and open questions, teaching sessions at which discussions where held freely and occasions with more static questions. The outcome of the project is strongly dependent on the information given by SLS, but since SLS is the project owner this is both necessary and inevitable.

The project was managed from SKF's headquarter in Gothenburg, Sweden. This has resulted in opportunities to visit the local SLS unit in Gothenburg. A warehouse visit was performed early in the project in order to get an idea of how the daily operations are performed and to understand the processes better. Information from SLS Sweden also provided an understanding of what kind of information other SLS units around the world might be able to provide [XI].

2.3.2 Questionnaires

Since the organization at SLS is rather decentralized a lot of decisions are made locally. In order to possess the data related to the locally made decisions a questionnaire was created and sent to eleven local SLS units around the world [II], [III], [IV], [V], [VI], [VII], [VIII], [IX], [XI], [XII], [XIII], [XIV], [XVII]. The list of interviews is found in Appendix A and the questionnaire can be reviewed in Appendix C. An answering sheet was attached to the questionnaire in order to get as standardized answers as possible. The data obtained was mainly parameters needed in the optimization model. Some parameters used in the model is described in Section 5.1

When the questionnaire was designed some help was provided by the SLS unit in Germany [XVII]. The questionnaire was discussed and input from SLS Germany were given in order to create a questionnaire that would be correctly read and interpreted by all the other SLS units in the world.

2.3.3 SKF's information systems

The optimization model needs data related to the amount of returnable packaging used when transporting products between different nodes in SKF's distribution network. The data used for the parameters was provided by two of SKF's information systems, Transport Management System (TrMS) and Logistics Charges Cube. TrMS tracks outbound flows of returnable packaging used when transporting products from warehouses in SKF's distribution network while Logistics Charges Cube tracks inbound flows at the warehouses.

2.4 Data Processing

In order to handle the huge amount of data needed for some of the parameters used in the model in a structured way, the software Microsoft Excel was used. The reason for the choice of software was that the reports created by TrMS and Logistics Charges Cube when retrieving data from the information systems are delivered as Excel spreadsheets. It is also favorable to use the same software as SKF since it facilitates communication and collaboration.

The structure and data format of the spreadsheets retrieved from the information systems are however not the same as the ones needed when loading data to AMPL. To be able to load data for a parameter to AMPL the data has to be saved in a text file "`.txt`", in which each line represents a specific index combination of that parameter. Excel has a lot of supporting functions for reorganizing data and these were used when generating the text files. Some parameters used in the model were not available directly from the reports created by the information systems but could easily be calculated using Excel and the data given by the reports.

3

Theory

THE previous chapter described the method used for creating an optimization model with the intention to find improvement potentials in SKF Logistics Services' distribution of returnable packaging. The subject of this chapter is to present the theory behind the two main topics of this thesis; the logistic concept of returnable packaging material and mathematical optimization. The distribution at SKF is based on a returnable packaging system and the benefit received from using such a system increases as the number of nodes involved in the system grows. The theory of mathematical optimization builds on some essential mathematical definitions which lead to the definition of linear programming.

3.1 Theory of Logistics: Returnable Packaging

Since the early '90s there has been an increased trend of diminishing the usage of one-way packaging in favor of returnable packaging. This is due to numerous reasons [7], of which one is the waste that the one-way packaging material gives rise to. Even if many one-way packaging, for example cardboard, are recyclable they create a disposal process which can be both costly and hindering companies from focusing on their core competences [7]. The waste that the one-way packaging generates is not in compliance with an environmental friendly policy that most companies strive for today. The combination of legislation and customers' requirements of companies' environmental responsibility force the producing industries to place their packaging at the top of the agenda [7]. But there are more reasons for companies starting to look into alternatives to one-way packaging: the cost of raw material is rising, the disposal costs increase, and the environmental legislation develops. All together this increases the incentives to reuse material to a larger extent [7].

Even though there is a large investment associated with the introduction of a returnable packaging system there is a lot of benefits to achieve. One obvious benefit is that the cost of one-way packaging purchases and the cost associated with its disposal handling can be decreased [8].

The returnable packaging, often made of wood or plastic, is also more robust than one-way packaging making the products carried more protected against the surroundings. This in turn decreases the degree of unsaleable products due to damages [9]. A well protected product is also less exposed to theft [7]. When a returnable packaging is introduced it will constitute the standard of the packaging in the system. This facilitates the handling within warehouse operations since standardized equipment can be used more extensively [10]. A more extensively use of standardized equipment makes the process more ergonomic and thus labour saving [7]. The standardized packaging should be designed to fit all products in the system, wherefore the need of specialized packaging is diminished. This results in a reduced need for packaging varieties which in turn decreases the total safety stock of packaging [11].

Most of the benefits are reaped when the returnable packaging system is used extensively; both for transport to customer, between companies' facilities, and for storage in warehouses [10]. If the incoming and outgoing goods, and all the operations in between, can be handled in the same way the efficiency of the processes increases [8].

To set up a returnable packaging system some criteria need to be fulfilled. Initially a closed loop is desired since this facilitates the control of the packaging [8]. Since the benefits gained increases when a whole supply chain uses the same standard it is common that the size of the loop grows in order to involve additional actors in the system. When more actors are involved the control of the system needs to be more extensive [11]. When the system expands, so must the number of collecting points of packaging for actors of the most downstream activities in the supply chain [8]. The system builds on end-customers returning the packaging, otherwise the efforts of having a returnable packaging system are useless. Some companies use a deposit system to create an incentive to keep packaging in the closed loop and hence increase the return rate of the packaging from the end-customers [9]. A shorter cycle yields a better control and hence a higher return statistic [11]. In order to make a system like this a reality a well-managed supply chain is required which entails a faultless forward logistics process [11]. The system works best when the distances between different locations in the supply chain is short, otherwise the transport cost of the returns tends to overshadow the benefits [11]. It is advantageous if the returnable packaging is collapsible because that decreases the volume of the return transports that each piece occupies [11].

Before introducing a returnable packaging system, some calculations on the investments are necessary. A returnable packaging system involves costs that a one-way packaging system is lacking; a new cost incurred in the purchase of returnable packaging, an increased cost for the transportation due to the returns, and cost associated with sorting, cleaning, and repair activities that the returned packaging gives rise to [7]. All these costs have to be covered by the savings that the increased handling efficiency, eliminated purchase of new one-way packaging, and disposal costs generate. If the savings are not covering the additional costs of a returnable system, it is not justified [11]. The suitability of a returnable packaging system also depends on the life length of the returnable packaging, which in turn depends on the purchase price versus the quality of the packaging's material. The number of cycles an item can perform during one year affects the total number of items needed; this have major effects on the initial investment cost [7].

3.2 Mathematical definitions

To be able to understand the basic theory behind mathematical optimization a few definitions are needed. These definitions are used when defining linear optimization in Section 3.3.

3.2.1 Convex set

Definition 1. *Let $\mathcal{S} \subseteq \mathbb{R}^n$. The set $\mathcal{S}$ is a convex set if*

$$\left.\begin{array}{r} \boldsymbol{x}^1, \boldsymbol{x}^2 \in S \\ \lambda \in [0,1] \end{array}\right\} \Rightarrow \lambda \boldsymbol{x}^1 + (1-\lambda)\boldsymbol{x}^2 \in S$$

holds [3].

This definition states that a set is convex if it is possible to chose any two points in the set and draw a straight line between these two points such that all the points on the line are also in the set. An other way to describe this is that all points in the set are visible to all other points in the set. A convex and a non-convex set are shown in Figure 3.1.

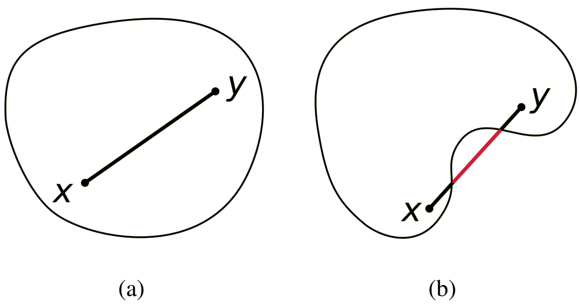


Figure 3.1: A convex set (a) and a non-nonvex set (b). Not all points on the straight line between the points x and y in (b) are in the set and consequently the set is a non-convex set.

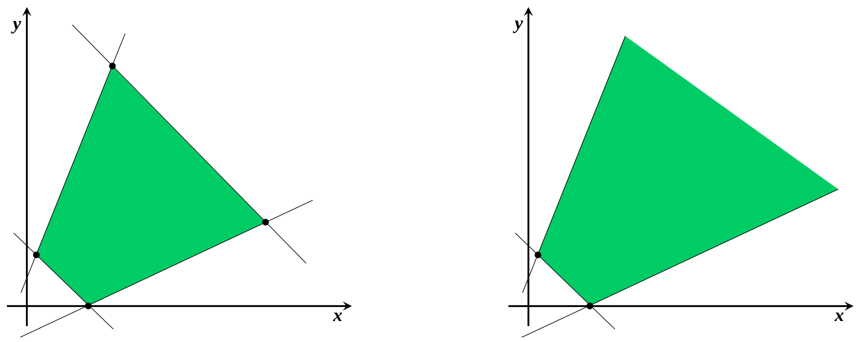
3.2.2 Polyhedron

Definition 2. *A subset $\mathcal{P} \subseteq \mathbb{R}^n$ is a polyhedron if there exists a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ and a vector $\boldsymbol{b} \in \mathbb{R}^m$ such that*

$$\mathcal{P} = \{\boldsymbol{x} \in \mathbb{R}^n \,|\, \boldsymbol{A}\boldsymbol{x} \leqslant \boldsymbol{b}\}$$

holds [3].

In Figure 3.2 a bounded and an unbounded polyhedron are shown. A polyhedron is a convex set.



(a) (b)

Figure 3.2: A bounded polyhedron (a) and an unbounded polyhedron (b). Both are convex sets.

3.2.3 Extreme points

An extreme point of a convex set $\mathcal{P}$ is a point in $\mathcal{P}$ which does not lie on any open line segment joining two points in $\mathcal{P}$. In Figure 3.2(a), the extreme points are the corners of the polyhedron, i.e., the vertices of the convex set.

3.3 Linear Programming

When an optimization problem has a linear objective function to be optimized over a polyhedral feasible region it is called a linear programming (LP) problem. When applying an LP model to a real world problem certain properties of the real world problem need to be fulfilled. These properties, which are listed below, are axioms that an LP model is based on [3].

- **Proportionality:** This means no economies of scale. For example; if a warehouse is associated with a storage cost defined per unit, then the total storage cost doubles if the number of units in the warehouse is doubled. Proportionality implies that the units are independent of each other both in quantity and over time. Further, proportionality means that an equation is of first order, i.e., linear.
- **Additivity:** The problem do not contain any fixed costs. Using the same warehousing example as above; no fixed costs means that if one unit is removed from the warehouse the total storage cost will decrease with the cost of one unit.
- **Divisibility:** The problem needs to be continuous and no integer properties are allowed. In the real world this is unfortunately seldom the case since many objects can not be divided. This indicates that integer solutions to an optimization model are preferable because continuous ones do not make any sense in a real world point of view.
- **Determinism:** No randomness is allowed in the problem. Randomness is a very common property in real life and simplifications of the problem is necessary if an LP model is

chosen.

Many of these properties are not very common in real world scenarios. An optimization model is a model of the real world and simplifications might be needed when formulating the model. In the best case scenario the real world is linear in nature with no or very few simplifications required.

3.3.1 Formulating a Linear Optimization model

All linear problems can be expressed in standard form which is formulated as

$$\begin{aligned} \text{minimize} \quad & z = \boldsymbol{c}^T \boldsymbol{x}, \\ \text{subject to} \quad & \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \\ & \boldsymbol{x} \geqslant \boldsymbol{0}^n, \end{aligned} \tag{3.1}$$

where $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, $\mathbf{b} \geqslant \mathbf{0}^m$, and $\boldsymbol{c} \in \mathbb{R}^n$. If a problem is not formulated in this way it can be transformed into the standard form using existing methods [3]. When implementing an LP model in AMPL it is not necessary to write it in standard form because CPLEX makes this transformation before solving the model.

To optimize is to find the best way of doing something given certain constraints. In standard form this objective is to minimize the objective function denoted by z. The objective function is a function of $\boldsymbol{x}$, which is the variable sought. If the optimal value of this variable is found a solution of how to do something in the best way is obtained. In the objective function $\boldsymbol{x}$ is multiplied by a vector $\boldsymbol{c}$ which is called the cost vector. Cost vector is just a name of the vector in the standard form and does not need to be a vector containing costs in the real world problem. When calling $\boldsymbol{c}$ a cost vector one can say that the objective is to minimize the cost. If the real problem is a maximization problem the objective function is simply multiplied by minus one and a maximization problem has then turned into a minimization problem.

The second row of an LP in standard form describes the constraints that the objective function is subjected to. The constraints are, as the word implies, something that prevents the variables of the objective function from obtaining certain values. These constraints constitutes a mathematical formulation of the constraints that exists in the real world problem. It could for example be capacity or demand constraints, logical constraints, or other constraints that makes certain values of $\boldsymbol{x}$ infeasible.

The last line of an LP in standard form is a constraint forcing the values of a variable $\boldsymbol{x}$ to be greater than or equal to zero. If the variable in the real world is allowed to take negative values it can still be written in standard form using certain transformation [3].

3.3.2 Existence and properties of optimal solutions

It is advantageous to work with LP models since they have very nice properties that makes the search for optimum much easier. One very important property is defined below.

Definition 3. *Let* $\mathcal{P} := \{\boldsymbol{x} \in \mathbb{R}^n \,|\, \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}; \boldsymbol{x} \geqslant \boldsymbol{0}^n\}$ *and consider the linear program*

$$\begin{aligned} &\text{minimize} \quad && z = \boldsymbol{c}^T\boldsymbol{x}, \\ &\text{subject to} \quad && \boldsymbol{x} \in \mathcal{P}. \end{aligned}$$

This problem has a finite optimal solution if and only if $\mathcal{P}$ *is nonempty and* z *is lower bounded on* $\mathcal{P}$*. If the problem has a finite optimal solution, then there exists an optimal solution among the extreme points.*

Definition 3 states that the optimum will be found among the extreme points. This decreases the number of possibly optimal points drastically compared to having to evaluate every point in the set. So how can these extreme points be found? One method developed is called the simplex method. In practice, the simplex method solves most LP problems in polynomial time, but there do exist problems that the simplex method solves in exponential time, see Section 3.6. Solving problems in exponential time is a rather poor behavior but that is fortunately happening rarely when using the simplex method [12]. Simplex uses information obtained from the problem to decide which extreme point to evaluate. It is an iterative process in which the method searches through extreme points and terminates when an optimal extreme point is found [3]. Further explanation of the simplex method is found in "An Introduction to Continuous Optimization" by Andréasson et al. [3], or in "Optimization" by Lundgren et al. [4]. The simplex method is one of the methods that are implemented in the optimization solver CPLEX.

3.3.3 Example of a simple LP model

In Figure 3.3 three warehouses are situated at the nodes WH 1, WH 2, and WH 3. Each of these warehouses has a demand of d_j, $j = 1, 2, 3$ of a product. Two distribution centers are located at the node DC 1 and DC 2. The number of products that each distribution center i is sending to each warehouse j is denoted by $\boldsymbol{x}_{ij}$. There are costs associated with transporting the products and these costs are denoted by $\boldsymbol{c}_{ij}$. In order to find the cheapest way to fulfill the warehouses' demand the LP model to

$$\begin{aligned} &\text{minimize} \quad && z = \sum_{i=1}^{2}\sum_{j=1}^{3} \boldsymbol{c}_{ij}\boldsymbol{x}_{ij}, \\ &\text{subject to} \quad && \sum_{i=1}^{2} \boldsymbol{x}_{ij} \geqslant \boldsymbol{d}_j, \quad \text{j = 1, 2, 3}, \\ & && \boldsymbol{x} \geqslant \boldsymbol{0}^n, \end{aligned} \tag{3.2}$$

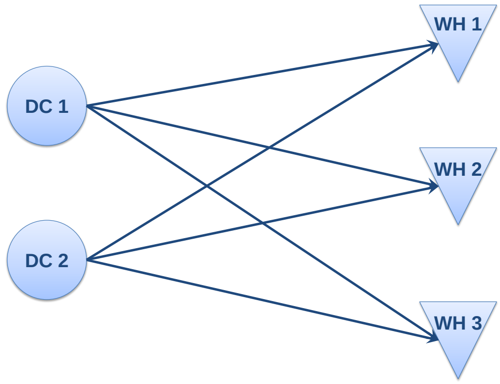


Figure 3.3: The network interpretation of a simple LP model where two distribution centers (DC) are to fulfill three warehouses' (WH) demand of a product at the lowest total cost.

is created. The objective function z represents the total cost of transporting all products, which is the sum of the product flows multiplied with their corresponding costs. In order to satisfy the warehouses' demand the sum of all products transported to each warehouse must be at least equal to the demand, which is denoted by the "greater than or equal" sign. The minimum cost solution will be found in one of the extreme points of the polyhedron defined by the constraints.

3.4 Mixed Integer Linear Programming: Branch and Bound

Sometimes when formulating an LP model it is necessary to add integer requirement on some variables in order to describe the real world problem in a proper way. This transforms the LP model into a Mixed Integer Linear Programming model (MILP). A MILP model contains one or more binary or integer variables and to solve a MILP model a method called Branch and Bound can be used [4].

The idea of Branch and Bound (B&B) is to find an efficient way to explore all possible solutions to a MILP problem. The method divides the feasible region, where the optimum is sought, into subregions which are explored such that it is possible to decrease the size of the problem [4]. In the worst case scenario the division does not lead to any size reduction and B&B is then no faster than a complete enumeration method. The division of the feasible region is graphically represented by a search tree. A search tree is a decision tree that has its root at the top and branches down; an example is shown in Figure 3.4.

The B&B method starts by solving an integrality and binary relaxed version of the original MILP problem to optimality. The relaxed version of the original problem is here assumed to be the MILP problem with the binary and integrality requirements removed. This transforms the MILP problem to an LP problem, which can be solved efficiently by using for example the simplex method. Relaxing MILP problems into efficiently solvable LP problems might reduce

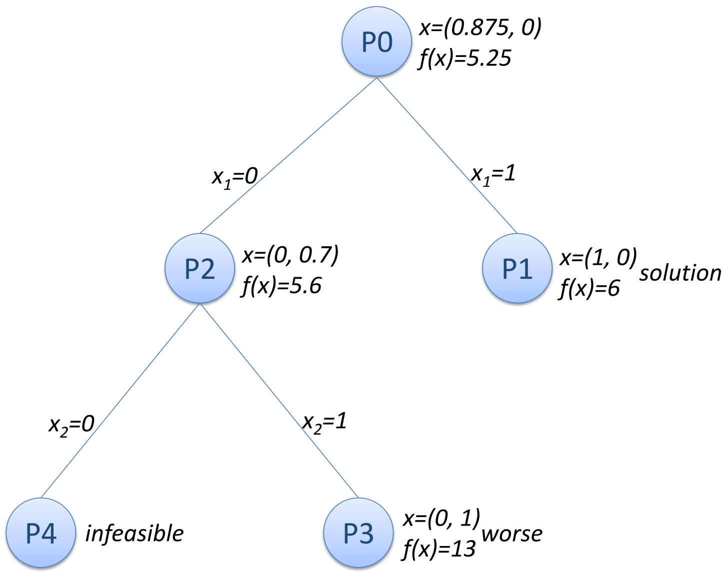


Figure 3.4: Branch and Bound divides the feasible region, where the optimum is sought, into subregions in order to find the optimum in a more efficient way. This can be visualized as a search tree.

the size of the search tree when combining it with a pruning technique described in this section, and these procedures are the strength of the B&B method. The optimal value obtained by solving the relaxed problem to optimum gives an optimistic bound to the optimal value of the original MILP problem's objective function. An optimistic bound means that the optimum of the original MILP problem can never be lower than this value, given a minimization problem. This is true since the relaxed problem is defined over a larger set and might then be able to find a better optimal value than what can be found using the set defined in the MILP problem. The optimum of the relaxed MILP problem represents the root in the search tree, which is shown as node P0 in Figure 3.4. The branches are the subregions that the feasible region of the original MILP problem is divided into. The subregions are created in such a way that they will represent the binary or integrality requirements of the MILP problem. In Figure 3.4 the search tree of the MILP to

$$\begin{aligned} \text{minimize} \quad & 6x_1 + 7x_2 \\ \text{s.t.} \quad & 8x_1 + 10x_2 \geqslant 7 \\ & x_1, x_2 \quad \text{binary}, \end{aligned} \tag{3.3}$$

is illustrated. Next to the root node P0 the optimal solution found to the relaxed MILP problem is indicated. The optimal value of the objective function is $f(\boldsymbol{x}) = 5.25$ and that objective value is obtained by the solution $x_1 = 0.875$ and $x_2 = 0$. The next step after finding the optimum of the completely relaxed MILP problem is to structure the next level of the tree which is represented by the branches P1 and P2 originating at the root node P0. The branches P1 and P2 are representing the case when the binary variable x_1 is set to 1 and 0, respectively. The branches P1 and P2 are evaluated by solving a relaxed version of the MILP problem with x_1 fixed to 1 or 0. This results in two new optima, one for each relaxed problem. When x_1 is set to 0 the optimal solution is $x = (0, 0.7)$ which results in the optimal value $f(\boldsymbol{x}) = 5.6$ and when $x_1 = 1$ the optimal value is $f(\boldsymbol{x}) = 6$ for $x = (1, 0)$. If no feasible solution of the MILP is found in this step another branching will take place from the existing branches. In this example, however, the obtained optimum of the relaxed MILP is feasible at one of the branches, namely P1. This means that no more branching is needed from that branch since the solution obtained is the best possible in that part of the tree. Stopping the branching from an existing branch is called "pruning" the tree and B&B has in this way managed to decrease the tree size.

The obtained optimal value of P1 is a pessimistic bound to the optimal value of the original MILP. From the optima of P0 and P1 a conclusion can be drawn that the optimal value of the objective function to the original MILP is in the interval $[5.25, 6]$, where the lowest value of the interval is called a lower bound (LBD) and the largest value is called upper bound (UBD).

The branch P2 has a relaxed solution which is lower than what was obtained at P1. This means that there is a possibility to find better feasible solutions at branches originating at P2 than what was found at P1 and no pruning will occur in this part of the tree. Further branching is made at P2, the branch P3 representing $x_2 = 1$ and the branch P4 representing $x_2 = 0$. The obtained optimal value from solving the MILP at P3 is $f(\boldsymbol{x}) = 13$ where $x_1 = 0$ and $x_2 = 1$ whereas the problem is infeasible when $x_1 = 0$ and $x_2 = 0$. If a branch representing a relaxed MILP is infeasible in the original MILP problem there is no need for further branching since even the relaxed problem is infeasible. The tree is also pruned at those locations.

The optimal value of the objective functions of P3 is compared to the already found feasible solution at P1. The optimal value found at P3 is $f(\boldsymbol{x}) = 13$ which is worse than the one already found at P1, $f(\boldsymbol{x}) = 6$, and will therefore be discarded. Since no better solutions was obtained then the one already found, the upper bound will stay the same. Since there is no other branches to investigate in the tree a conclusion can be made that the upper bound, 6, is also the optimal value to the original MILP problem. If the problem would have been bigger and included more branches this procedure would continue until all branches were investigated, either direct or indirect by using the pruning procedure. In the example in this chapter B&B was able to decrease the size of the search tree by 33 %, since 2 out of 6 branches could be cut of.

3.5 Divergency from optimum

Depending on the complexity of an optimization problem and the size of the problem instance to be solved, the time or computer resources needed to solve the problem to optimum might not be available. An alternative can be to not solve the problem to optimum but to accept a deviation from optimum where the solution obtained will be considered good enough [4]. How far from optimum a non-optimal solution is in the worst case can be calculated and is the gap between the verified upper and lower bound on the objective value that has been calculated. As mentioned in Section 3.4 the upper (UBD) and lower (LBD) bounds can be calculated during the B&B procedure. The optimum is confirmed to be within the bound defined by $[\text{LBD}, \text{UBD}]$. The relative gap is defined as

$$\text{relative gap} = \frac{\text{UBD} - \text{LBD}}{\text{LBD}}, \tag{3.4}$$

which gives information on how big the interval $[\text{LBD}, \text{UBD}]$ is in relation to the optimistic bound LBD. The gap is therefore a measurement on how close the value of an obtained feasible solution is to that of a optimal solution in worst case. When a solution algorithm used for finding optimum reaches a feasible solution within a predefined relative gap size, this gap can work as a termination criterion for the algorithm. The obtained solution from the solver is in that case a solution within the predefined gap size. The acceptable size of the gap depends on the optimization problem in question and is decided from case to case.

3.6 Complexity

The complexity of a problem is generally separated into two parts; the complexity of the algorithm used for solving the problem and the complexity of the problem per se. It is, however, the complexity inherent in the problem that governs the total complexity of the problem. Complexity is measured in the amount of resources needed to find a solution to a problem. The resources can either be time, computer capacity, or other physical restrictions [4].

The complexity of an algorithm is proportional to the computation time needed to perform the operations necessary to solve the problem, which in turn depends on the problem's input size. Therefore the complexity of a certain algorithm is calculated as a function of the input size n of the problem [13], where n can be the number of variables present or the number of constraints needed to formulate the mathematical model. How the complexity scales when n increases is the focus when studying complexity theory. This is because even an arbitrarily bad algorithm can perform well on small problems, but when the size of the problem increases the amount of required resources for solving the problem irrupts. To be able to compare different algorithms' complexity the notation O is used. The O-function expresses the upper bound on the number of operations needed to solve a problem in the worst-case. Different instances of the same size of a problem can be differently hard to solve and therefore a worst-case and an average-case of the complexity are introduced. If the complexity of an algorithm in the worst-case is upper bounded by a polynomial function of n, the algorithm prossesses a polynomial complexity and is in $O(n^c)$, where $c \geqslant 0$ is a constant [4]. Another category of algorithms is in $O(2^n)$ which means an exponential complexity. This is a very poor behavior. The simplex method, mentioned in Chapter 3.3.2, is an algorithm used for solving LP problems. The simplex method is in $O(2^n)$, but is still considered to be an efficient algorithm. The reason is that in average-case the simplex method is behaving like a polynomial algorithm, but there exists instances of LP problems for which the complexity is exponential [4].

It is possible to divide problems in to classes depending on their complexity. In order to do so it is necessary to make a transformation of the problem into a decision problem. A decision problem is a formal question where the only possible answer is yes-or-no [4]. A problem that can be transformed into a decision problem that is solvable in polynomial time is said to belong to the class **P**. A problem is solvable in polynomial time if it can be solved by an algorithm of polynomial complexity. This is a desirable property of a problem since this means that it can be solved efficiently [14]. LP problems belong to this class [12]. It is necessary to distinguish between practice and theory when it comes to complexity analysis. There exists an algorithm for solving LP problems in polynomial time, the ellipsoid algorithm, why it is possible to say that LP problems belong to **P**. This algorithm is however very inefficient in practice, while the simplex method performs much better [12].

Problems belonging to the class **NP** have a bisect definition; there exists a non-deterministic algorithm that in polynomial time can provide solution candidates to the problem, and the correctness of these solution candidates can be verified polynomial time. This means that if you are lucky and the first guess, or solution candidate, you come up with is the solution the problem is solvable in polynomial time. A class of problems that are harder to solve than **NP** are the class of **NP**-complete problems. For these problems there exists no polynomial algorithm to solve the decision problem, but if a solution was given its correctness could be verified in polynomial time. The only known algorithms for solving the problems of this class are running in exponential time [15]. This property makes large instances of the problem intractable and impossible to solve in acceptable time. The class of **NP**-complete problems contains many well-known problems of which the 0-1-knapsack and the traveling salesperson problems are two examples [16]. Even MILP belong to the **NP**-complete class [14].

4

SKF's Distribution Network

THE characteristics of SKF's products constituted the main reason for developing a returnable packaging system. The products are heavy and require a durable packaging material that can resist the weight exposure. The returnable packaging material at SKF is named SKF Group Standard Pallet (GSP) and is the primarily used packaging material in SKF's distribution network. SKF's distribution network consists of a large number of nodes located across the world. These nodes can be divided into six different categories depending on the function of the nodes. The different categories are; Pallet Pool, GSP Production, SKF Factory, External Factory, SLS Warehouse, and Customer. The nodes in the network are linked together depending on their function and the type of flow present, and rules regarding this has been formulated to represent SKF's distribution network as good as possible. The costs related to the distribution network are dependent on handling, transport, and purchasing.

4.1 SKF's returnable packaging

The history of wooden packaging material at SKF extends far back in time and the remains of a carpentry workshop producing wooden packaging can still be found at SKF's grounds in Gothenburg today. The reasons behind the choice of packaging material were many. One of the reasons was that SKF's products are heavy and therefore require a strong and durable packaging material. Another reason is that it is necessary to protect the products during transportation and storage since they are sensitive to vibrations and external influences. Most of SKF's products are expensive which means that damage during transportation or storage would be very costly. The weight of the products played an important role when the returnable packaging material was developed at SKF. To be able to increase the fill rate of vehicles and warehouses it must be possible to stack the packages. This leads to a great weight exposure of the lowermost package which puts a lot of requirements on the packaging material's strength. Since SKF had to use very durable packaging to protect its products it also became possible to reuse them since they could last for longer time without being damaged. The reusing of packaging has then developed into the well-functioning returnable packaging system it is today.

The weight of the products also mattered when SKF chose to use a standard pallet with the size of half a EUR-pallet. If products would fill an entire EUR-pallet the load would become too heavy to be maneuvered by a manual forklift. Today, the main part of SKF's distribution network is designed to fit SKF's standard packaging size. Essentially all physical resources, including conveys, shelves, and loading areas in factories and warehouses, are designed to fit a pallet having the size of half a EUR-pallet.

During the years the strength of SKF's packaging material has been reviewed. It has turned out that the strength of the packaging material can be decreased and still protect the products satisfactory. SKF strives to decrease the strength of the packaging material in order to decrease the purchasing cost, but not to such an extent that it affects the reusability of the packaging.

The theory in Section 3.1 states that a returnable packaging system is in line with environmental policies and that the system's benefits increase when the size of the network increase. At SKF a decision about the use of returnable packaging has never been made for primarily environmental reasons. The decision has always been made with focus on SKF's ambition to satisfy its customers in an efficient way, but the set-up has turned out to fit the company's environmental policies very well. A major reason is that returnable packaging decrease the waste which is one of SKF's environmental goals. The returnable packaging also reduce the material consumption since the material can be reused. It does however increase the demand of transport and the emission associated with transport which contradicts SKF's environmental goals. SKF Logistics Services, which today is responsible for the returnable packaging, impose high requirements regarding environmental impact on their packaging suppliers. All wood in the packaging must be FSC approved, which is a certification of responsible forestry given by Forest Stewardship Council. The wood must be treated to resist varmints according to ISPM15, which is an international standard that facilitate international movement of the packaging. This treatment can either be done by using chemicals or heat. SLS is however requiring their suppliers of packaging to treat the wood only with heat since this lowers the environmental impact of the process.

In addition to SKF units external suppliers to SKF can also use the returnable packaging when sending products to SKF. SLS puts a lot of effort into increasing the number of external suppliers that use SKF's returnable packaging. The reason for this is that the total cost analysis has shown that the main area of savings when using a returnable system of packaging is the increased handling efficiency due to standardized items. The idea is that – where suitable – products entering, leaving, or being stored at SKF units shall be loaded in SKF's returnable packaging in order to utilize the system's full potential and increase the efficiency of the goods handling. For example if a component supplier packs its components in SKF's returnable packaging there is no need for the receiving SKF unit to repack them during goods reception, which leans the handling processes.

Most of SKF's customers are of Original Equipment Manufacturer (OEM) character, which means that the customers use SKF's products as components in their end products. Since SKF's products are not visible in the end product there are few branding opportunities for SKF. A synergy effect of using returnable packaging is that it gives SKF an opportunity for branding since all packaging are marked with the SKF logo.

4.2 SKF Group Standard Pallet

SLS's returnable packaging is called SKF Group Standard Pallets (GSP); see Figure 4.1. These GSPs are composed by three types of GSP parts; pallets, collars, and lids; see Figure 4.2. The most common GSP configuration used at SKF is when one pallet, two collars and one lid are assembled and this is what the name "Standard" comes from. A GSP can however be composed of anything between a single pallet up to the very tall composition of one pallet, nine collars and one lid. The common factor among all the different GSP configurations is that everyone is made up of one, and only one pallet. If a GSP is composed of collars, a lid is always added on top to keep the GSP steady and stackable. When the collars are not in use they are folded to minimize their volume. A GSP can serve as both primary and secondary packaging. In a primary packaging the products are placed directly in the packaging, whereas secondary packaging contains smaller packagings, usually made of cardboard – which in turn contain products.

Figure 4.1: An SKF Group Standard Pallet (GSP), comprising one pallet, two collars, and one lid, that are assembled. This is the most common configuration of a GSP at SKF.

(a) (b) (c)

Figure 4.2: A pallet (a), a collar (b), and a lid (c). These parts are used when assembling a GSP.

4.2.1 Different statuses of GSP parts

To be able to differ between GSP parts used in an assembled GSP and GSP parts not currently used, the parts are considered to have different statuses. GSP parts can have three different statuses and an example of flows of GSP parts with different statuses in SKF's distribution network is visualized in Figure 4.3.

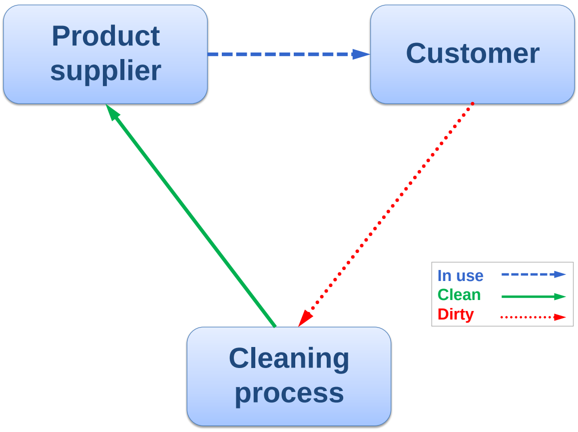


Figure 4.3: An example of a very simple distribution network. The blue dashed arrow represents a flow of GSP parts having the status "in use", the green solid arrow represents a flow of GSP parts with the status "clean", and the red dotted arrow represents a flow of GSP parts with the status "dirty".

The first status is "in use". When a GSP part has this status it means that this part is used in an assembled GSP. Any flow of GSP parts which has the status "in use" are visualized as a blue dashed arrow when describing SKF's distribution network. Using the flow of GSP parts instead of the flow of assembled GSPs enables exact counting of GSP parts in the network. This is important since there exist many different GSP configurations and there is no way to know how many GSP parts exist in the network if only assembled GSPs were counted.

When GSP parts are not in use they can either have the status "clean" or "dirty". Before a GSP part can be used in an assembled GSP it needs to be cleaned. With clean means no labels, staples, or deformations. Any flow of GSP parts with the status "clean" are visualized as green solid arrows. GSP parts that is neither in use nor clean has the status dirty. The status dirty means that the GSP part needs to be cleaned before it can be a part of an assembled GSP. Any flow of GSP parts with the status "dirty" are visualized as red dotted arrows in SKF's distribution network.

4.2.2 Piles

Clean GSP parts are bundled into piles before transported in the network. A pile consisting of one type of GSP part where the GSP parts has been folded, in the case of collars, and stacked

(a) (b) (c)

Figure 4.4: A pile of bases (a), collars (b), and lids (c). GSP parts are bundled into piles in order to facilitate the transport.

together. Piles of bases, collars, and lids respectively, are shown in Figure 4.4. The pile sizes varies between the different GSP parts and has been dimensioned to utilize the volume of a trailer or a container in an efficient way.

The piles are sent by truck to the receiver if the receiver is within a reasonable distance from the sender. If the receiver is situated overseas the GSP parts are shipped by sea freight. GSP parts can be transported by air freight if it is very urgent. This happens, however, on very rare occasions and is not something SLS wants to do considering the cost and the environmental impact involved.

4.3 Nodes

The six different categories of nodes mentioned in the introduction to this chapter have different functions within SKF's distribution network. The nodes and their functions are explained in this section.

4.3.1 Pallet Pool

Since GSPs are used for packing products at SKF units around the world, a well-managed distribution of GSP parts is necessary. Nodes whose only purpose is to take care of the distribution of clean and dirty GSP parts are called Pallet Pools. The Pallet Pools function as pools for clean GSP parts and are responsible for supplying all units within the network with the needed amount of clean GSP parts. The Pallet Pools are therefore the main stock keeper of clean GSP parts in SKF's distribution network. The Pallet Pools are located strategically around the world.

The GSP parts are reusable and therefore in the need for occasional cleaning. The cleaning of dirty GSP parts is taken care of by the Pallet Pools. Some Pallet Pools take care of this themselves and some outsource the activity to companies specializing in cleaning. During the cleaning process damaged GSP parts are either repaired or scrapped. The dirty GSP parts are added to the stock of clean GSP parts after the cleaning process is finished.

The scrapping, among other things, gives rise to losses in the system which lead to a need of refilling the system with new GSP parts. Another reason for the need of refilling is an increased demand of SKF's products. To fulfill their obligations, the Pallet Pools are therefore buying new GSP parts from GSP Production sites, which will be further described in the Section 4.3.2. The appropriate stock level at a Pallet Pool varies depending on the demand of clean GSP parts in the near future. The demand is forecasted by communicating with the supplied units and by taking overall industry changes into account. Each Pallet Pool is free to decide its own appropriate stock level, which is decided based on the number of weeks of demand that the Pallet Pool wants to keep in stock. A common stock level at a Pallet Pool is 2–6 weeks of future demand of clean GSP parts.

4.3.2 GSP Production

The flow of GSP parts within SKF's distribution network is not a closed flow; losses exist due to wear and tear, which result in scrapping, and the demand of GSP parts changes. Therefore, there is a need of suppliers within the network which are producing new GSP parts. These nodes in the network are called GSP Production. The GSP Production sites are chosen strategically to meet the demand of GSP parts where such is present. SKF Logistics Services has negotiated framework agreements with the GSP Production sites in which the prices are specified one year ahead and the transport details are settled. The agreements also include the smallest amount of GSP parts that SLS is undertaken to purchase during the agreement period. The agreements enables SLS to make call-offs and to have more or less instantaneous deliveries. A combination of multiple sourcing and long-term relationships gives SLS both barging power, benchmarking opportunities, and assured quality.

4.3.3 SKF Factory

All SKF's products and some components used in SKF's products are manufactured at SKF-owned factories situated on almost all continents. Nodes representing SKF-owned factories are called SKF Factory in the network. The factories have a demand of clean GSP parts since they are packing their finished products in GSPs.

4.3.4 External Factory

Some of SKF's external suppliers are packing components ordered by SKF units into GSPs. This increases the handling efficiency of the components when arriving at SKF units since all equipments are adapted to handle GSPs. The external factories using GSPs are called External Factories in SKF's distribution network.

4.3.5 SLS Warehouse

Products manufactured and assembled in the network are being stored in warehouses and these nodes are called SLS Warehouses. All SLS Warehouses and Pallet Pools located in the same country are situated very close to each other. This is due to historical reasons when the returnable packaging system developed and the Pallet Pools' locations were chosen. The locations of SLS Warehouses are strategically chosen to serve the customers in the best way.

4.3.6 Customer

The last category of nodes are Customers. SKF's Customers are spread all over the world and order products from SKF which are delivered to them packed in GSPs.

4.4 Links

Each node in the network is linked to other nodes depending on the function of the node. In this section the outbound and inbound flow of GSP parts at nodes of different categories are presented.

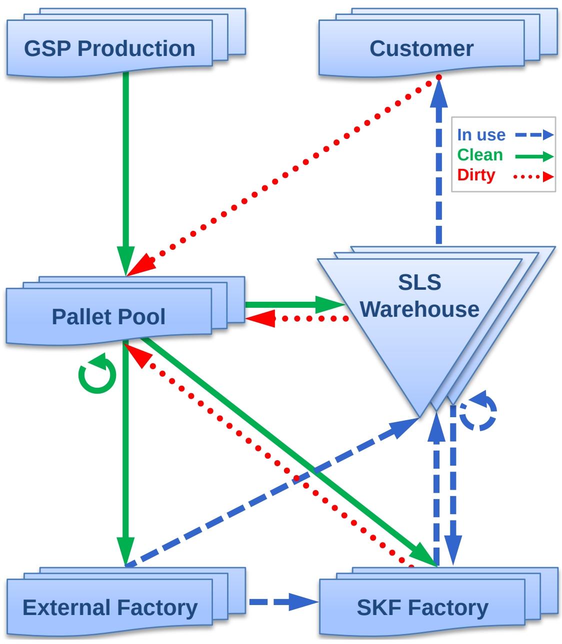


Figure 4.5: SKF's distribution network of GSP parts.

4.4.1 Pallet Pool

Pallet Pools supply SKF Factories, External Factories, SLS Warehouses, and other Pallet Pools with clean GSP parts. This is represented by green solid arrows in the distribution network shown in Figure 4.5. Since shortage and overflow of GSP parts can occur at Pallet Pools it is possible to send GSP parts between the Pallet Pools. This is represented by a green solid arrow forming a circle situated next to the Pallet Pool.

Not all Pallet Pools are supplying all SKF Factories, External Factories, and SLS Warehouses. Nodes situated in the same country as a certain Pallet Pool always get their supply of clean GSP parts from that Pallet Pool. The GSP parts are in that case sent in quantities of piles. How many piles that are sent depends on the demand of GSP parts at the receiving node. SKF Factories and SLS Warehouses not having a domestic Pallet Pool are supplied by any Pallet Pool located on the same continent. The transportation of GSP parts is then carried out in quantities of full loads. This is because SKF wants to minimize the transportation cost and environmental impact involved when transporting longer distances. External Factories are, however, always supplied by a predetermined Pallet Pool since it is important to keep a good relation with those factories; changing the supplying Pallet Pool often is then not a good idea. The GSP parts are always sent pile by pile to External Factories. When Pallet Pools send clean GSP parts to other Pallet Pools it is carried out in full loads. A summary of the transportation rules are presented in Table 4.1. The abbreviation D stands for domestic and I stands for international.

Table 4.1: Rules for transportation of clean GSP parts from Pallet Pools.

	Pallet Pool	GSP Prod.	SKF Factory		Ext. Factory	SLS Wh.		Cust.
			D	I		D	I	
Pallet Pool	full load	-	pile	full load	pile	pile	full load	-

The transportation of GSP parts from a Pallet Pool is planned according to a schedule determined by a forecasted demand of GSP parts at each destination node. The schedule is created for a few weeks at a time. If the demand of GSP parts happens to be greater or lower than what is scheduled, the amount transported is adjusted to meet the revealed demand. Knowing how many GSP parts that should be transported helps the Pallet Pools forecasting their appropriate stock level.

4.4.2 GSP Production

SLS has decided to work in a partnership like relation with the suppliers since it has discovered that a close cooperation increases both the quality of the products and the innovation rate. The selection of suppliers is based on a variety of conditions. GSP parts are low-valued items which makes the transport cost an essential part of the purchasing cost. It is therefore important that the GSP Production sites are located close to the Pallet Pools that they are supplying in order to minimize the transport cost. To decrease the risk associated with purchasing from only one supplier due to price risings, delivery problems, and scarcity, SLS has decided to increase the supplier base to at least two GSP Production sites for each Pallet Pool that purchase new GSP parts.

When transporting GSP parts from GSP Production sites, full truck loads of only one type of GSP parts is used. The flow of new GSP parts from GSP Production sites to Pallet Pools is illustrated in Figure 4.5 as green solid arrows pointing from GSP Production to Pallet Pool.

4.4.3 SKF Factory

SKF's products manufactured and assembled at SKF Factories are stored at SLS Warehouses. The flow of GSP parts in use that this gives rise to is represented by a blue dashed arrow pointing from SKF Factory to SLS Warehouse in Figure 4.5. If the SKF Factories need components, they are either being supplied by SLS Warehouses or External Factories; this is indicated by blue dashed arrows pointing towards SKF Factory.

SKF Factories receive components packed in GSPs, which after unpacking will be empty. At SKF Factories situated in countries with no Pallet Pool any labels on the GSP parts are removed and the GSP parts are used for assembling GSPs used in their outbound flows. By doing so the factories are changing the GSP parts status from dirty to clean. This means that these SKF Factories do not have to send dirty GSP parts for cleaning all the time, which would be a very costly and non-environmental friendly set-up. If, however, the SKF Factories have a domestic Pallet Pool they send all dirty GSP parts for cleaning to a predetermined Pallet Pool and this is carried out part by part, no requirements to bundling the GSP parts is present. Flows of dirty GSP parts from SKF Factory to Pallet Pool is presented in Figure 4.5 as a red dotted arrow between these nodes.

Since the deliveries of clean GSP parts to SKF Factories are performed in discrete entities, either piles or full loads, SKF Factories need to be able to store redundant GSP parts that are not needed in the immediate goods handling. This together with the fact that some SKF Factories clean most GSP parts themselves, mean that SKF Factories keep a stock of clean GSP parts. The Pallet Pools still have the main stock function in the network, but small stocks do occur at SKF Factories as well. The stock levels at SKF Factories having a domestic Pallet Pool are very low since they send all dirty GSP parts for cleaning and the only stock present occurs because the clean GSP parts arrive in discrete piles. The stock at SKF Factories with no domestic Pallet Pool may, however, be larger. The level should still not be big, but the stock at these factories are larger because of the fact that they do clean most GSP parts themselves. If, however, the stock of GSP parts at these SKF Factories become too large, they will send a full truck load with dirty GSP parts to a predetermined Pallet Pool. Table 4.2 summarizes these transportation rules of dirty GSP parts and is presented at the end of Section 4.4.

4.4.4 External Factory

The transportation of clean GSP parts to External Factories is carried out pile by pile which means that External Factories are also keeping a small stock for handling redundant GSP parts. External Factories are supplying SLS Warehouses and SKF Factories with components which are seen as blue dashed arrows from External Factory in Figure 4.5. Since they are suppliers of components to SKF they do not receive any SKF products themselves. Therefore, there is no blue dashed arrow pointing towards External Factory.

4.4.5 SLS Warehouse

As for SKF Factories and External Factories, SLS Warehouses receive GSP parts supplied in discrete sizes and a small stock of GSP parts is therefore present. If the warehouses are situated in the same country as a Pallet Pool the dirty GSP parts are sent part by part to that Pallet Pool for cleaning. If, on the other hand, the SLS Warehouses do not have a domestic Pallet Pool the warehouses reuse dirty GSP parts right away without sending them for cleaning, just as for SKF Factories. If the stock of GSP parts becomes too large a full truck load of dirty GSP parts are sent to a predetermined Pallet Pool for cleaning. This is illustrated in Figure 4.5 as a red dotted arrow pointing from SLS Warehouse to Pallet Pool.

Besides receiving products from External Factories and SKF Factories, SLS Warehouses can send products amongst themselves, which is illustrated by a blue dashed arrow forming a circle placed at SLS Warehouse. The reason for this is to transfer products closer to the market demanding them.

4.4.6 Customer

Customers place their orders at sales units which act as intermediaries, while SLS Warehouses are responsible for the delivery of the ordered products. An intermediary is an actor that assists a company to reach its customers. This means that the material flows origin at SLS Warehouses and end at Customers, whereas the monetary flows origin at Customers, pass through sales units, and end at SLS Warehouses. The distribution of products to Customers is represented by a blue dahsed arrow pointing from SLS Warehouse to Customer in Figure 4.5. The Customers pay a deposit for the GSPs they receive to the sales unit at which they have placed their order. The deposit is returned from the same sales unit if the Customers return the GSPs to a Pallet Pool. If Customers are not returning the received GSPs, losses occur in the distribution system. The return rate of GSPs from Customers varies a lot depending on which country they are situated in. Some of the reasons for the differences in return rate might be cultural differences, how the Customer values the GSP in comparison to the deposit, and the effort the sales units put into campaigns for getting the GSP parts returned. SLS has realized that the return rate is an important part of their returnable packaging system and has therefore initiated projects for increasing the return rates. In Figure 4.5 the flow of dirty GSP parts from Customer to Pallet Pool is illustrated as a red dotted arrow, and parts are always returned unbundled.

A summary of the transportation rules for dirty GSP parts to Pallet Pools is presented in Table 4.2.

Table 4.2: Rules for transportation of dirty GSP parts to Pallet Pools.

	Pallet Pool	
	Domestic	International
Pallet Pool	-	-
GSP Production	-	-
SKF Factory	part by part	full load
External Factory	-	-
SLS Warehouse	part by part	full load
Customer	part by part	part by part

4.5 Distribution Costs

The distribution of GSP parts is associated with costs and to be able to create a mathematical optimization model with the objective to minimize the cost, it is important to understand the cost related to distribution of the GSP parts. SLS charges the units using GSP parts in order to cover the additional costs that the returnable packaging system generates. The charge is divided into two parts; Returnable Deposit (RD) and Pallet Pool Handling Fee (PPHF). The RD works as an incentive for customers to return the GSP parts that they have received, which is crucial for a returnable packaging system's functionality. According to the theory in Section 3.1, a returnable deposit is something that considerably increases the return rates. The purpose of the PPHFs is to cover the costs associated with cleaning and handling. These costs arise first when a returnable packaging system is introduced and if the costs can be covered by a direct charge it will increase the margins of the system.

The price of transport depends on the weight and volume of the goods transported. SLS is a large buyer of transportation services why it, through partnerships with the transport providers, has lowered its total cost for the services. The main task of SLS is to deliver products to SKF's end customers and between SKF's facilities and these links are therefore centrally priced.

5

The Optimization Model

KNOWING how SKF's distribution network is designed is important when creating a mathematical optimization model in order to make the right simplifications and create a model that is as conform as possible with the real network. The design of the network described in Chapter 4 is in this chapter mathematically expressed and translated into constraints that concern each node category and the links between them. The objective function reflects the purpose of the project and is also presented here.

Some variables, parameters, sets, and indices used in the model occur very often. In order to not be too repetitive they are introduced in this section and will not be defined any further in the constraints they occur. A summary of these components are presented in Table 5.1. Other variables, parameters, sets, and indices used are defined where they occur and are summarized in tables at the end of each section.

The set $\mathcal{T}$ contains the time steps that the model is defined over. For a model instance with 12 months the set will be $\mathcal{T} = \{1, 2, 3, \dots, 12\}$. Whenever an index t is used, it will be defined as $t \in \mathcal{T}$.

There are three different types of GSP parts used at SKF which are represented by the set $\mathcal{P} = \{\text{bases, collars, lids}\}$ and whenever an index p is used, it will be defined as $p \in \mathcal{P}$. The GSP parts can have three types of statuses; clean GSP parts, dirty GSP parts, and GSP parts in use which are represented by the set $\mathcal{S} = \{s^{\text{clean}}, s^{\text{dirty}}, s^{\text{in-use}}\}$.

Two of the most common variables are $x_{ijtps^{\text{clean}}}$ and $x_{ijtps^{\text{dirty}}}$. The variable $x_{ijtps^{\text{clean}}}$ is defining the amount of clean GSP parts p sent from node i to node j, at time t. The variable $x_{ijtps^{\text{dirty}}}$ is defined in the same way but are defining the amount of dirty GSP parts sent. In addition to these two variables are variables representing the amount of GSP parts kept in stock and how the stock is changing very common. The stock is represented by s_{itp} and the stock change is represented by s_{itp}^{change}, where i is the stock keeper, t the time step and p the type of GSP part the stock and stock change consists of.

Lastly, one parameter is defined in this section, $f_{ijtps^{\text{in-use}}}$. This parameter represent the amount of GSP parts in use p sent from node i to node j, at time t.

Table 5.1

Variables	
$x_{ijtps^{\text{clean}}}$	Amount of clean GSP parts p sent from node i to node j, at time t.
$x_{ijtps^{\text{dirty}}}$	Amount of dirty GSP parts p sent from node i to node j, at time t.
s_{itp}^{change}	The change in stock of GSP parts p, at node i, at time t.
s_{itp}	Stock of GSP parts p, at node i, at time t.
Parameters	
$f_{ijtps^{\text{in-use}}}$	Amount of GSP parts p with status "in use" sent from node i to node j at time t.
Sets	
$\mathcal{T}$	The time interval used in the model.
$\mathcal{P}$	The different GSP parts; {bases, collars, lids}.
$\mathcal{S}$	The different statuses of a GSP parts; $\{s^{\text{clean}}, s^{\text{dirty}}, s^{\text{in-use}}\}$
Indexes	
t	$\in \mathcal{T}$
p	$\in \mathcal{P}$
$s^{\text{clean}}, s^{\text{dirty}}, s^{\text{in-use}}$	$\in \mathcal{S}$

5.1 Pallet Pool

The Pallet Pools in SKF's distribution network function as hubs for clean and dirty GSP parts. In order to fulfill the demand of clean GSP parts at the nodes in the network a proper sized stock is necessary at each Pallet Pool. These stocks are updated by summarizing the flows of GSP parts at the Pallet Pools. This, and how the stocks are allowed to vary, is described mathematically as constraints. The transportation of clean GSP parts from Pallet Pools is arranged in a certain way and this is also restricted by constraints.

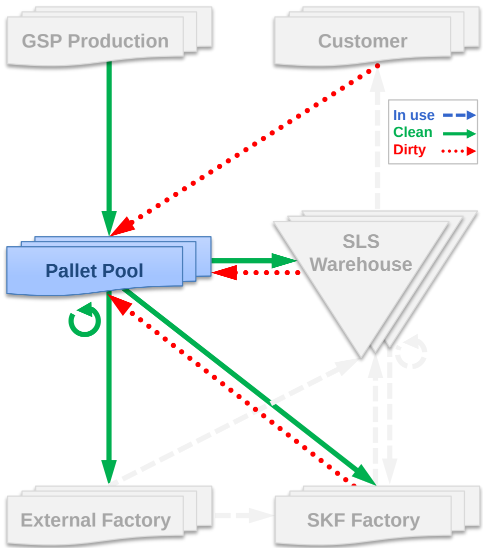


Figure 5.1: Flows of GSP parts present at Pallet Pools. Pallet Pools collect dirty GSP parts and by sorting, scraping, repairing, and cleaning, most GSP parts with status dirty are transformed into GSP parts with status clean and are added to the stock. If the stock of clean GSP parts at Pallet Pools is not large enough to fulfill the demand of clean GSP parts in the network, new GSP parts can either be purchased from GSP Production sites or be sent from other Pallet Pools.

5.1.1 Stock change constraint

The stock change s_{itp}^{change} at a Pallet Pool is a function of three different types of flows; inbound flow of clean and dirty GSP parts and outbound flow of clean GSP parts. These flows are presented in Figure 5.1.

The first type of flow is highlighted in Figure 5.2. Pallet Pools can receive clean GSP parts from two node categories; GSP Production sites and other Pallet Pools and which node that are sending to which Pallet Pool is described by a set of links, $\mathcal{M}^{\text{1a}}$. A mathematical description of this sum of flows is then formulated as

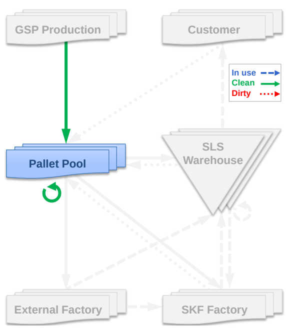


Figure 5.2

$$\sum_{(m,i)\in\mathcal{M}^{1a}} x_{mitps^{\text{clean}}}, \quad i \in \mathcal{I}^{1a},$$

where $\mathcal{I}^{1a}$ is a set containing all Pallet Pools. The expression describes the sum of the flows from GSP Production sites and Pallet Pools m to a certain Pallet Pool i, at time t for clean GSP parts p.

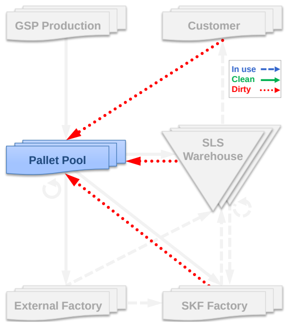


Figure 5.3

The inbound flow of dirty GSP parts at a Pallet Pool is presented in Figure 5.3. Nodes in the categories Customer, SLS Warehouse, and SKF Factory are sending dirty GSP parts to predetermined Pallet Pools and these links are gathered in a set, $\mathcal{N}^{1a}$. The flow of dirty GSP parts within the set $\mathcal{N}^{1a}$ is mathematically formulated as

$$\sum_{(n,i)\in\mathcal{N}^{1a}} q_{nitp} \cdot x_{nitps^{\text{dirty}}}, \quad i \in \mathcal{I}^{1a},$$

where the parameter q_{nitp} is a scrap rate of dirty GSP parts p at Pallet Pool i, which depends on the sending node n and varies over time. The scrap rate is a percentage of the amount of dirty GSP parts scraped when arriving at Pallet Pools.

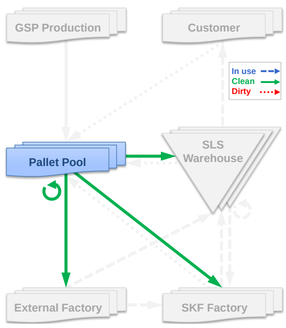


Figure 5.4

The last type of flow consists of clean GSP parts from Pallet Pools, which is highlighted in Figure 5.4. Four node categories have a demand for clean GSP parts; other Pallet Pools, External Factory, SKF Factory, and SLS Warehouse, and the links from Pallet Pools to these nodes form the set $\mathcal{O}^{1a}$. The flow of clean GSP parts from a Pallet Pool i within the set $\mathcal{I}^{1a}$ to a destination node in the set $(i,o) \in \mathcal{O}^{1a}$ is formulated as

$$\sum_{(i,o)\in\mathcal{O}^{a}} x_{iotps^{\text{clean}}}, \quad i \in \mathcal{I}^{1a}.$$

Stock change at Pallet Pools is the change in stock of clean GSP parts; yet it is a function of both clean and dirty GSP parts. This is because dirty GSP parts arriving at Pallet Pools are either scrapped or cleaned, which in the latter case means they are ready for use again. The stock change is the difference between inbound and outbound flow of dirty and clean GSP parts which is mathematical formulated as

$$s^{\text{change}}_{itp} = \sum_{(m,i)\in\mathcal{M}^{1a}} x_{mitps^{\text{clean}}} + \sum_{(n,i)\in\mathcal{N}^{1a}} q_{nitp} \cdot x_{nitps^{\text{dirty}}} - \sum_{(i,o)\in\mathcal{O}^{1a}} x_{iotps^{\text{clean}}}, \quad i \in \mathcal{I}^{1a}. \tag{5.1a}$$

The first sum is the inbound flow of clean GSP parts, the second sum is the inbound flow of dirty GSP parts and the last sum is the outbound flow of clean GSP parts.

5.1.2 Stock update constraint

When knowing the stock change at Pallet Pools, an updated stock for the next time step can be calculated. The updated stock is the current stock plus the stock change which is mathematically formulated as

$$s_{i(t+1)p} = s_{itp} + s_{itp}^{\text{change}}, \qquad i \in \mathcal{I}^{1\text{b}}, \tag{5.1b}$$

where $s_{i(t+1)p}$ is the updated stock and $\mathcal{I}^{1\text{b}}$ is a set containing all Pallet Pools.

5.1.3 Stock limit constraints

The size of the stock should neither be too small nor too big. If the stock is too small, there is a risk that the Pallet Pools will be unable to meet the demand for clean GSP parts. A safety stock is therefore required. If, on the other hand, the stock is too big, the risk of obsolete GSP parts increases and there might not be enough physical space at the Pallet Pools to accommodate all GSP parts. Moreover, there are always risks involved in having a large stock that ties up capital [17].

The stock limits need to be formulated mathematically in order to add them to the optimization model. If the stock limit is a function of the demand of clean GSP parts, as they are at the Pallet Pools, then the constraint representing the stock limit will not be linear. This is because the stock of clean GSP parts is a function of the stock change variable, which in turn is a function of the inbound and outbound flow of GSP parts, which are determined by the demand of clean GSP parts. This makes the stock a function of the demand of GSP parts, which in turn puts a requirement on its limits to not be functions of the demand if the model's linearity properties should be retained.

To solve this issue the stock limits are expressed as parameters instead of variables. The minimum stock limit parameter is represented by s_{ip}^{min} and the maximum stock limit parameter is represented by s_{ip}^{max}. The stock limit constraints are then formulated as

$$s_{itp} \geqslant s_{ip}^{\text{min}}, \qquad i \in \mathcal{I}^{1\text{c}}, \tag{5.1c}$$

$$s_{itp} \leqslant s_{ip}^{\text{max}}, \qquad i \in \mathcal{I}^{1\text{d}}, \tag{5.1d}$$

where $\mathcal{I}^{1\text{c}}$ and $\mathcal{I}^{1\text{d}}$ are sets containing all Pallet Pools. The inequalities state that the stock should not be smaller than the minimum stock level and not greater than the maximum stock level.

5.1.4 Transportation constraints

Clean GSP parts transported from Pallet Pools, see Figure 5.4, are either send in piles filling exactly one full load, or sent in piles with no limitations on how many piles sent each time, depending on the destination nodes. Nodes only receiving GSP parts which are filling a full load from Pallet Pools are summarized in a set $\mathcal{L}^{1\text{e}}$ of links between the sending and receiving node.

The transportation rules for clean GSP parts between different nodes are stated in Table 4.1.

In order to formulate the equation representing this transportation constraint, some additional parameters are needed. The first parameter determines how many GSP parts a pile consists of, denoted by b_{ip}^{pile}. The index i represents the Pallet Pool from where the pile is sent and p the type of GSP parts the pile consists of. This index is needed because different Pallet Pools have different pile standards which depend on the type of GSP parts. The second parameter is $b_{ijp}^{\text{full-load}}$ and states the number of piles that forms a full load. This parameter depends on the GSP part p, the Pallet Pool i, and the receiving node j, because the size of a full load is not the same for all links. Within Europe a full load size is an European trailer while overseas transportation defines a full load size of a shipping container.

The equation representing the transportation of full loads between nodes in the set $\mathcal{L}^{1\text{e}}$ is

$$x_{ijtps^{\text{clean}}} = g_{ip}^{\text{mix}} \cdot b_{ip}^{\text{pile}} \cdot b_{ijp}^{\text{full-load}} \cdot y_{ijts^{\text{clean}}}^{\text{full-load}}, \qquad (i,j) \in \mathcal{L}^{1\text{e}}, \tag{5.1e}$$

where g_{ip}^{mix} is a parameter representing the mixture of GSP parts in a full load, b_{ip}^{pile} is the number of parts in a pile, $b_{ijp}^{\text{full-load}}$ is the number of piles in a full load and $y_{ijts^{\text{clean}}}^{\text{full-load}}$ is an integer variable denoting the number of full loads of clean GSP parts sent from node i to node j, at time t.

A set $\mathcal{L}^{1\text{f}}$ represents links between Pallet Pools and receiving nodes that only receive GSP parts sent pile by pile. Which nodes this applies to is also stated in Table 4.1. The constraint representing this transportation rule is formulated as

$$x_{ijtps^{\text{clean}}} = b_{ip}^{\text{pile}} \cdot y_{ijtp}^{\text{pile}}, \qquad (i,j) \in \mathcal{L}^{1\text{f}}, \tag{5.1f}$$

where b_{ip}^{pile} is the number of GSP parts in a pile, y_{ijtp}^{pile} is an integer variable representing how many piles of clean GSP parts p sent from node i to node j at time t, and $\mathcal{L}^{1\text{f}}$ is a set of links of Pallet Pools and destination nodes that only receive GSP parts transported pile by pile.

A summary of additional variables, parameters and sets used in this section is listed in Table 5.2.

Table 5.2: Additional components used in constraint at Pallet Pools.

Variables	
$y^{\text{full-load}}_{ijts^{\text{clean}}}$	Number of full loads of clean GSP parts sent from node i to node j, at time t.
y^{pile}_{ijtp}	Number of piles of clean GSP parts p sent from node i to node j, at time t.
Parameter	
q_{ijtp}	Scrap rate of dirty GSP parts p at node j, depending on sending node i, at time t.
$s^{\min}_{ip}$	Minimum stock level of GSP parts p, at node i.
$s^{\max}_{ip}$	Maximum stock level of GSP parts p, at node i.
b^{pile}_{ip}	Number of GSP parts p in a pile originating from node i.
$b^{\text{full-load}}_{ijp}$	Number of piles of GSP parts p in a full load, from node i to node j.
g^{mix}_{ip}	Mix of GSP parts p in a full load sent from node i.
Sets	
$\mathcal{M}^{1a}$	Set of links between nodes sending clean GSP parts and receiving Pallet Pools.
$\mathcal{N}^{1a}$	Set of links between nodes sending dirty GSP parts and receiving Pallet Pools.
$\mathcal{O}^{1a}$	Set of links between Pallet Pools and nodes with a demand of clean GSP parts.
$\mathcal{I}^{1a}, \ldots, \mathcal{I}^{1d}$	Set of all Pallet Pools.
$\mathcal{L}^{1e}$	Set of links between Pallet Pools and destination nodes receiving GSP parts in full loads.
$\mathcal{L}^{1f}$	Set of links between Pallet Pools and destination nodes receiving GSP parts pile by pile.

5.2 GSP Production

If the demand of clean GSP parts exceeds the supply at Pallet Pools it is necessary to purchase new GSP parts from GSP Production sites. These flows are illustrated in Figure 5.5. The transport of GSP parts between GSP Production sites and Pallet Pools are regulated by a couple of constraints, which determine the load sizes and the minimum yearly delivery from each GSP supplier.

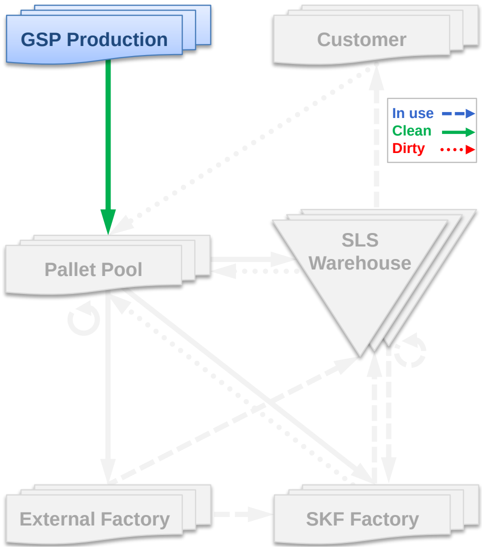


Figure 5.5: The flow of GSP parts present at GSP Production. The flow represents purchased GSP parts delivered to Pallet Pools.

5.2.1 Transportation constraint

The GSP parts are transported in full loads to the Pallet Pools and which GSP Production site that supplies which Pallet Pool is stated in a set $\mathcal{L}^{2a}$ containing these links. The constraint specifying that the transportation only occurs in full loads is

$$x_{ijtps^{\text{clean}}} = b^{\text{pile}}_{ip} \cdot b^{\text{full-load}}_{ijp} \cdot z_{ijtp}, \qquad (i,j) \in \mathcal{L}^{2a}, \tag{5.2a}$$

where b^{pile}_{ip} is the number of GSP parts in a pile, $b^{\text{full-load}}_{ijp}$ is the number of piles in a full load and z_{ijtp} is an integer variable representing the number of full loads of new GSP parts p send from GSP Production site i to Pallet Pool j, at time t.

5.2.2 Delivery constraint

To keep a good relation with GSP Production sites, SLS has determined a lower limit on the delivery of GSP parts from a GSP Production site each year. This constraint is mathematically formulated as

$$\sum_{(i,m)\in\mathcal{M}^{2b}} \sum_{t\in\mathcal{T}} x_{imtps^{\text{clean}}} \geqslant d_{ip}^{\text{min}}, \qquad i \in \mathcal{I}^{2b}, \tag{5.2b}$$

where $\mathcal{M}^{2b}$ is a set of links between GSP Production sites and receiving Pallet Pools, $\mathcal{I}^{2b}$ is a set containing all GSP Production sites and d_{ip}^{min} is the minimum amount of new GSP parts delivered each time period from GSP Production site i.

The additional variables, parameters and sets used in this section are listed in Table 5.3.

Table 5.3: Additional components used in constraints at GSP Production.

Variables	
z_{ijtp}	Number of full loads of new GSP parts p sent from node i to node j, at time t.
Parameters	
b_{ip}^{pile}	Number of GSP parts p in a pile originating from node i.
$b_{ijp}^{\text{full-load}}$	Number of piles of GSP parts p in a full load, from node i to node j.
d_{ip}^{min}	Minimum amount of new GSP parts p delivered each year from node i.
Sets	
$\mathcal{L}^{2a} = \mathcal{M}^{2b}$	Set of links between GSP Production sites and receiving Pallet Pools.
$\mathcal{I}^{2b}$	Set of all GSP Production sites.

5.3 SKF Factory

SKF Factories have no ability to store products and are therefore sending them to SLS Warehouses. This gives rise to either a demand or an abundance of GSP parts depending on the inbound and outbound flows of products packed in GSPs. How the stock is updated and its stock levels are regulated by constraints described in this section. Also the transportation rules of clean and dirty GSP parts are mathematically formulated.

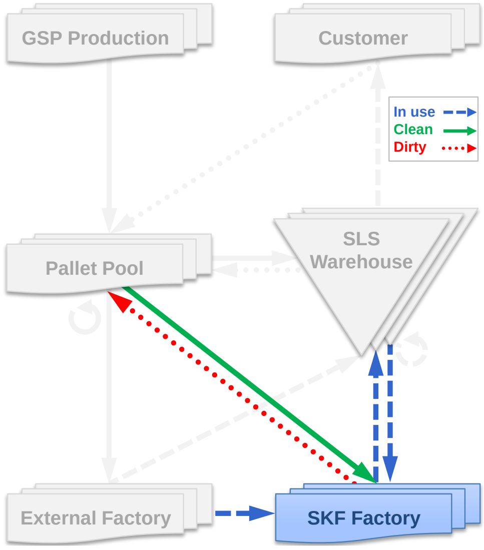


Figure 5.6: Flows of GSP parts at SKF Factories. The facts that all SKF Factories lack storage possibilities and use GSPs as packaging result in two different kinds of GSP flows; flows of GSP parts in use to SLS Warehouses and flows of clean and dirty GSP parts between Pallet Pools and SKF Factories.

5.3.1 Stock change constraint

The stock change at a SKF Factory is a function of four different types of flows; inbound flows of GSP parts in use and clean GSP parts, and outbound flows of GSP parts in use and dirty GSP parts, which all are represented in Figure 5.6.

SKF Factories are manufacturing products which are stored at SLS Warehouses. The products are transported in GSPs to the warehouses and the transportation results in a flow of GSP parts in use. This flow is highlighted in Figure 5.7. The flow of GSP parts in use is mathematically represented by

$$\sum_{m \in \mathcal{M}^{3a}} f_{imtps^{\text{in-use}}}, \qquad i \in \mathcal{I}^{3a},$$

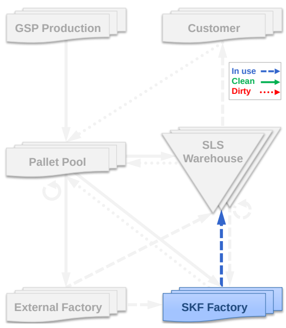


Figure 5.7

where $\mathcal{M}^{3a}$ is a set of all SLS Warehouses and $\mathcal{I}^{3a}$ is a set of all SKF Factories.

There are two different node categories sending components to SKF Factories; External Factories and SLS Warehouses which are defining a set $\mathcal{N}^{3a}$. The components are transported in GSPs which result in flows of GSP parts in use and is shown in Figure 5.8. The inbound flow of GSP parts in use sent from nodes in set $\mathcal{N}^{3a}$ is formulated as

$$\sum_{n \in \mathcal{N}^{3a}} f_{nitps^{\text{in-use}}}, \qquad i \in \mathcal{I}^{3a}.$$

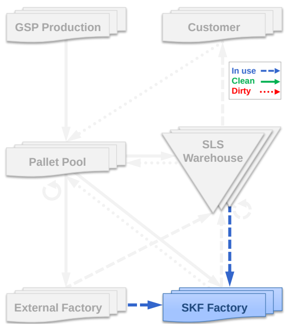


Figure 5.8

Pallet Pools are supplying SKF Factories with clean GSP parts, which is illustrated in Figure 5.9. Which Pallet Pool that is supplying which SKF Factory is presented in Section 4.4.1. The flow of clean GSP parts from Pallet Pools to SKF Factories is formulated as

$$\sum_{(o,i) \in \mathcal{O}^{3a}} x_{oitps^{\text{clean}}}, \qquad i \in \mathcal{I}^{3a},$$

where $\mathcal{O}^{3a}$ is a set that contains links between Pallet Pools and the SKF Factories that they supply.

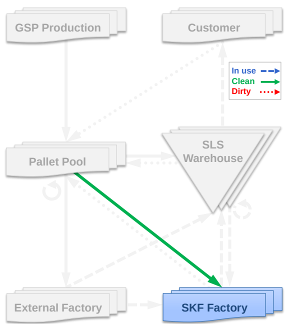


Figure 5.9

SKF Factories can send dirty GSP parts to Pallet Pools and which Pallet Pool that is responsible for handling which SKF Factory's dirty GSP parts is stated in Table 4.4. These relations are represented by a set of links $\mathcal{L}^{3a}$ containing SKF Factories and their corresponding Pallet Pool. The flow of dirty GSP parts from SKF Factories, which is illustrated in Figure 5.10, is mathematically formulated as

$$x_{ijtps^{\text{dirty}}}, \qquad (i,j) \in \mathcal{L}^{3a}.$$

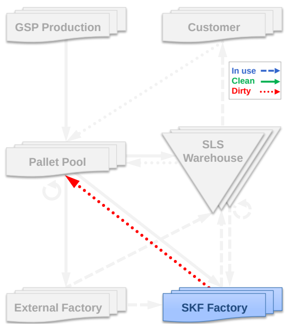


Figure 5.10

The stock change is the difference in inbound and outbound flows and the constraint representing this is

$$s_{itp}^{\text{change}} = -\sum_{m \in \mathcal{M}^{3a}} f_{imtps^{\text{in-use}}} + \sum_{n \in \mathcal{N}^{3a}} f_{nitps^{\text{in-use}}} + \sum_{(o,i) \in \mathcal{O}^{3a}} x_{oitps^{\text{clean}}} - x_{ijtps^{\text{dirty}}}, \qquad (i,j) \in \mathcal{L}^{3a}. \quad (5.3a)$$

5.3.2 Stock update constraint

A new stock can be calculated when knowing the stock change and this is performed in the same way as for the stock update at Pallet Pools;

$$s_{i(t+1)p} = s_{itp} + s_{itp}^{\text{change}}, \qquad i \in \mathcal{I}^{3b}, \tag{5.3b}$$

where $s_{i(t+1)p}$ is the updated stock at next time step and $\mathcal{I}^{3b}$ is a set containing all SKF Factories.

5.3.3 Stock limits constraints

The stock at SKF Factories should not be smaller or larger than a minimum or maximum stock level, respectively, and this constraint is formulated in the same way as for Pallet Pools;

$$s_{itp} \geqslant s_{ip}^{\min}, \qquad i \in \mathcal{I}^{3c}, \tag{5.3c}$$

$$s_{itp} \leqslant s_{ip}^{\max}, \qquad i \in \mathcal{I}^{3d}, \tag{5.3d}$$

where $\mathcal{I}^{3c}$ and $\mathcal{I}^{3d}$ are sets of all SKF Factories, $s_{ip}^{\min}$ is the stock minimum and $s_{ip}^{\max}$ is the stock maximum.

5.3.4 Transportation constraint

For SKF Factories not having a domestic Pallet Pool special rules apply for transportation of dirty GSP parts. It is quite costly to send GSP parts to Pallet Pools for cleaning if the factory is situated far away and GSP parts are therefore sent for cleaning only if the stock of GSP parts has become large enough to fill a full load. If the stock is not too large, the GSP parts arriving with status "in use" are cleaned at the factory instead of at a Pallet Pool and the GSP parts are ready for use again, now with the status "clean". Hence all abundant GSP parts at SKF Factories possess the status "clean" and are added to stock.

To figure out whether the stock is large enough to fill a full load a few constraints and two binary variables are needed, w_{ijtp}^{part} and w_{ijt}^{sum}. The binary variable w_{ijtp}^{part} is set to 1 if the stock of GSP part p is large enough to send a full load. This will give information about the stock of each type of GSP part. A full load can be sent only if the stock of each of the three parts is at least as large as what is needed to fill a full load and this is represented by the binary variable w_{ijt}^{sum} which is set to 1 if all three stock types are large enough.

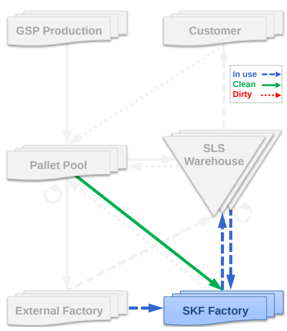


Figure 5.11

The stock at a SKF Factory when no flow of dirty GSP parts is present equal the current stock plus the stock change without the flow of dirty GSP parts, these flows are presented in Figure 5.11. Two constraints are needed to set w_{ijtp}^{part} to either 0 or 1 depending on whether the stock is smaller or larger than a number b_{ijp}^{needed}. The amount of each type of GSP parts that is needed to fill one full load is $b_{ijp}^{\text{needed}} = g_{ip}^{\text{mix}} \cdot b_{ip}^{\text{pile}} \cdot b_{ijp}^{\text{full-load}}$, were the parameters involved possess the same value as the corresponding parameters for clean GSP parts. The first constraints needed is mathematically formulated as

$$s_{itp} + s_{itp}^{\text{change}} - x_{ijtps^{\text{dirty}}} \leqslant b_{ijp}^{\text{needed}} + M^{3e} \cdot w_{ijtp}^{\text{part}}, \qquad (i,j) \in \mathcal{L}^{3e}, \tag{5.3e}$$

where the parameter M^{3e} is a large enough number and $\mathcal{L}^{3e}$ is a set of links between SKF Factory not having a domestic Pallet Pool and its corresponding Pallet Pool. This inequality constraint states that if the updated stock is larger than b_{ijp}^{needed}, then w_{ijtp}^{part} is forced to be 1 since the equation would be violated otherwise. When w_{ijtp}^{part} is set to 1 the inequality constraint states that the updated stock needs to be less than $b_{ijp}^{\text{needed}} + M^{3e}$, which is true also for stocks larger than b_{ijp}^{needed}. The large number M^{3e} is chosen in a way such that there is no risk of any updated stocks being larger than that number. It does not, however, need to be larger than the largest number that $s_{itp} + s_{itp}^{\text{change}} - x_{ijtps^{\text{dirty}}} - b_{ijp}^{\text{needed}}$ can take, which is calculated by using the restrictions on the stock and transportation capacities.

The second equation is

$$s_{itp} + s_{itp}^{\text{change}} - x_{ijtps^{\text{dirty}}} \geqslant b_{ijp}^{\text{needed}} - M^{3f} \cdot (1 - w_{ijtp}^{\text{part}}), \qquad (i,j) \in \mathcal{L}^{3f}, \tag{5.3f}$$

where the set $\mathcal{L}^{3f}$ is equivalent to the one used in the equation 5.3e and M^{3f} is equivalent to M^{3e}. This inequality constraint states that if the updated stock is smaller than b_{ijp}^{needed}, then w_{ijtp}^{part} is forced to be 0 since the equation would be violated otherwise. When w_{ijtp}^{part} is set to 0 the inequality constraint states that the updated stock needs to be larger than $b_{ijp}^{\text{needed}} - M^{3f}$ which is true for any updated stock, when the value of M^{3f} is sufficiently large.

When knowing whether the stocks are larger or smaller than b_{ijp}^{needed} a summation of the variables is needed. This summation will tell whether all the stocks are large enough and, if so, a full load can be sent. The constraints representing this summation are formulated as

$$\sum_{p \in \mathcal{P}} w_{ijtp}^{\text{part}} \leqslant 3 + M^{3g} \cdot w_{ijt}^{\text{sum}}, \qquad (i,j) \in \mathcal{L}^{3g}, \tag{5.3g}$$

where w_{ijt}^{sum} is the binary variable representing if a full load is feasible to send or not, $\mathcal{L}^{3g}$ is the same set used in previous equation and M^{3g} is a large enough number, and

$$\sum_{p \in \mathcal{P}} w_{ijtp}^{\text{part}} \geqslant 3 - M^{3h} \cdot (1 - w_{ijt}^{\text{sum}}), \qquad (i,j) \in \mathcal{L}^{3h}, \tag{5.3h}$$

where $\mathcal{L}^{3h}$ is the same set used in previous equation. The large numbers M^{3g} and M^{3h} do in constraints 5.3g and 5.3h equal to three.

These two constraints will force w_{ijt}^{sum} to 0 if not all w_{ijtp}^{part} are set to 1. The constraints presented in this section will not force any transportation to occur if the stock is large enough, it will only prohibit transportation if equation 5.3h is infeasible. The variable w_{ijt}^{sum} will either activate or deactivate the flow of dirty GSP parts from SKF Factories which will be carried out as full loads of dirty GSP parts. It is desirable to be able to send more than one full load and this creates a need for one more constraints, which is formulated in the same way as some of the previous constraints. The integer variable $y_{ijts^{\text{dirty}}}^{\text{full-load}}$ represents how many full loads sent from node i to j. The additional constraint needed is

$$y_{ijts^{\text{dirty}}}^{\text{full-load}} \leqslant M^{3\text{i}} \cdot w_{ijt}^{\text{sum}}, \qquad (i,j) \in \mathcal{L}^{3\text{i}}, \tag{5.3i}$$

where $\mathcal{L}^{3\text{i}}$ are sets of links between SKF Factories not having a domestic Pallet Pool and their corresponding Pallet Pool and $M^{3\text{i}}$ is a large enough number. In this constraint the large number does not need to be larger than the larges amount of full loads that SKF's distribution network can provide between two nodes. The value of $x_{ijtps^{\text{dirty}}}$ is defined by the number of full loads being sent and this constraint is formulated as

$$x_{ijtps^{\text{dirty}}} = b_{ijp}^{\text{needed}} \cdot y_{ijts^{\text{dirty}}}^{\text{full-load}}, \qquad (i,j) \in \mathcal{L}^{3\text{j}}, \tag{5.3j}$$

where b_{ijp}^{needed} is the amount of GSP parts defining a full load, $y_{ijts^{\text{dirty}}}^{\text{full-load}}$ is the integer variable indicating how many full load of dirty GSP parts should be sent and $\mathcal{L}^{3\text{j}}$ is a set of links of SKF Factories not having a domestic Pallet Pool and their corresponding Pallet Pool.

The transportation of dirty GSP parts from SKF Factories situated in the same country as a Pallet Pool has not yet been addressed. This is because there are no special restrictions on those flows. Since these SKF Factories are situated closer to the Pallet Pools there are no large costs associated with transportation. They are therefore allowed to send dirty GSP parts to Pallet Pools as soon as the stock has reached its maximum level. No restriction on how they are sent is present which means they are sent part by part.

A summary of additional variables, parameters and sets used in this section is shown in Table 5.4.

Table 5.4: Additional components used in constraints at SKF Factories.

Variables	
w_{ijtp}^{part}	Binary variable stating whether transportation of a GPS part p from node i to node j, at time t is feasible or not.
w_{ijt}^{sum}	Binary variable stating whether transportation of dirty GSP parts from node i to node j, at time t is feasible or not.
$y_{ijts^{\text{dirty}}}^{\text{full-load}}$	Number of full loads of dirty GSP parts sent from node i to node j, at time t.
Parameters	
$s_{ip}^{\min}$	Minimum stock level of GSP parts p, at node i.
$s_{ip}^{\max}$	Maximum stock level of GSP parts p, at node i.
$M^{3\text{e}}, \ldots, M^{3\text{i}}$	Large enough numbers.
b_{ip}^{pile}	Number of GSP parts p in a pile originating from node i.
$b_{ijp}^{\text{full-load}}$	Number of piles of GSP parts p in a full load, from node i to node j.
g_{ip}^{mix}	Mix of GSP parts p in a full load sent from node i.
b_{ijp}^{needed}	Number of GSP parts p filling a full load sent from node i to node j.
Sets	
$\mathcal{M}^{3\text{a}}$	Set of all SLS Warehouses.
$\mathcal{N}^{3\text{a}}$	Set of all External Factories and SLS Warehouses.
$\mathcal{O}^{3\text{a}}$	Set of links between Pallet Pools and the SKF Factories they supply.
$\mathcal{L}^{3\text{a}}$	Set of links between SKF Factories and their corresponding Pallet Pool.
$\mathcal{I}^{3\text{b}}, \ldots, \mathcal{I}^{3\text{d}}$	Set of all SKF Factories.
$\mathcal{L}^{3\text{e}}, \ldots, \mathcal{L}^{3\text{j}}$	Set of links between SKF Factories not having a domestic Pallet Pool and their corresponding Pallet Pool.

5.4 External Factory

External Factories are receiving GSP parts in which they pack their components sent to either SLS Warehouses or SKF Factories. The demand of clean GSP parts puts requirements on their stock which are formulated as constraints restricting how the stock shall change and be updated, and between which levels it is allowed to vary.

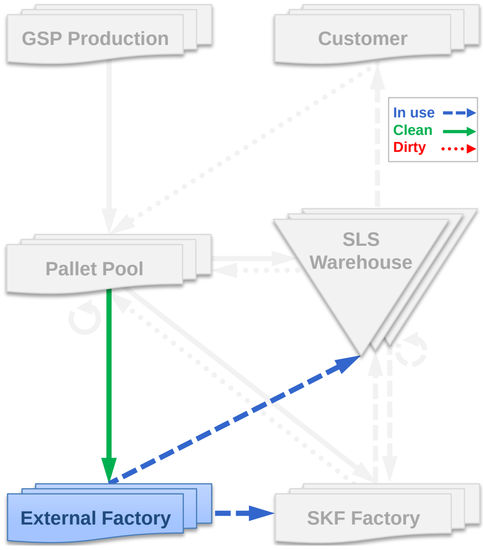


Figure 5.12: Flows of GSP parts at External Factories. External Factories have a demand of clean GSP parts that are equal to their outbound flows of GSP parts in use.

5.4.1 Stock change constraint

The stock change is a function of two types of flows; inbound flow of clean GSP parts and outbound flows of GSP parts in use; these are presented in Figure 5.12. The first type of flow is highlighted in Figure 5.13. Pallet Pools are providing clean GSP parts to External Factories and these flows are represented by

$$\sum_{(m,i)\in\mathcal{M}^{4a}} x_{mitps^{\text{clean}}}, \qquad i \in \mathcal{I}^{4a},$$

where $\mathcal{M}^{4a}$ is a set of links between Pallet Pools and the External Factories that they are supplying and $\mathcal{I}^{4a}$ is a set containing all the External Factories.

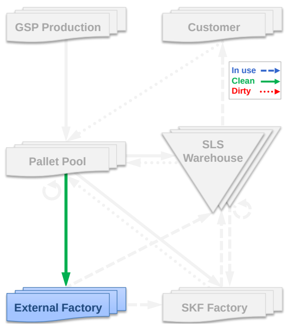


Figure 5.13

The External Factories are providing both SKF Factories and SLS Warehouses with components which result in a flow of GSP parts in use. These flows are highlighted in Figure 5.14. The mathematical representation of the flow is

$$\sum_{n \in \mathcal{N}^{4a}} f_{intps^{\text{in-use}}}, \qquad i \in \mathcal{I}^{4a},$$

where $\mathcal{N}^{4a}$ is a set of all SLS Warehouses and all SKF Factories, and $\mathcal{I}^{4a}$ is a set of all External Factories.

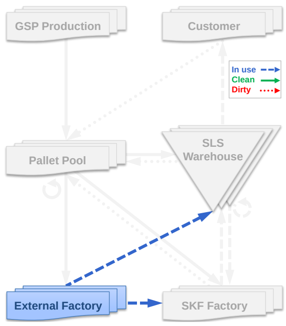


Figure 5.14

The stock change is the difference in inbound flow of clean GSP parts and outbound flows of GSP parts in use, which is formulated as

$$s_{itp}^{\text{change}} = \sum_{(m,i) \in \mathcal{M}^{4a}} x_{mitps^{\text{clean}}} - \sum_{n \in \mathcal{N}^{4a}} f_{intps^{\text{in-use}}}, \qquad i \in \mathcal{I}^{4a}, \tag{5.4a}$$

where $\mathcal{M}^{4a}$ is a set of links between Pallet Pools and the External Factories that they supply, $\mathcal{N}^{4a}$ is a set of all SLS Warehouses and all SKF Factories, and $\mathcal{I}^{4a}$ is a set of all the External Factories.

5.4.2 Stock update constraint

An updated stock for the next time step can be calculated when the stock change is known. As for Pallet Pools it equals the current stock plus the stock change, which is mathematically formulated as

$$s_{i(t+1)p} = s_{itp} + s_{itp}^{\text{change}}, \qquad i \in \mathcal{I}^{4b}, \tag{5.4b}$$

where $\mathcal{I}^{4b}$ is a set of all External Factories.

5.4.3 Stock limits constraints

The stock level is regulated by the parameter s_{ip}^{min} for the lowest stock level and s_{ip}^{max} for the highest stock level. This is formulated as the constraints

$$s_{itp} \geqslant s_{ip}^{\text{min}}, \qquad i \in \mathcal{I}^{4c}, \tag{5.4c}$$

$$s_{itp} \leqslant s_{ip}^{\text{max}}, \qquad i \in \mathcal{I}^{4d}, \tag{5.4d}$$

where $\mathcal{I}^{4c}$ and $\mathcal{I}^{4d}$ are sets of all the External Factories.

A summary of additional variables, parameters and sets used in this section is listed in Table 5.5.

Table 5.5: Additional components used in the constraints concerning the External Factories.

Parameters	
$s_{ip}^{\min}$	Minimum stock level of GSP parts p, at node i.
$s_{ip}^{\max}$	Maximum stock level of GSP parts p, at node i.
Sets	
$\mathcal{M}^{4a}$	Set of links between Pallet Pools and the External Factories they supply.
$\mathcal{N}^{4a}$	Set of all SLS Warehouses and all SKF Factories.
$\mathcal{I}^{4a}, \ldots, \mathcal{I}^{4d}$	Set of all External Factories.

5.5 SLS Warehouse

The Pallet Pools function as hubs for GSP parts whereas SLS Warehouses function as hubs for all products in SKF's distribution network. SLS Warehouses serve as storages for finished goods waiting for customer orders and components that are going to be used in SKF Factories' production. The flows give rise to a need for clean GSP parts in order to assemble GSPs for outbound flows. These conditions are translated into constraints on how the stock of clean GSP parts is updated, the limits on the stock, and how the transportation is handled.

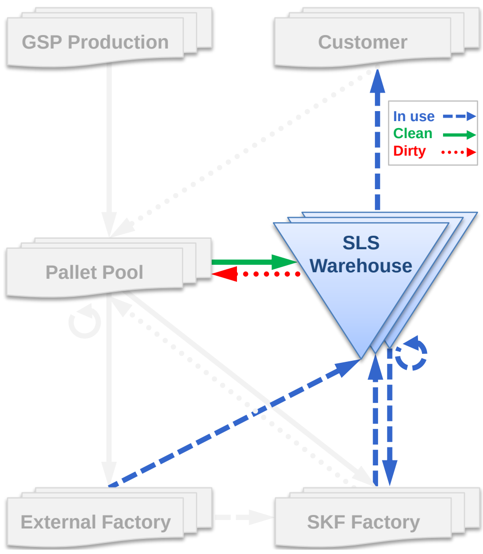


Figure 5.15: Flows of GSP parts at the SLS Warehouses. The commitment of an SLS Warehouse is to serve as storage of finished goods and components for delivery to Customers, SKF Factories, and other SLS Warehouses. To be able to supply these nodes with products packed in GSPs a demand of clean GSP parts might occur.

5.5.1 Stock change constraint

The stock change is a function of four types of flows, just as for SKF Factories; inbound flows of GSP parts in use and clean GSP parts and outbound flows of GSP parts in use and dirty GSP parts, which are all represented in Figure 5.15.

The node categories that manufacture products are SKF Factories and External Factories and the products are stored at SLS Warehouses. This and the fact that SLS Warehouses can send products between each other are creating the flows of GSP parts in use illustrated in Figure 5.16. The inbound flow of GSP parts in use is formulated as

$$\sum_{m \in \mathcal{M}^{5a}} f_{mitps^{\text{in-use}}}, \qquad i \in \mathcal{I}^{5a},$$

were $\mathcal{M}^{5a}$ is a set of all External Factories, SKF Factories and SLS Warehouses and $\mathcal{I}^{5a}$ is a set of all SLS Warehouses.

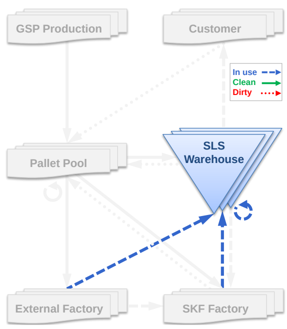


Figure 5.16

SLS Warehouses are sending products to SKF Factories, Customers and other SLS Warehouses which is creating a flow of GSP parts in use; see Figure 5.17. The outbound flow of GSP parts in use are formulated as

$$\sum_{n \in \mathcal{N}^{5a}} f_{intps^{\text{in-use}}}, \qquad i \in \mathcal{I}^{5a},$$

where $\mathcal{N}^{5a}$ is a set of all SKF Factories, Customers, and SLS Warehouses and $\mathcal{I}^{5a}$ is a set of all SLS Warehouses.

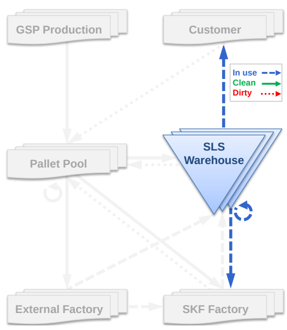


Figure 5.17

SLS Warehouses also possess demand of clean GSP parts since they pick orders, which are packed in GSPs. Which Pallet Pool that is supplying which SLS Warehouses with clean GSP parts is presented in Section 4.4.1 and the flow induced by this demand is illustrated in Figure 5.18. The flow of clean GSP parts is formulated as

$$\sum_{(o,i) \in \mathcal{O}^{5a}} x_{oitps^{\text{clean}}}, \qquad i \in \mathcal{I}^{5a},$$

where $\mathcal{O}^{5a}$ is a set of links between Pallet Pools and the SLS Warehouses they are supplying and $\mathcal{I}^{5a}$ is a set of all SLS Warehouses.

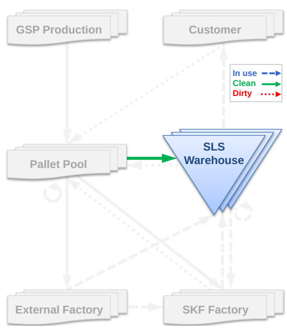


Figure 5.18

Dirty GSP parts can be sent to Pallet Pools for cleaning which is represented by a flow of dirty GSP parts in Figure 5.19. Which Pallet Pool that takes care of the dirty GSP parts is also presented in Section 4.4. The outbound flow is mathematically expressed as

$$x_{ijtps^{\text{dirty}}}, \qquad (i,j) \in \mathcal{L}^{5a},$$

where $\mathcal{L}^{5a}$ is a set of links between SLS Warehouses and their corresponding Pallet Pool.

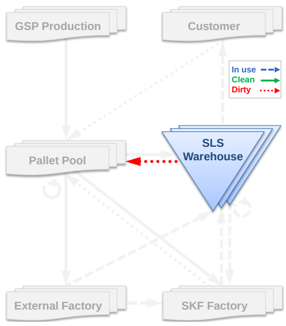


Figure 5.19

The stock change is the difference in the inbound and outbound flows of GSP parts and the stock change constraint is therefore formulated as

$$s_{itp}^{\text{change}} = \sum_{m \in \mathcal{M}^{5a}} f_{mitps^{\text{in-use}}} - \sum_{n \in \mathcal{N}^{5a}} f_{intps^{\text{in-use}}} + \sum_{(o,i) \in \mathcal{O}^{5a}} x_{oitps^{\text{clean}}} - x_{ijtps^{\text{dirty}}}, \quad (i,j) \in \mathcal{L}^{5a}, \quad (5.5a)$$

where $\mathcal{M}^{5a}$ is a set of all External Factories, SKF Factories and SLS Warehouses, $\mathcal{N}^{5a}$ is a set of all SKF Factories, Customers and SLS Warehouses, $\mathcal{O}^{5a}$ is a set of links between Pallet Pools and the SLS Warehouses they supply, and $\mathcal{L}^{5a}$ is a set of links between SLS Warehouses and their corresponding Pallet Pools.

5.5.2 Stock update constraint

The updated stock at an SLS Warehouse is calculated as for all other node categories having a stock, e.i.,

$$s_{i(t+1)p} = s_{itp} + s_{itp}^{\text{change}}, \qquad i \in \mathcal{I}^{5b}, \qquad (5.5b)$$

where $\mathcal{I}^{5b}$ is the set of all SLS Warehouses.

5.5.3 Stock limit constraints

The stock limits are also calculated as for other node categories having a stock,

$$s_{itp} \geqslant s_{ip}^{\min}, \qquad i \in \mathcal{I}^{5c}, \qquad (5.5c)$$

$$s_{itp} \leqslant s_{ip}^{\max}, \qquad i \in \mathcal{I}^{5d}, \qquad (5.5d)$$

where $\mathcal{I}^{5c}$ and $\mathcal{I}^{5d}$ are sets of all SLS Warehouses, $s_{ip}^{\min}$ is the stock minimum and $s_{ip}^{\max}$ is the stock maximum.

5.5.4 Transportation constraint

SLS Warehouses possess the same transportation constraint as SKF Factories. If they have a domestic Pallet Pool they send dirty GSP parts part by part to that Pallet Pool, which does not need any constraints to describe, and SLS Warehouses without a domestic Pallet Pool are only sending dirty GSP parts in full loads if it is feasible. All the constraints needed to reflect the transportation of dirty GSP parts from SLS Warehouses not having a domestic Pallet Pool are

$$s_{itp} + s_{itp}^{\text{change}} - x_{ijtps^{\text{dirty}}} \leqslant b_{ijp}^{\text{needed}} + M^{5\text{e}} \cdot w_{ijtp}^{\text{part}}, \qquad (i,j) \in \mathcal{L}^{5\text{e}}, \tag{5.5e}$$

$$s_{itp} + s_{itp}^{\text{change}} - x_{ijtps^{\text{dirty}}} \geqslant b_{ijp}^{\text{needed}} - M^{5\text{f}} \cdot (1 - w_{ijtp}^{\text{part}}), \qquad (i,j) \in \mathcal{L}^{5\text{f}}, \tag{5.5f}$$

$$\sum_{p \in \mathcal{P}} w_{ijtp}^{\text{part}} \leqslant 3 + M^{5\text{g}} \cdot w_{ijt}^{\text{sum}}, \qquad (i,j) \in \mathcal{L}^{5\text{g}}, \tag{5.5g}$$

$$\sum_{p \in \mathcal{P}} w_{ijtp}^{\text{part}} \geqslant 3 - M^{5\text{h}} \cdot (1 - w_{ijt}^{\text{sum}}), \qquad (i,j) \in \mathcal{L}^{5\text{h}}, \tag{5.5h}$$

$$y_{ijts^{\text{dirty}}}^{\text{full-load}} \leqslant M^{5\text{i}} \cdot w_{ijt}^{\text{sum}}, \qquad (i,j) \in \mathcal{L}^{5\text{i}}, \tag{5.5i}$$

$$x_{ijtps^{\text{dirty}}} = b_{ijp}^{\text{needed}} \cdot y_{ijts^{\text{dirty}}}^{\text{full-load}}, \qquad (i,j) \in \mathcal{L}^{5\text{j}}, \tag{5.5j}$$

where all the sets used in the constraints in this sections are sets of links between SLS Warehouses not having a domestic Pallet Pool and its corresponding Pallet Pool and the largest limits to how large the numbers $M^{5\text{e}}, \ldots, M^{5\text{i}}$ should be are found following the same reasoning as in Section 5.3.4.

A summary of additional variables, parameters and sets used in this section is presented in Table 5.6.

Table 5.6: Additional components used in constraints at SLS Warehouses.

Variables	
w_{ijtp}^{part}	Binary variable stating whether transportation of a GPS part p from node i to node j, at time t is feasible or not.
w_{ijt}^{sum}	Binary variable stating whether transportation of GSP parts from node i to node j, at time t is feasible or not.
$y_{ijts^{\text{dirty}}}^{\text{full-load}}$	Number of full load sent from node i to node j, at time t.
Parameters	
$M^{5\text{e}}, \ldots, M^{5\text{i}}$	Large enough numbers.
b_{ip}^{pile}	Number of GSP parts p in a pile.
$b_{ijp}^{\text{full-load}}$	Number of piles of GSP parts p in a full load, from node i to node j.
g_{ip}^{mix}	Mix of GSP parts p in a full load sent from node i.
b_{ijp}^{needed}	Number of GSP parts p filling a full load sent from node i to node j.
Sets	
$\mathcal{M}^{5\text{a}}$	Set of all External Factories, SKF Factories and SLS Warehouses
$\mathcal{N}^{5\text{a}}$	Set of all SKF Factories, Customers and SLS Warehouses.
$\mathcal{O}^{5\text{a}}$	Set of links between Pallet Pools and the SLS Warehouses they supply.
$\mathcal{L}^{5\text{a}}$	Set of links between SLS Warehouses and their corresponding Pallet Pool.
$\mathcal{I}^{5\text{b}}, \ldots, \mathcal{I}^{5\text{d}}$	Set of all SLS Warehouses.
$\mathcal{L}^{5\text{e}}, \ldots, \mathcal{L}^{5\text{j}}$	Set of links between SLS Warehouses not having a domestic Pallet Pool and their corresponding Pallet Pool.

5.6 Customer

In a perfectly functioning returnable packaging systems, the loop of packaging is closed which means that all packaging shipped are returned. This is however an utopian scenario and rarely true in reality. In SKF's distribution network the Customers are returning the GSP parts they have received to Pallet Pools at a certain return rate which is formulated as a return rate constraint.

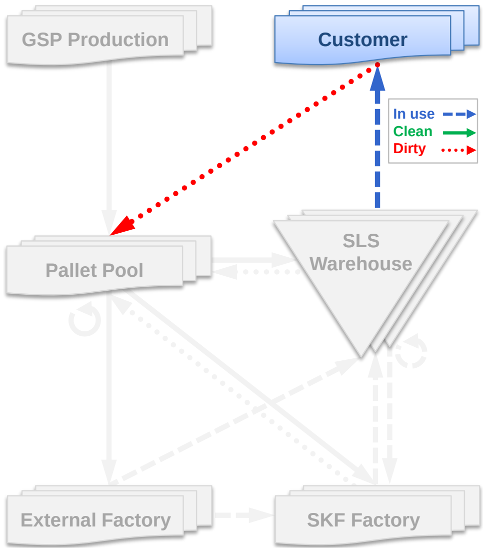


Figure 5.20: Flow of GSP parts at Customers. Customers are receiving ordered products in GSPs which they can chose to return to predetermined Pallet Pools.

5.6.1 Return rate constraint

The rate at which Customers return GSP parts is represented by a parameter r_{ijtp} which depend on the Customer i, the receiving Pallet Pool j, the time t and GSP part p. The return rate is defined as the ratio of the inbound flow of GSP parts in use and the outbound flow of dirty GSP parts. These two flows can be seen in Figure 5.20.

Which Pallet Pools that are receiving dirty GSP parts from which Customer are predetermined and the set containing these links are named $\mathcal{L}^{6a}$. The mathematical formulation of the relation between inbound and outbound flow at Customers are

$$x_{ijtps^{\text{dirty}}} = r_{ijtp} \cdot \sum_{m \in \mathcal{M}^{6a}} f_{mitps^{\text{in-use}}}, \qquad (i,j) \in \mathcal{L}^{6a}, \tag{5.6a}$$

where r_{ijtp} is the return rate, $\mathcal{M}^{6a}$ is a set of all SLS Warehouses and $\mathcal{L}^{6a}$ is a set of links between Customers and their corresponding Pallet Pool.

A summary of additional variables, parameters and sets used in this section is listed in Table 5.7.

Table 5.7: Additional components used in constraint at Customer.

Parameters	
r_{ijtp}	Return rate of GSP parts p sent from node i to node j, at time t.
Sets	
$\mathcal{L}^{6a}$	Set of links between Customers and their corresponding Pallet Pools.
$\mathcal{M}^{6a}$	Set of all SLS Warehouses.

5.7 Objective Function

The total cost of the optimization model is given by the variables and their corresponding cost parameters. As the purpose of this project states, only transportation and purchasing costs are considered. Variables not having transportation or purchasing costs are therefore omitted in the objective function. The Returnable Deposit (RD) and Pallet Pool Handling Fee (PPHF) costs arising when using the returnable packaging system at SKF is not included in the purpose since these costs are equal at all nodes in the network and, therefore does not affect the choice of flows.

In the optimization model ten variables are used, namely

$$x_{ijtps^{\text{clean}}},\quad x_{ijtps^{\text{dirty}}},\quad y^{\text{full-load}}_{ijts^{\text{clean}}},\quad y^{\text{full-load}}_{ijts^{\text{dirty}}},\quad y^{\text{pile}}_{ijtp},\quad s_{itp},\quad s^{\text{change}}_{itp},\quad w^{\text{part}}_{ijtp},\quad w^{\text{sum}}_{ijt}, and\quad z_{ijtp}.$$

Amongst them, the variables having transportation costs or purchasing costs related to them are; $y^{\text{full-load}}_{ijts^{\text{clean}}}$, y^{pile}_{ijtp}, and z_{ijtp}.

The binary variables w^{part}_{ijtp} and w^{sum}_{ijt} do not have any costs related to them since their only purpose is to establish whether or not the stock level at SKF Factories or SLS Warehouses not situation in the same country as a Pallet Pools is large enough to be able to send a full load of dirty GSP parts. If this is the case a transportation constraint will be activated which specifies how many full loads and how many dirty GSP parts that should be sent to a predetermined Pallet Pool. Sending dirty GSP parts $y^{\text{full-load}}_{ijts^{\text{dirty}}}$ is not a cost paid by SKF Logistics Services and is therefore omitted in the objective function.

The flow of dirty GSP parts represented by the variable $x_{ijtps^{\text{dirty}}}$ have no associated costs in the model. Transportation costs are of course present in the real world but they are not of interest in the model. This is because the flow of dirty GSP parts is a consequence of the flow of clean GSP parts and it can be seen as a flow which the model cannot affect without changing the flow of clean GSP parts. The destinations of the dirty GSP parts are predefined and hence no optimization of where these flows occur can be made with regard to cost. The omitted flow of dirty GSP parts is shown as shaded arrows in Figure 5.21, these arrows represent no costs being involved in these flows.

The stock change s_{itp}^{change} is a variable whose only purpose is to provide information on how the stock is changing and no costs are therefore related to this variable either. A stock of clean GSP parts s_{itp} exists at all node categories except for GSP Production and Customer. The stock of clean GSP parts at SLS Warehouses, External Factories and SKF Factories should be so small that the costs related to them are assumed to be negligible. The stock of clean GSP parts at Pallet Pools are, however, larger. There are always costs generated by keeping a stock, for example costs of tied-up capital and risks of damaged and obsolete items. The optimization model could have taken these costs into account as well when optimizing the flow of GSP parts, but SLS was not interested in optimizing with respect to these costs. The true costs involved in the storage can be difficult to possess because no uniform definition of stock costs is present at SKF. An alternative way could be to assume that the storage costs are equal at all Pallet Pools, but then the storage costs do not matter in the optimization model anyway since no Pallet Pool will be preferred with regard to cost minimization.

Earlier in this chapter constraints for transportation of clean GSP parts was introduced; see Section 5.1.4 and 5.2.1. These constraints stated that clean GSP parts sent from Pallet Pools are transported in piles, y_{ijtp}^{pile}, or in full loads, $y_{ijts^{\text{clean}}}^{\text{full-load}}$, and new GSP parts are transported in full loads, z_{ijtp}. The costs of interest is therefore those of transporting clean GSP parts in these entities, not as single GSP parts. The variable $x_{ijtps^{\text{clean}}}$ is therefor not incorporated in the objective function.

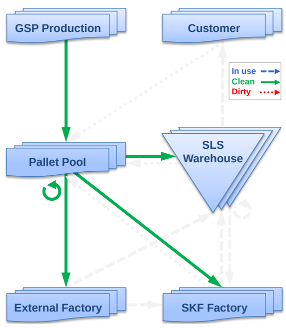


Figure 5.21

The variables that do affect the objective function are y_{ijtp}^{pile}, z_{ijtp}, and $y_{ijts^{\text{clean}}}^{\text{full-load}}$ containing clean GSP parts, which represent the flows illustrated in Figure 5.21. The costs related to these variables have a purchasing cost embedded if the flow is between a GSP Production site and a Pallet Pool. Otherwise, the cost parameters only include transportation costs and custom fees where such are present. The cost of transporting a full load of clean GSP parts is denoted $c_{ijp}^{\text{full-load}}$ and the cost of transporting a pile of clean GSP parts is denoted c_{ijp}^{pile}. These costs depend on the departure node i, the destination node j, and the type of GSP part p. The cost is depending on the type of GSP part because the parts are bundled into piles which have different shapes and sizes depending on the type of GSP parts they are holding. The transportation costs depend on how much space the goods occupy in the trailer or the container.

The total cost of transporting full loads of clean GSP parts in SKF's distribution network is formulated as

$$\sum_{t \in \mathcal{T}} \sum_{p \in \mathcal{P}} \left(\sum_{(i,j) \in \mathcal{M}^{7\text{a}}} c_{ijp}^{\text{full-load}} \cdot y_{ijts^{\text{clean}}}^{\text{full-load}} + \sum_{(k,l) \in \mathcal{N}^{7\text{a}}} c_{klp}^{\text{full-load}} \cdot z_{kltp} \right),$$

where $c_{ijp}^{\text{full-load}}$ is the cost of transporting one full load of clean GSP parts, $y_{ijts^{\text{clean}}}^{\text{full-load}}$ is an integer variable representing the number of full loads sent on links within set $\mathcal{M}^{7a}$, which is a set of links between Pallet Pools. The set $\mathcal{N}^{7a}$ is a set of links between GSP Production sites and Pallet Pools and z_{kltp} is an integer variable representing the number of full loads sent on those links.

The total cost of transporting piles of clean GSP parts is formulated as

$$\sum_{t\in\mathcal{T}}\sum_{p\in\mathcal{P}}\sum_{(m,n)\in\mathcal{O}^{7a}} c^{\text{pile}}_{mnp}\cdot y^{\text{pile}}_{mntp},$$

where c^{pile}_{mntp} is the cost of transporting one pile of clean GSP parts, y^{pile}_{mntp} is an integer variable representing the number of piles sent, $\mathcal{O}^{7a}$ is a set of links between Pallet Pools and nodes receiving clean GSP parts pile by pile.

The objective function is to minimize the sum of the costs of the three types of flows in interest, namely to

$$\begin{aligned}\text{minimize}\quad \sum_{t\in\mathcal{T}}\sum_{p\in\mathcal{P}}\Bigg(&\sum_{(i,j)\in\mathcal{M}^{7a}} c^{\text{full-load}}_{ijp}\cdot y^{\text{full-load}}_{ijts^{\text{clean}}} + \sum_{(k,l)\in\mathcal{N}^{7a}} c^{\text{full-load}}_{klp}\cdot z_{kltp}\\ &+\sum_{(m,n)\in\mathcal{O}^{7a}} c^{\text{pile}}_{mnp}\cdot y^{\text{pile}}_{mntp}\Bigg)\end{aligned} \tag{5.7a}$$

where $\mathcal{M}^{7a}$ is a set of links between Pallet Pools and nodes receiving clean GSP parts in full loads, $\mathcal{N}^{7a}$ is a set of links between GSP Production sites and Pallet Pools and $\mathcal{O}^{7a}$ is a set of links between Pallet Pools and nodes receiving clean GSP parts pile by pile.

The additional variables, parameters and sets used in this section are listed in Table 5.8.

Table 5.8: Additional components used in the objective function.

Variables	
$y^{\text{full-load}}_{ijts^{\text{clean}}}$	Number of full loads of clean GSP parts sent from i to node j, at time t.
y^{pile}_{ijtp}	Number of piles of GSP parts p sent from node i to node j, at time t.
z_{ijtp}	Number of full loads of new GSP parts sent from node i to node j, at time t.
Parameters	
$c^{\text{full-load}}_{ijp}$	Cost of transporting a full load of GSP parts p sent from node i to node j.
c^{pile}_{ijp}	Cost of transporting a pile of GSP parts p sent from node i to node j.
Sets	
$\mathcal{M}^{7\text{a}}$	Set of links between Pallet Pools and nodes receiving clean GSP parts in full loads.
$\mathcal{N}^{7\text{a}}$	Set of links between GSP Production and Pallet Pools.
$\mathcal{O}^{7\text{a}}$	Set of links between Pallet Pools and nodes receiving clean GSP parts pile by pile.

5.8 The complete mathematical model

When the objective function and all the constraints defined in the previous subsections are merged, the mathematical model is fully formulated. The result received from solving the mathematical model will describe how the flow of GSP parts should be distributed in order to minimize the total costs. In addition to the constraints described in the previous chapter, all but one variable used in the optimization model are restricted to be greater than zero. The only variable allowed to be negative is the stock change s^{change}_{itp}. The reasons for the other variables required to be non-negative are physical restrictions. The stock s_{itp} can not be negative and there is no definition for flows carrying negative amounts of clean, $x_{ijtps^{\text{clean}}}$, or dirty, $x_{ijtps^{\text{dirty}}}$, GSP parts. The same argument holds for y^{pile}_{ijtp}, $y^{\text{full-load}}_{ijts^{\text{clean}}}$, $y^{\text{full-load}}_{ijts^{\text{dirty}}}$, and z_{ijtp}. The variables w^{part}_{ijtp} and w^{sum}_{ijt} are defined as binary since they reflect logical restrictions.

To minimize the cost related to transportation and purchasing of GSP parts world wide the optimization objective is to

$$\text{minimize} \quad \sum_{t\in\mathcal{T}}\sum_{p\in\mathcal{P}}\left(\sum_{(i,j)\in\mathcal{M}^{7\text{a}}} c^{\text{full-load}}_{ijp}\cdot y^{\text{full-load}}_{ijts^{\text{clean}}} + \sum_{(k,l)\in\mathcal{N}^{7\text{a}}} c^{\text{full-load}}_{klp}\cdot z_{kltp} + \sum_{(m,n)\in\mathcal{O}^{7\text{a}}} c^{\text{pile}}_{mnp}\cdot y^{\text{pile}}_{mntp}\right) \tag{5.7a}$$

subject to

$$s^{\text{change}}_{itp} = \sum_{(m,i)\in\mathcal{M}^{1a}} x_{mitps^{\text{clean}}} + \sum_{(n,i)\in\mathcal{N}^{1a}} q_{nitp} \cdot x_{nitps^{\text{dirty}}} - \sum_{(i,o)\in\mathcal{O}^{1a}} x_{iotps^{\text{clean}}}, \quad i \in \mathcal{I}^{1a}, \tag{5.1a}$$

$$s^{\text{change}}_{itp} = -\sum_{m\in\mathcal{M}^{3a}} f_{imtps^{\text{in-use}}} + \sum_{n\in\mathcal{N}^{3a}} f_{nitps^{\text{in-use}}} + \sum_{(o,i)\in\mathcal{O}^{3a}} x_{oitps^{\text{clean}}} - x_{ijtps^{\text{dirty}}}, (i,j) \in \mathcal{L}^{3a}, \tag{5.3a}$$

$$s^{\text{change}}_{itp} = \sum_{(m,i)\in\mathcal{M}^{4a}} x_{mitps^{\text{clean}}} - \sum_{n\in\mathcal{N}^{4a}} f_{intps^{\text{in-use}}}, \quad i \in \mathcal{I}^{4a}, \tag{5.4a}$$

$$s^{\text{change}}_{itp} = \sum_{m\in\mathcal{M}^{5a}} f_{mitps^{\text{in-use}}} - \sum_{n\in\mathcal{N}^{5a}} f_{intps^{\text{in-use}}} + \sum_{(o,i)\in\mathcal{O}^{5a}} x_{oitps^{\text{clean}}} - x_{ijtps^{\text{dirty}}}, \quad (i,j) \in \mathcal{L}^{5a}, \tag{5.5a}$$

$$s_{i(t+1)p} = s_{itp} + s^{\text{change}}_{itp}, \quad i \in \mathcal{I}^{1b} \cup \mathcal{I}^{3b} \cup \mathcal{I}^{4b} \cup \mathcal{I}^{5b}, \tag{5.1b, 5.3b, 5.4b, 5.5b}$$

$$s_{itp} \geqslant s^{\min}_{ip}, \quad i \in \mathcal{I}^{1c} \cup \mathcal{I}^{3c} \cup \mathcal{I}^{4c} \cup \mathcal{I}^{5c}, \tag{5.1c, 5.3c, 5.4c, 5.5c}$$

$$s_{itp} \leqslant s^{\max}_{ip}, \quad i \in \mathcal{I}^{1d} \cup \mathcal{I}^{3d} \cup \mathcal{I}^{4d} \cup \mathcal{I}^{5d}, \tag{5.1d, 5.3d, 5.4d, 5.5d}$$

$$s_{itp} + s^{\text{change}}_{itp} - x_{ijtps^{\text{dirty}}} \leqslant b^{\text{needed}}_{ijp} + M^1 \cdot w^{\text{part}}_{ijtp}, \quad (i,j) \in \mathcal{L}^{3e} \cup \mathcal{L}^{5e}, \tag{5.3e, 5.5e}$$

$$s_{itp} + s^{\text{change}}_{itp} - x_{ijtps^{\text{dirty}}} \geqslant b^{\text{needed}}_{ijp} - M^1 \cdot (1 - w^{\text{part}}_{ijtp}), \quad (i,j) \in \mathcal{L}^{3f} \cup \mathcal{L}^{5f}, \tag{5.3f, 5.5f}$$

$$\sum_{p\in\mathcal{P}} w^{\text{part}}_{ijtp} \leqslant 3 + M^2 \cdot w^{\text{sum}}_{ijt}, \quad (i,j) \in \mathcal{L}^{3g} \cup \mathcal{L}^{5g} \tag{5.3g, 5.5g}$$

$$\sum_{p\in\mathcal{P}} w^{\text{part}}_{ijtp} \geqslant 3 - M^2 \cdot (1 - w^{\text{sum}}_{ijt}), \quad (i,j) \in \mathcal{L}^{3h} \cup \mathcal{L}^{5h}, \tag{5.3h, 5.5h}$$

$$y^{\text{full-load}}_{ijts^{\text{dirty}}} \leqslant M^3 \cdot w^{\text{sum}}_{ijt}, \quad (i,j) \in \mathcal{L}^{3i} \cup \mathcal{L}^{5i}, \tag{5.3i, 5.5i}$$

$$x_{ijtps^{\text{dirty}}} = b^{\text{needed}}_{ijp} \cdot y^{\text{full-load}}_{ijts^{\text{dirty}}}, \quad (i,j) \in \mathcal{L}^{3j} \cup \mathcal{L}^{5j}, \tag{5.3j, 5.5j}$$

$$x_{ijtps^{\text{dirty}}} = r_{ijtp} \cdot \sum_{m\in\mathcal{M}^{6a}} f_{mitps^{\text{in-use}}}, \quad (i,j) \in \mathcal{L}^{6a}, \tag{5.6a}$$

$$x_{ijtps^{\text{clean}}} = g^{\text{mix}}_{ip} \cdot b^{\text{pile}}_{ip} \cdot b^{\text{full-load}}_{ijp} \cdot y^{\text{full-load}}_{ijts^{\text{clean}}}, \quad (i,j) \in \mathcal{L}^{1e}, \tag{5.1e}$$

$$x_{ijtps^{\text{clean}}} = b^{\text{pile}}_{ip} \cdot b^{\text{full-load}}_{ijp} \cdot z_{ijtp}, \quad (i,j) \in \mathcal{L}^{2a}, \tag{5.2a}$$

$$x_{ijtps^{\text{clean}}} = b^{\text{pile}}_{ip} \cdot y^{\text{pile}}_{ijtp}, \quad (i,j) \in \mathcal{L}^{1f}, \tag{5.1f}$$

$$\sum_{(i,m)\in\mathcal{M}^{2b}} \sum_{t\in\mathcal{T}} x_{imtps^{\text{clean}}} \geqslant d^{\min}_{ip}, \quad i \in \mathcal{I}^{2b}, \tag{5.2b}$$

$$x_{ijtps^{\text{clean}}}, x_{kltps^{\text{dirty}}} \geqslant 0, \quad (i,j) \in \mathcal{L}^{8a}, (k,l) \in \mathcal{L}^{8b}$$

$$s_{itp} \geqslant 0, \quad i \in \mathcal{I}^{8c},$$

$$y^{\text{pile}}_{ijtp}, y^{\text{full-load}}_{klts^{\text{clean}}}, z_{kltp} \geqslant 0, \quad \text{integer} \quad (i,j) \in \mathcal{L}^{8d}, (k,l) \in \mathcal{L}^{8e}$$

$$y^{\text{full-load}}_{klts^{\text{dirty}}} \geqslant 0, \quad \text{integer} \quad (i,j) \in \mathcal{L}^{8f},$$

$$w^{\text{part}}_{ijtp}, w^{\text{sum}}_{ijt}, \quad \text{binary} \quad (i,j) \in \mathcal{L}^{8g},$$

where the components used in the model are described in Table 5.9.

Table 5.9: Variables, parameters, and sets used in the complete mathematical model.

Continuous variables	
$x_{ijtps^{\text{clean}}}$	Amount of clean GSP parts p sent from node i to node j, at time t.
$x_{ijtps^{\text{dirty}}}$	A mount of dirty GSP parts p sent from node i to node j, at time t.
s_{itp}	Stock of GSP parts p at node i, at time t.
s_{itp}^{change}	The change in stock of GSP parts p at node i, at time t.
Integer variables	
$y_{ijts^{\text{clean}}}^{\text{full-load}}$	Number of full loads of clean GSP parts sent from i to node j, at time t.
$y_{ijts^{\text{dirty}}}^{\text{full-load}}$	Number of full loads of dirty GSP parts sent from i to node j, at time t.
y_{ijtp}^{pile}	Number of piles of GSP parts p sent from node i to node j, at time t.
z_{ijtp}	Number of full loads of new GSP parts sent from i to node j, at time t.
Binary variables	
w_{ijtp}^{part}	Stating if transportation of GSP parts p from node i to node j, at time t is feasible.
w_{ijt}^{sum}	Stating if transportation of dirty GSP parts from node i to node j, at time t is feasible.
Parameters	
$c_{ijp}^{\text{full-load}}$	Cost of transporting a full load of GSP parts p sent from node i to node j.
c_{ijp}^{pile}	Cost of transporting a pile of GSP parts p sent from node i to node j.
$f_{ijtps^{\text{in-use}}}$	Amount of GSP parts p with status "in use" sent from node i to node j, at time t.
q_{ijtp}	The scrap rate of dirty GSP parts p at node j, depending on sending node i, at time t.
s_{ip}^{min}	Minimum stock level of GSP parts p, at node i.
s_{ip}^{max}	Maximum stock level of GSP parts p, at node i.
M^1, M^2, M^3	Large enough numbers, defined in Sections 5.3.4 and 5.5.4.
b_{ijp}^{needed}	Number of GSP parts p filling a full load sent from node i to node j.
r_{ijtp}	Return rate of GSP parts p sent from node i to node j, at time t.
g_{ip}^{mix}	Mix of GSP parts p in a full load sent from node i.
b_{ip}^{pile}	Number of GSP parts p in a pile originating from node$i$.
$b_{ijp}^{\text{full-load}}$	Number of piles of GSP parts p in a full load, from node i to node j.
d_{ip}^{min}	Minimum amount of new GSP parts p delivered each year from node i.

Most common sets	
$\mathcal{T}$	The time interval used in the model.
$\mathcal{P}$	The different GSP parts; {bases, collars, lids}.
$\mathcal{S}$	The different statuses of a GSP parts; $\{s^{\text{clean}}, s^{\text{dirty}}, s^{\text{in-use}}\}$
Sets used in the objective function (5.7a)	
$\mathcal{M}^{7a}$	Set of links between Pallet Pools and nodes receiving clean GSP parts in full load.
$\mathcal{N}^{7a}$	Set of links between GSP Production and Pallet Pools.
$\mathcal{O}^{7a}$	Set of links between Pallet Pools and nodes receiving GSP parts pile by pile.
Sets used in constraint (5.1a)	
$\mathcal{M}^{1a}$	Set of links between nodes sending clean GSP parts and receiving Pallet Pools.
$\mathcal{N}^{1a}$	Set of links between nodes sending dirty GSP parts and receiving Pallet Pools.
$\mathcal{O}^{1a}$	Set of links between Pallet Pools and nodes with a demand of clean GSP parts.
$\mathcal{I}^{1a}$	Set of all Pallet Pools.
Sets used in constraint (5.3a)	
$\mathcal{M}^{3a}$	Set of all SLS Warehouses.
$\mathcal{N}^{3a}$	Set of all External Factories and SLS Warehouses.
$\mathcal{O}^{3a}$	Set of links between Pallet Pools and the SKF Factories they supply.
$\mathcal{L}^{3a}$	Set of links between SKF Factories and their corresponding Pallet Pool.
Sets used in constraint (5.4a)	
$\mathcal{M}^{4a}$	Set of links between Pallet Pools and the External Factories they supply.
$\mathcal{N}^{4a}$	Set of all SLS Warehouses and all SKF Factories.
$\mathcal{I}^{4a}$	Set of all External Factories.
Sets used in constraint (5.5a)	
$\mathcal{M}^{5a}$	Set of all External Factories, SKF Factories and SLS Warehouses
$\mathcal{N}^{5a}$	Set of all SKF Factories, Customers and SLS Warehouses.
$\mathcal{O}^{5a}$	Set of links between Pallet Pools and the SLS Warehouses they supply.
$\mathcal{L}^{5a}$	Set of links between SLS Warehouses and their corresponding Pallet Pool.
Sets used in constraint (5.1b), (5.3b), (5.4b), (5.5b)	
$\mathcal{I}^{1b} \cup \mathcal{I}^{3b} \cup \mathcal{I}^{4b} \cup \mathcal{I}^{5b}$	Union of sets containing Pallet Pools, SKF Factories, External Factories and SLS Warehouses.
Sets used in constraint (5.1c), (5.3c), (5.4c), (5.5c)	
$\mathcal{I}^{1c} \cup \mathcal{I}^{3c} \cup \mathcal{I}^{4c} \cup \mathcal{I}^{5c}$	Union of sets containing Pallet Pools, SKF Factories, External Factories and SLS Warehouses.

Sets used in constraint (5.1d), (5.3d), (5.4d), (5.5d)	
$\mathcal{I}^{1d} \cup \mathcal{I}^{3d} \cup \mathcal{I}^{4d} \cup \mathcal{I}^{5d}$	Union of sets containing Pallet Pools, SKF Factories, External Factories and SLS Warehouses.

Sets used in constraint (5.3e), (5.5e)	
$\mathcal{L}^{3e} \cup \mathcal{L}^{5e}$	Union of sets containing links between SKF Factories and SLS Warehouses not having a domestic Pallet Pool and their corresponding Pallet Pool.

Sets used in constraint (5.3f), (5.5f)	
$\mathcal{L}^{3f} \cup \mathcal{L}^{5f}$	Union of sets containing links between SKF Factories and SLS Warehouses not having a domestic Pallet Pool and their corresponding Pallet Pool.

Sets used in constraint (5.3g), (5.5g)	
$\mathcal{L}^{3g} \cup \mathcal{L}^{5g}$	Union of sets containing links between SKF Factories and SLS Warehouses not having a domestic Pallet Pool and their corresponding Pallet Pool.

Sets used in constraint (5.3h), (5.5h)	
$\mathcal{L}^{3h} \cup \mathcal{L}^{5h}$	Union of sets containing links between SKF Factories and SLS Warehouses not having a domestic Pallet Pool and their corresponding Pallet Pool.

Sets used in constraint (5.3i), (5.5i)	
$\mathcal{L}^{3i} \cup \mathcal{L}^{5i}$	Union of sets containing links between SKF Factories and SLS Warehouses not having a domestic Pallet Pool and their corresponding Pallet Pool.

Sets used in constraint (5.3j), (5.5j)	
$\mathcal{L}^{3j} \cup \mathcal{L}^{5j}$	Union of sets containing links between SKF Factories and SLS Warehouses not having a domestic Pallet Pool and their corresponding Pallet Pool.

Sets used in constraint (5.6a)	
$\mathcal{L}^{6a}$	Set of links between Customers and their corresponding Pallet Pools.
$\mathcal{M}^{6a}$	Set of all SLS Warehouses.

Sets used in constraint (5.1e)	
$\mathcal{L}^{1e}$	Set of links between Pallet Pools and destination nodes receiving GSP parts in full loads.

Sets used in constraint (5.2a)	
$\mathcal{L}^{2a}$	Set of links between GSP Production sites and receiving Pallet Pools.

Sets used in constraint (5.1f)	
$\mathcal{L}^{1f}$	Set of links between Pallet Pools and destination nodes receiving GSP parts pile by pile.

Sets used in constraint (5.2b)	
$\mathcal{M}^{2b}$	Set of links between GSP Production sites and receiving Pallet Pools.
$\mathcal{I}^{2b}$	Set of all GSP Production sites.
Sets used when restricting variables	
$\mathcal{L}^{8a}$	Set of links between nodes sending and receiving clean GSP parts.
$\mathcal{L}^{8b}$	Set of links between nodes sending and receiving dirty GSP parts.
$\mathcal{I}^{8c}$	Set of all nodes holding a stock eg. Pallet Pools, External Factories, SKF Factories and SLS Warehouses.
$\mathcal{L}^{8d}$	Set of links between nodes sending and receiving clean GSP parts pile by pile.
$\mathcal{L}^{8e}$	Set of links between nodes sending and receiving clean GSP parts as full load.
$\mathcal{L}^{8f} = \mathcal{L}^{8g}$	Set of links between SKF Factories and SLS Warehouses not having a domestic Pallet Pool and their corresponding Pallet Pool.

5.9 Classification of the optimization problem

In Chapter 3 some axioms where stated which need to be fulfilled in order to reap the benefits that is associated with linear programming [3]. The first and second axiom are proportionality and additivity, which means that no economies of scale or fixed cost can be present and only first order terms can exist. The costs in the objective function are defined per unit with no fixed cost and throughout the constraints and objective function no higher order terms were used. The first and second axiom are fulfilled and the problem is hence linear. The third axiom is divisibility, which means that the problem needs to be continuous. When describing the constraints both continuous, integer, and binary variables were used. The problem is therefore not a linear program, but a MILP. The last axiom is determinism which means no randomness present, which also is fulfilled in this project. In SKF's distribution network there are of course non-deterministic elements present which the model does not take into account. Some of the parameters which are not deterministic in the real problem are minimum and maximum stock limits, return rates and scrap rates.

Since the optimization model formulated is a MILP it can reap the benefits associated with LP to some extent. The Branch and Bound method explained in Chapter 3 introduced how a MILP can be solved and the complexity of a MILP was also presented in the same chapter.

6

Tests and Results

TESTING of the model is important to be sure that the obtained result is reliable. Given the result obtained from testing, further modification of the model can be necessary to make before solving the problem, as described in Chapter 2.1. The tests used for investigating the correctness of the model is presented in this chapter along with modifications of the model and results obtained by solving the modified model using real data.

6.1 Testing of the model

The mathematical model was programmed in AMPL and to test whether the model is correctly formulated, artificial data was used and the result analyzed. The artificial data covered all sets and all parameters included in the model. To create a distribution network of manageable size nine nodes were chosen to define the network, since the size of such a small network enables a manual study of the flows and errors occurring are easily traceable.

The nine nodes included; two Pallet Pools, two SLS Warehouses, two SKF Factories, one External Factory, one Customer, and one GSP Production site. The properties of the nodes and the links between them were chosen in a way that all constraints present in the model would be tested. A table with the nodes and links' properties are presented in Table 6.1

The properties listed in Table 6.1 allows testing of transportation constraints for full loads and pile by pile of clean GSP parts when destination nodes are situated both close to a Pallet Pool and far away. They also allow testing of transportation of dirty GSP parts transported either as full loads or part by part. Transportation constraint for GSP Production sites, External Factories, and Customers are also tested. Besides testing different transportation constraints, stock change, stock update, and stock limits constraints were also tested for the different node properties. The parameter valuses were also stated and different values were chosen in order to study different scenarios of the model.

Table 6.1: Links between nodes in the small network and their transportation properties.

	Pallet Pool 1		Pallet Pool 2	
	Clean	Dirty	Clean	Dirty
SLS WH 1 , close	piles	parts	-	-
SLS WH 2, far away	piles	-	piles	parts
SKF Factory 1, close	piles	parts	-	-
SKF Factory 2, far away	full load	full load	full load	-
External Factory	-	-	piles	parts
Customer	-	parts	-	parts
GSP Production	full load	-	-	-

First, the value of each type of parameter was chosen to be the same, independent of nodes and type of GSP part. This made testing of stock change and stock update constraints easy. The result of this test showed that all constraints concerning the stock change and stock update are functioning properly for all node categories. Most transportation constraints were also tested with this parameter setting and a conclusion can be made that the transportation constraints are stated correctly. When testing the stock limit constraints, the stock maximum parameter was chosen in a way that would made some nodes sending dirty GSP parts for cleaning, and some nodes would not. By varying this parameter for all node categories the result showed that the stock limit constraints work properly.

The model only gives a result if the parameters are chosen wisely. A problem occurs when the parameter GSP parts in use, $f_{ijtps^{\text{in-use}}}$, is given different values for different parts. The model is then becoming infeasible. Why this is happening is because of the mix parameter g_{ip}^{mix}. When a node is receiving and sending GSP parts in use, the differences in these flows will add up to the stock, or create a demand of clean GSP parts if the outbound flow is larger than the inbound flow. If the node is situated such that it only sends and/or receives GSP parts in full loads, the inbound or outbound flows of clean and dirty GSP parts can only take place in quantities predetermined by the mix parameter. The mix parameter is, however, not reflecting reality since it is impossible to retrieve exact data for every full load's mix of GSP parts. The lack of accuracy of the mix parameters disables the nodes possibility to receive or send clean and dirty GSP parts in the quantities they really need. The stock of some of the GSP parts is inevitable going to build up and at some point become larger than its stock maximum, at the same time as the stock of other GSP parts are going to be kept at its stock minimum. Neither inbound flow nor outbound flow of clean and dirty GSP parts can take place in this scenario and the model is then becoming infeasible. This infeasible property is present everywhere in the model where the parameter g_{ip}^{mix} is present.

From this finding it is possible to conclude that the requirements of transportation in full loads consisting of a mixture of GSP parts is infeasible and it must therefore be removed from the model. A modified version of the model presented in Chapter 5.8 is created by deleting the constraint (5.3e) – (5.3j) and (5.5e) – (5.5j), and modify the constraint (5.1e) and (5.1f). By deleting these constraints, the transportation of dirty GSP parts is not restricted to be performed in full loads. Hence all transportation of dirty GSP parts are performed part by part in the modified model. The constraints (5.1e) and (5.1f) are modified by deleting the mix-parameters in the constraint (5.1e), which will result in two changes. The first change is that transportation performed in full loads will only occur in loads containing one type of GSP parts. The second change is that transportation in full loads will only occur between Pallet Pools and between GSP Production sites and Pallet Pools, which are concerning both the constraints (5.1e) and (5.1f). The modification of the constraint of (5.1e) is formulated as

$$x_{ijtps^{\text{clean}}} = b_{ip}^{\text{pile}} \cdot b_{ijp}^{\text{full-load}} \cdot y_{ijtps^{\text{clean}}}^{\text{full-load}}, \qquad (i,j) \in \mathcal{L}^1, \tag{6.1}$$

where $\mathcal{L}^1$ is a set containing all links between Pallet Pools and between GSP Production sites and Pallet Pools. The links which do not have transportation restricted to full loads are instead receiving GSP parts pile by pile in the modified model. The reason for nodes that previously received GSP parts in full loads now receiving them pile by pile is that it is not realistic to let a node receive a full load containing only one type of GSP parts. The amount of GSP parts filling an entire full load is often way too big in comparison to the demand at the node. This changes the set of which the constraint (5.1f) is defined over;

$$x_{ijtps^{\text{clean}}} = b_{ip}^{\text{pile}} \cdot y_{ijtp}^{\text{pile}}, \qquad (i,j) \in \mathcal{L}^2, \tag{6.2}$$

where $\mathcal{L}^2$ is a set of links between Pallet Pools and destination nodes receiving clean GSP parts pile by pile. The modifications change the formulation of the objective function 5.7a to

$$\begin{aligned} \text{minimize} \quad \sum_{t\in\mathcal{T}} \sum_{p\in\mathcal{P}} \Bigg(& \sum_{(i,j)\in\mathcal{M}^{7\text{a}}} c_{ijp}^{\text{full-load}} \cdot y_{ijtps^{\text{clean}}}^{\text{full-load}} + \sum_{(k,l)\in\mathcal{N}^{7\text{a}}} c_{klp}^{\text{full-load}} \cdot z_{kltp} \\ & + \sum_{(m,n)\in\mathcal{O}^{7\text{a}}} c_{mnp}^{\text{pile}} \cdot y_{mntp}^{\text{pile}} \Bigg). \end{aligned} \tag{6.3}$$

By making the suggested changes, a modified model can be defined. This model is feasible for any choice of parameters, given that they are somewhat reasonable. The model can always become infeasible when unrealistic parameter values are chosen, such as too small stock limits.

6.2 Testing of the modified model using real data

It is necessary to test the reformulated model by using sets and parameters defining SKF's distribution network in order to assure its correctness. The AMPL code representing the modified model is presented in Appendix B.

When testing the modified model using real data, a special characteristic occurs due to the model's ability to scrap dirty GSP parts. If a Pallet Pool's stock is larger than its maximum limit, it is intended to send clean GSP parts to other Pallet Pools, but there is a way for the Pallet Pool to avoid this requirement. The model is always trying to find the lowest total cost of transportation and purchasing given that the demand at each node is satisfied. If the transportation cost to a node, which is sending dirty GSP parts back to the Pallet Pool, is low, it could be worth creating huge flows of clean GSP parts to that node, which will respond by sending almost all parts back to the Pallet Pool to not exceed its stock maximum. When dirty GSP parts arrive at the Pallet Pool, a certain percentage of them are scrapped. In this way the Pallet Pool is decreasing its stock level without transporting anything to other Pallet Pools.

This is not a desired befall and to prohibit it from happen there are two ways to approach the problem. Either the scrapping parameters' value can be set to 1 for all nodes that are receiving and sending clean and dirty GSP parts respectively, or the costs of transporting GSP parts to those nodes are increased to the point where it will be too costly for this scenario to happen. If the scrap rate's value is set to 1 the amount of scrapped GSP parts in the model will be lower than the amount scrapped in SKF's network.

The second alternative would be a cost increase on these links. A suitable cost increase would be to choose the largest cost existing between two Pallet Pools in the network and add an extra small cost to be on the safe side. These costs is then add to the transportation costs on the links in question. The extra small cost is added to the largest costs between Pallet Pools because the transportation costs on some of these links are zero, and a scenario where it cost the same when choosing either scenario is not wanted. By adding the same cost to all the links it will still exist a difference in costs between different links and the property that some links have lower costs than others will be unaffected. Which approach to the problem is best regarding to the model's solvability is difficult to say, wherefore all scenarios tested are concerning both alternatives.

Data for the pile sizes were collected using the questionnaire; see Appendix C. It turned out that most Pallet Pools use the same number of parts in a pile, but some use there own standards. Since the Pallet Pools are not using uniform pile sizes it could be suitable to test whether the model's solvability changes if uniform pile sizes are used. These instances of the problem are therefore also tested.

The set $\mathcal{T}$, which defines the time span, has not yet been determined and how many time steps the set can consist of can only be found through testing. In section 3.6 the complexity of the problem was introduced, and depending on the size of the problem the complexity property is noticeable to different extent. By increasing the number of time steps the set T can be defined. After a certain time steps the size of the problem makes it impossible to find a solution in a reasonable time or using reasonable amount of computer resources.

If the amount of time steps where the complexity property becomes an issue is very small, it can be justified to make further simplifications of the problem in order to decrease the problem size.

The most suitable simplification of the modified problem is to relax the integrality requirements on GSP parts transported in piles. Since almost all transportation of clean GSP parts are carried out in piles, the relaxation will result in a significant change in the relation between continuous and integer variables in the model. The effect the integer relaxation of the variable has on the result depends on the size of the flows present. If the size of the flows are much larger than the pile sizes, the relaxation will not affect the model's result significantly. If, on the other hand, the flows are small, the affect would be large.

6.3 Number of variables in the models

The original model contains ten types of variables. Two of these, s_{itp}^{change} and s_{itp}, are, however, functions of other variables. This reduces the types of variables in the model to eight. All variables have between 3 and 5 different indexes which means that there exists a large number of variables in the models. In Table 6.2 the number of continuous, integer, and binary variables for one time step is stated for the three models introduced. To obtain the total amount of variables used in the models these numbers are multiplied by the number of time steps in the set $\mathcal{T}$.

Table 6.2: The amount of variables of different types in the three models. "(var/t)" denotes the number of variables present at each time step and the percentage stated in the table is the percentage of the total amount of variables present at each time step for each type of variable.

Variable type	Original model	Modified model		Modified model, relaxed	
	(var/t)	(var/t)	%	(var/t)	%
Continuous	954	954	60	1197	75
Integer	302	627	40	384	25
Binary	52	0	0	0	0
Total	1308	1581	100	1581	100

When the original model is modified all binary variables are deleted and many integer variables are added. When changing the transportation requirements in the modified model many links where changed from transporting in full loads $y_{ijts^{\text{clean}}}^{\text{full-load}}$ and $y_{ijts^{\text{dirty}}}^{\text{full-load}}$ to pile by pile y_{ijtp}^{pile} for transportation of clean GSP parts. When deleting the requirements on transporting dirty GSP parts in full loads containing a mix of GSP parts, binary and integer variables are deleted. Transporting clean GSP parts are, however, depending on the type of GSP part after the modifications are made. Adding some variables and deleting some result in an increased total amount of variables in the modified model. Relaxing the integrality requirements on y_{ijtp}^{pile} will not result in any new variables, only a change in the proportion between continuous and integer variables. This makes the model easier to solve since the B&B tree size depends on the number of integer and binary variables in the model, as described in Section 3.4.

6.4 Result from the modified model using real data

In Section 3.6 it was mentioned that the solvability of different instances of a problem having the same size can be different. Changing some of the parameters' values can therefore affect the solvability of the model and in this section the results of the instances tested and discussed in Section 6.2 are presented.

6.4.1 Modified model

Table 6.3 presents the results obtained when testing the modified model. When setting the value of the scrap rate parameter to 1, using the real costs at the links which was used by Pallet Pools to enable large scrapping, and using non-uniform pile sizes, the model required more than 20 minutes to solve one time step of the instance to find optimum. This is an ominous result since it is desirable to solve the model for 12 time steps. When the number of time steps was increased to two, the computer capacity restricted the solver from verifying a solution with gap smaller than 24.95 %. For three time steps the solver could not solve the problem at all due to lack of computer capacity. When the pile sizes were uniform, the solvability of the model changed marginally. Since this did not contribut to a more solvable model, no further testings using uniform pile sizes were performed. None of the instances could however provide a solution for the desired amount of time steps.

When using the real scrap rates and increasing the costs on the links, the obtained solution for one time step possessed a gap of 0.07 %. A solution for more time steps was not obtained due to lack of computer capacity. This instance of the problem has shown to be harder to solve than the previous one and a solution could not be obtained for the desired number of time steps.

Table 6.3: Results obtained when testing different instances of the modified model.

Time steps	Scrap rate		Cost		Pile sizes		Gap
	Set to 1	Real	High	Real	Uniform	Real	
1	x			x		x	0
2	x			x		x	24.95 %
3	x			x		x	-
1	x			x	x		0
2	x			x	x		21.17 %
3	x			x	x		-
1		x	x			x	0.07 %
2		x	x			x	-

6.4.2 Relaxed modified model

To obtain a result derived from a model defined over more than two time steps, a relaxation of the integrality requirements on y^{pile}_{ijtp} was required. In Table 6.5 solutions to two instances of the problem, with the integrality requirement on y^{pile}_{ijtp} relaxed, is presented. For 12 time steps, the solutions obtained possess a gap of 0.41 % and 0.67 %, respectively, for the two instances, which shows that the objective value for these two instances of the relaxed modified model is not sensitive to the chosen approach of dealing with the scrapping issue.

Table 6.4: Results obtained when testing two instances of the modified model when the integrality requirements of y^{pile}_{ijtp} are relaxed. The first instance had scrap rate set to 1, real costs for links having problems with scrapping, and uniform pile sizes and the second instance had real scrap rate, increased costs for links having problems with scrapping, and uniform pile sizes.

Time steps	Scrap rate		Cost		Pile sizes		Gap
	Set to 1	Real	High	Real	Uniform	Real	
12	x			x		x	0.41 %
12		x	x			x	0.67 %

Relaxing the integrality requirements on y^{pile}_{ijtp} affects the result of the model to different extents depending on the sizes of the flows. To know whether the relaxation is an acceptable approach the sizes of the flows need to be studied. In Figure 6.1 the percentages of the flows of bases, which are greater than 0 but smaller than multiples of the pile size of bases, are presented. The percentages are in relation to the total amount of links concerning y^{pile}_{ijtp}.

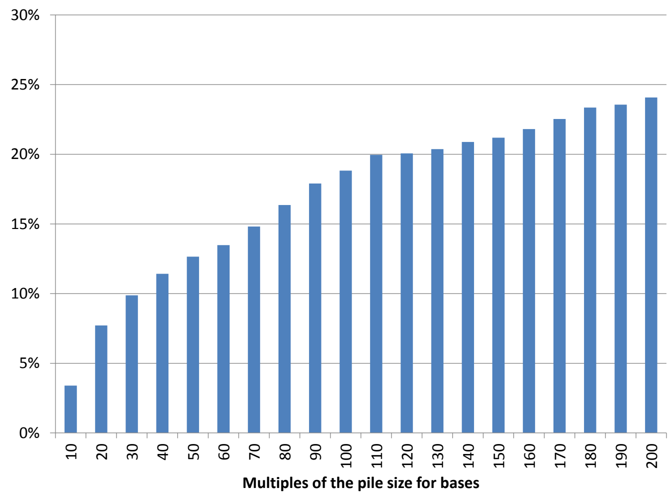


Figure 6.1: The percentage of existing flows of bases which are smaller than a multiple of the pile size of bases. 10 % of the links having a flow of bases possess a flow that is less than or equal to 30 times the size of a pile of bases.

By calculating the percentages of links which have a flow smaller than a multiple of a pile size, an understanding on how big the flows are in relation to a pile size is obtained, which in turn is an indication of whether or not the variables on those links are acceptable to relax. The links of which the flow is 0 are omitted since most of these flows are not affected when relaxing the integrality requirements. There are quite a lot of links having zero flow, which depends on many nodes are supplied by many Pallet Pools which in turn creates a lot of possible links to use. The result is, however, not to use very many of them. The result illustrated in Figure 6.1 shows that only 10 % of the links having a flow of bases possess a flow that is less than or equal to 30 times the size of a pile of bases. Another way to interpret the results is that only 10 % of the links which possessing a flow of bases, have a flow of which the pile size, if such would be used, are 3.3 % of the size of the flow of bases. The same type of data for collars and lids are presented in Figures 6.2 and 6.3, which show that about 10 % of the GSP parts have flows which are less than 30 times the pile sizes. This is considered to be a low percentage and the smallest flows exist on links between Pallet Pools and External Factories. Which Pallet Pools that is supplying which External Factory is predetermined so no optimization in terms of where the flows should be can be made. This makes the relaxation on the integrality requirements on these links affect the result less.

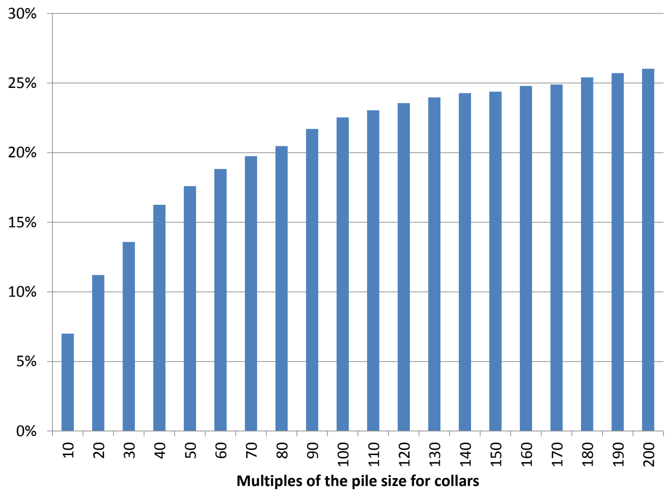


Figure 6.2: The percentage of existing flows of collars which are smaller than a multiple of the pile size of collars. 14 % of the links having a flow of collars possess a flow that is less than or equal to 30 times the size of a pile of collars.

The number of relaxed variables should always be kept at minimum and to test how many variables that actually need to be relaxed and still having a solvable model, the links between Pallet Pools and External Factories were studied. The flows on these links are often quite small, so keeping the integrality requirement on the corresponding variables could improve the solution's correctness. In Table 6.5 the solutions obtained by not relaxing some of the links related to y^{pile}_{ijtp} are presented. The links chosen for keeping the integrality requirements in the first row of the

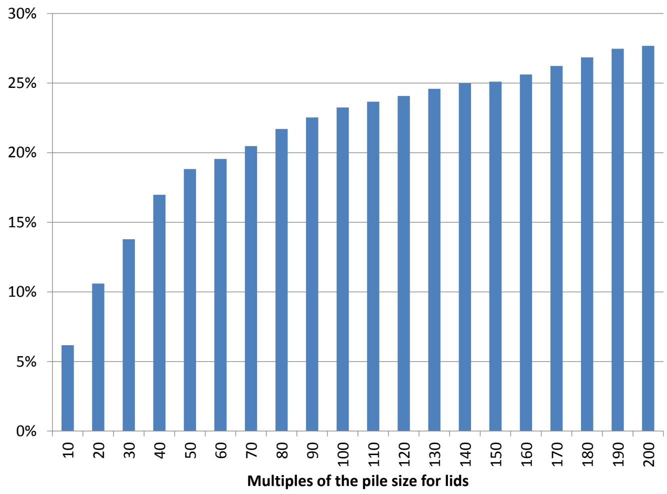


Figure 6.3: The percentage of existing flows of lids which are smaller than a multiple of the pile size of lids. 14 % of the links having a flow of lids possess a flow that is less than or equal to 30 times the size of a pile of lids.

table are flows concerning one External Factory for bases, collars, and lids during 12 time steps, in total 36 links, and the second row presents the result obtained by keeping the integrality for links concerning External Factories, in total 324 links.

Table 6.5: Results obtained when testing different instances of the modified model when only some of the integrality requirements of y^{pile}_{ijtp} are relaxed.

Time steps	# $y^{\text{pile}}_{ijtp^{\text{clean}}}$	Scrap rate		Cost		Pile sizes		Gap
	not relaxed	Set to 1	Real	High	Real	Uniform	Real	
12	36		x	x			x	13.45 %
12	324		x	x			x	21.37 %

For both cases the solutions obtained have large gaps. These gaps are too big to accept and this result shows that an integrality relaxation of y^{pile}_{ijtp} is needed on all links where clean GSP parts are transported pile by pile.

7

Discussion

The conformity of the results to the real world problem is heavily dependent on the accuracy of the data that they are based on and the simplifications made in the model formulation phase. The results have been evaluated both considering the accuracy of the data and the correctness of the model. They have then been compared to how SKF Logistics Services is managing the flows today.

7.1 Data accuracy

In order to receive a result that conforms well with the real world problem the data accuracy is crucial. The model created in this project comprise many parameters which resulted in an extensive data collection from numerous sources. The accuracy of the different types of data varies depending on the source, availability, and method used. A summary of the data accuracy is present in Table 7.2.

7.1.1 Cost parameters

The values for the cost parameters, $c_{ijp}^{\text{full-load}}$ and c_{ijp}^{pile}, were acquired from SLS Business Support Transport department [XVI]. The department uses three different methods for retrieving the costs which depend on the data availability. As mentioned in Chapter 1 a simplification of the distribution network was made to include nodes at a country level only. If more than one node of a certain category is present in a country, these nodes are aggregated into one virtual node. The virtual node was assumed to be located at the site of the largest of the aggregated nodes, and the related costs were calculated accordingly. The reason for this location decision is to reduce the effects of the simplification.

Clean GSP parts are sent in piles within SKF's distribution network and the piles' volumes and weights affect the transportation costs. At SKF, transportation costs are stated in currency per weight unit which mean that the volume of the piles needs to be translated into weight units. The transportation costs was calculated pessimistically. This means that either the real weight or the volume stated in weight units of the piles where used depending on which would give the highest cost. In this project the chosen currency is €, which was a recommendation from SLS.

To some links in the network the related costs have been centrally negotiated. This applies to both transportation and purchasing costs where such are present. The links having these centrally negotiated costs concerns GSP parts in use. When clean GSP parts are transported on the same routes as GSP parts in use, the transportation costs of clean GSP parts are derived from the rate sheets containing the centrally negotiated costs. Since Pallet Pools are always situated at the same site as SLS Warehouses, there is a lot of links whose costs are stated in the rate sheets. 80 % of all cost parameters were given data retrieved from the rate sheets which are considered to be accurate.

For links whose costs could not be found from the rate sheets, estimations of the costs had to be made. When the cost of one link was known, but the costs of the link with opposite direction was unknown, the cost of these links where assumed to be equal. Due to built-in imbalances in the global transport system, costs can depend on the direction of the flow, but since these variations are assumed to be small this assumption does not introduce a too large loss of accuracy. 7 % of all cost parameters where estimated this way. The remaining 13 % of the cost parameters where estimated through qualified guesses. Many of the costs on these links where estimated by comparing them to other links with known costs and with origin and destination nodes situated close to the start and end nodes of the unknown links. If this estimation was impossible to make, other kinds of qualified guesses were made by SLS [XVI]. Estimates were sometimes stated in other currencies than €. SKF has centrally stated exchange rates which are established once a year which were used when transforming the currencies into €.

When transporting full loads, all costs related to trailer freight were retrieved from the rate sheets, which means that all links transporting full loads using trailers have accurate costs. The estimated costs related to transportation of full loads are primarily made at links where no significant amount of flow exists today where mainly sea freight is used. The costs related to these links where however necessary to add to the model since the model requires costs of all links where flow of clean GSP parts can be present.

7.1.2 Flows of GSP parts in use

The parameters representing the flow of GSP parts in use, $f_{ijtps^{\text{in-use}}}$, are the most data-intensive in the model. The main portion of this data was retrieved from SKF's information systems TrMS and Logistic Charge Cube, see Table 7.1 for specification. To be able to distinguish between the registered data acquired from the systems, it is necessary to know both the origin and destination of the flows and the category of the associated nodes. This information was not available for all of the tracked flows which made part of the data impossible to utilize in the model, which are refereed to as losses in Table 7.1. For flows with only one node category present in the corresponding destination/origin country, the data from the systems could be used even when some information was missing by using an exclusion method. All flows to/from such countries was then assumed to be associated with that single node. The quality of the data caused that some of the data tracked by SKF's information systems was lost in the process; this reduce the accuracy of the solution in comparison with the real world problem.

When the project was initiated there existed a conviction that data of all the flows present in the model was available in SKF's information systems. As the project progressed and the work of retrieving the data was intensified, it appeared that this was not the case. The flows from External Factories to SLS Warehouses and SKF Factories was impossible to capture due to internal company policies. Since these flows were considered too large to be excluded from the model, a method was needed to receive appropriate estimates of the flows. By asking the Pallet Pools about the amount of clean GSP parts they send to each External Factory and assume that the inbound and outbound flows of GSP parts are equal at External Factories a fairly good estimation could be made. A problem arose when determining the destinations of the outbound flows. This knowledge is not present centrally at SLS and could not be captured elsewhere, wherefore SLS has to make some qualified guesses regarding these flows [I]. In Table 7.1 are these flow stated with a degree of estimation of 100 %. The accuracy of these guesses are considered to be as high as possible since SLS is the most suitable organization at SKF for making assumptions about entities in this area.

The flows of GSP parts in use with close connection to the monetary flow are better tracked than others. This is because the company uses this data when charging. The flows to Customers and from SKF Factories to SLS Warehouses are examples of relations where the material and monetary flows are correlated why the quality of this data are assumed to be rather accurate.

Table 7.1: Summary of the data accuracy for GSP parts in use, $f_{ijtps^{\text{in-use}}}$. Estimates refer to the percentage of the flows that have been estimated. Losses refer to the percentage of the flows that were lost due to lack of precise specification of the data.

From	To	Estimates (%)	Losses (%)	Source
SLS Warehouse	Customer	0	0	TrMS
External Factories	SLS Warehouses	100	0	[I]
External Factories	SKF Factories	100	0	[I]
SLS Warehouses	SKF Factories	0	4.8	TrMS
SLS Warehouses	SLS Warehouses	0	10.0	TrMS
SKF Factory	SLS Warehouses	0	0	Charge Cube

7.1.3 Data retrieved from the questionnaires

The answering frequency was 91 % in questionnaire. Some of them could not give answers to all of the questions. Therefore some assumptions have been inevitable to make which decreases the accuracy of the result from the solution to the model. There is also a doubt, as in all surveys, on how the receivers interpret the questions and how they answer, which also reduces the precision of the result. To reduce this element of uncertainty an email conversation succeeded the questionnaire in order to clarify the questions asked and get a better understanding of the answers received. The data collected from the questionnaire concern the parameters q_{itp}, r_{ijtp}, s_{ip}^{min}, s_{ip}^{max}, and b_{ip}^{pile}.

Data for the scrap rate, q_{itp}, was missing at many Pallet Pools and 64 % of the parameters have data acquired from the questionnaire. The stated reason why they were unable to provide the data was that the activity associated with scrapping is outsourced to other companies. An estimation for the values of the lacking parameters was necessary to make. By removing the outliers from the received data and then calculate the average for bases, collars, and lids, respectively was the value of the estimates determined.

The return rate, r_{ijtp}, was calculated as the ratio between the GSP parts that a Customer receives, where the data originates from SKF's information systems, and the GSP parts the Pallet Pools stated that they received from each Customer. The Pallet Pools do not know the return rates from the Customers since they do not know how many GSP parts each Customer have received, they only know how many parts the Customers return to their Pallet Pool. Data for 10 % of these parameters were not possible to receive through the questionnaire, wherefore an assumption was made from using the respective yearly return rates for the year 2010 and apply these also for 2011. The yearly return rate of 2010 was collected centrally at SLS [I].

The hardest question for the Pallet Pools to answer concerned the stock limits. Appropriate constant stock limits – in absolute numbers – were asked for, one maximum value s_{ip}^{max} and one minimum value s_{ip}^{min}, but the Pallet Pools are used to determine their stock level depending on the demand of clean GSP parts a few weeks ahead. The answers to this question in the survey are therefore based on judgement which may have the effect that different people respond in different ways due to personal preference. Consistency of the data can therefore be missing, even though the assumption of constant stock limits is accurate since the demand of products are not varying much throughout the year. For the 27.3 % of the Pallet Pools that did not provide these limits an assumption for the stock limits was made by SLS [I].

The parameter b_{ip}^{pile} represents the number of GSP parts in a pile. For the 21 % of the Pallet Pools which did not answered this question the parameter was given the same value as the median of the retrieved answered. The most frequently answered value is also what SLS see as standard, which makes this estimate reasonable [I].

7.1.4 Data acquired from SLS

Some parameter values were searched for centrally at SLS. The reason for this choice of source was to gain a consistency between the transport related parameters.

A trailer and a container which are of equal size, are considered as standard loads in the global transport network. An assumption that these standard loads are used in SKF's distribution network is hence reasonable [XV]. The collection of data of all load sizes used on all links in SKF's distribution network was not an option due to the huge number of links and especially since this is controlled by the transport providers. The values of the parameters $b_{ijp}^{\text{full-load}}$, which represent the number of piles in full standard loads, was stated based on information gained from SLS Sweden [X]. These numbers where then applied for all links in the distribution network. The number of piles in a load is independent of the status of the GSP parts, but dependent on the sending and receiving nodes. This is because the number of piles in a full load depends on whether sea or land freight is used.

SLS's strive to keep a good relation with its suppliers of new GSP parts which in the model resulted in the parameters $d_{ip}^{\min}$. The data for these parameters are collected centrally at SLS and reflects its purchasing process well, why the parameters have high accuracy [I].

Table 7.2: Summary of the data accuracy. The right column refers to the percentage of the total number of parameters of each type which were necessary to estimate.

Parameters	Estimates (%)	
$c_{ijp}^{\text{full-load}}$	0	(from GSP Production)
$c_{ijp}^{\text{full-load}}$	40	(between Pallet Pools)
c_{ijp}^{pile}	36	
q_{itp}	36	
r_{ijtp}	10	
$s_{ip}^{\min}$	27	
$s_{ip}^{\max}$	27	
b_{ip}^{pile}	21	
$b_{ijp}^{\text{full-load}}$	70	
$d_{ip}^{\min}$	0	
Total	26	

7.2 Model accuracy

To be able to create a solvable model comprising 12 time steps many simplifications had to be made. The simplifications made are the following:

- Only nodes on country level are included.
- Full loads contain only one type of GSP parts.
- Transportation of clean GSP parts are carried out in full loads only to Pallet Pools and all other transportation of clean or dirty GSP parts are carried out part by part.

The decision to include nodes on country level only is made because SLS was not interested in knowing the flows of GSP parts at a more detailed level. It was also important in order to decrease the number of nodes in the network, which reduces the size of the MILP model. This simplification affects the model in the definition of the cost parameters. As previously mentioned the nodes of the same node category in a country are merged to one virtual node which make the cost parameters differ somewhat from reality. When defining the scope of this project SLS made clear that they are mostly interested in analyzing the flow of clean GSP parts between the Pallet Pools and being able to decrease the amount of new GSP parts purchased. To optimize the flow between other nodes was considered merely as a bonus. With this in mind it could have been a good idea to stop the level of detail on an even higher level than the country level. If more nodes are merged together, so that each Pallet Pool is only connected to one node from the categories SKF Factories, SLS Warehouse, and Customers, it would substantially reduce the size of the model and the individual demand that each Pallet Pool is facing would still be present. The information losses resulting from a simplification like this would only affect flows between nodes which was not of big interest in the first place. After revealing all the simplifications required to solve the problem by defining nodes on a country level, it would probably have been a better approach to decrease the number of nodes in the model instead. This could make the relaxation of the integrality requirements on the variables y_{ijtp}^{pile} unnecessary. Loosing information of the flows on a country level might be in favor of relaxing the integrality requirement on piles.

The mixture parameter, which was initially introduced in Section 5.1.4, was abandoned since it caused an infeasibility of the model. Therefore it was also necessary to drop the requirement of transporting full loads of clean GSP parts between nodes which are not Pallet Pools or GSP Production sites. The reasoning leading to this decision was based on the fact that it is not realistic to send full loads containing only one type of GSP parts to SKF Factories and SLS Warehouses. A full load containing only one type of GSP parts can carry a large amount of parts which might greatly exceed the demand of parts at the nodes. The excess GSP parts received will be sent back to the Pallet Pools if the excess amount exceeds the stock maximum. It is, however, realistic to send loads containing a single type of GSP parts between the Pallet Pools since the they possess a greater demand.

Relaxing the integrality requirements on y_{ijtp}^{pile} is of course affecting the result of the model, but it is not considered to affect it too much. Most of the existing flows of GSP parts consists of a number of parts which is quite large compared to the pile sizes, which makes the error occurring due to the integer relaxation less prevalent. The relaxation of the integrality requirement is therefore considered acceptable.

7.3 Comparing the result to how SKF manages the flows today

The chosen problem instance is to set the scrap rate parameter value to the real value, to set the costs on links where scrapping becomes an issue to a high value, use the real pile sizes, and to relax the integrality requirements on y_{ijtp}^{pile}. Results from solving this instance of the problem is obtained and different aspects of the results are compared to the current situation at SKF in order to see whether the result is reasonable. One of the aspects studied is the total number of bases, collars, and lids that each Pallet Pool sends to other Pallet Pools each year. The comparison showed that the obtained result did not differ much from the numbers of GSP parts that the Pallet Pools are sending today, which indicates that the model reflects the real world problem well concerning to the overflow of GSP parts at each Pallet Pool. The total demand of GSP parts at each Pallet Pool was also studied by summarizing the inbound flow of bases, collars, and lids that each Pallet Pool receives from either other Pallet Pools or from GSP Production units. Comparing these numbers to the current numbers, a conclusion can be made that this aspect of the problem also reflects SKF distribution network well. Comparing the inbound and outbound flows of GSP parts at each Pallet Pools confirms that the data used in the model reflect SKF's distribution network well and that the result can be analyzed with this assuredness.

The result showed, however, that the distribution of the overflow of GSP parts in some parts of the network and the losses in other parts is not distributed in the same way as SLS manage the flows today. Substantial differences are present for the flows of some GSP parts between some Pallet Pools and these differences were analyzed in order to understand the result. The distribution of the number of each type of GSP part that each Pallet Pool purchased within a year was analyzed which showed that this distribution also differed substantially from the real world scenario. To understand these differences the purchasing prices for each GSP part was compared. A conclusion could be made that the difference in flows between the model's result and the real world scenario depends on the purchasing prices. It is always more expensive to purchase new GSP parts than to transport existing ones when only transportation and purchasing costs are considered. This means that the purchase price will always have a large impact on the directions of the flows.

The purchasing prices of GSP parts are not the main decision factors on where the flows should be directed at SLS today. Besides transportation and purchasing costs there are many other factors involved in the decision process; such as risk factors involved when choosing the supplier base, political factors, and the capacity of the suppliers. The result showed that the GSP Production sites which offer the cheapest purchasing costs are supplying the vast majority of new GSP parts in the network. In the result, the Pallet Pools which can purchase new GSP parts from these GSP Production units fulfill their entire demand of GSP parts by purchasing new ones. The opposite effect can be seen at the Pallet Pools which could only purchase GSP parts from

GSP Production units with high prices. To minimize the total cost in the network all overflows of GSP parts were sent from Pallet Pools with overflow to these Pallet Pools, which could only purchase GSP parts to a high cost. This result is very reasonable but makes the flows in the network unevenly distributed. A full implementation of the result at SKF would not be possible due to politics, relations to suppliers, and, sometimes, environmental impacts. Parts of the result is, however, possible to implement or worth striving for. One, quite easy, implementation for SKF to make is to work more with differentiating the three types of GSP parts. Today, the decisions are mainly taken without much consideration of the different types of GPS parts and the results from the model showed a great difference regarding how the GSP parts should be managed. Working more consciously with each type of GSP parts could improve the handling without making a lot of changes to the decision process.

Summarizing the costs of the total number of purchased GSP parts within a year retrieved from the result and comparing it with what SKF spent on purchasing during the year 2011, the purchasing costs decreased by 8.2 %. Considering the large amount of money that SKF spends on new GSP parts each year, the saving is considerable. The results also show that SLS is managing the returnable packaging system in a proper way today, but many improvement areas exist.

7.4 Further studies

The modified model with no relaxation of the integrality requirements on the variables y_{ijtp}^{pile} was not solvable for more than two time steps. In this project the approach chosen for handling this issue was to relax the integrality requirements on the variables y_{ijtp}^{pile}. One alternative way to obtain a solution to the modified model without relaxing the integrality requirements on y_{ijtp}^{pile} could be to use some kind of decomposition algorithm. Decomposition algorithms can generate a better approximation of the problem's optimal value compared to that obtained from the relaxed version of the problem [18]. This indicates that the modified model could be solved for more than one time step with a smaller gap than what has been found using the chosen solution approach. It is, however, not possible to determine to what degree the solution could be improved without making a deeper study of the model's characteristics. Also, the choice of decomposition algorithm is impossible to determine without any deeper study of the model's properties.

When changing a solution method both advantages and disadvantages arise, which must be dealt with in order to find a solution [19]. The obvious advantage is that there is a chance that even large problem can be solved to optimum. Some of the features of many decompositions algorithm are however not very desirable; for example they often show slow convergence of the gap. Examples of two decomposition algorithms are Column Generation and Benders Decomposition. The efficiency of these algorithms depends heavily on the problem to which they are applied, why a general recommendation is difficult to give [18].

8

Conclusion

QUESTIONING whether or not mathematical optimization can be applied to such a dynamic system as SKF's distribution of returnable packaging is eligible. This project has shown that many simplifications were needed to be able to formulate a linear optimization model and further simplifications were needed to make the model solvable. The results showed that the total inbound and outbound flows of clean GSP parts at Pallet Pools did not differ greatly from the flows present at SKF today, which shows that the results are reasonable. The required simplifications do affect the results, which is still good enough to make an impact of SKF Logistics Service's future management of the distribution of the returnable packaging.

A lot of questions arose when creating the model; some of which were more difficult for SLS to answer than others. These questions initiated further discussions concerning how well the distribution network is described within the company. The complexity of the network makes it difficult to grasp even for people working at SLS and form many business areas at SKF it is important to describe the system in a simple way. The results also creates discussions within SLS concerning how the business area can improve and also the importance of having a well managed system. This project has made managers at SLS more aware about the improvement potentials and the discussions initiated from the presentation of the results made SLS questioning how the system is managed today to a greater extent.

The results from this project will be considered when SLS makes the business plan for 2013, which is the first step into an implementation of some of the improvement areas indicated by the results. A full implementation of the improvement areas suggested by analyzing the result is not likely to be made, but they will act as guidelines for future improvements of the system.

Bibliography

[1] S. Chopra, P. Meindl, Supply Chain Management: Strategy, planning, and operation, Prentice Hall, cop., Upper Saddle River, N.J., 2007.

[2] M. Christopher, Logistics and supply chain management: creating value-adding networks, Prentice Hall/Financial Times, Harlow, 2005.

[3] N. Andréasson, A. Evgrafov, M. Patriksson, An Introduction to Continuous Optimization, Studentlitteratur, Lund, 2005.

[4] J. Lundgren, M. Rönnqvist, P. Värbrand, Optimization, Studentlitteratur, Lund, 2010.

[5] R. Fourer, D. M. Gay, B. W. Kernighan, AMPL: A Modeling Language For Mathematical Programming, Thomson/Brooks/Cole, Cambridge, 2003.

[6] ILOG CPLEX 10.0 - User's Manual (2006).

[7] Anonymous, Issues and impact of reusable packaging, Purchasing & Supply Management (1995) 20–22.
URL http://search.proquest.com.proxy.lib.chalmers.se/docview/224994120

[8] Luton, Can returnable containers work for you?, Materials Management and Distribution (1996) 17–19.
URL http://search.proquest.com.proxy.lib.chalmers.se/docview/227979103/abstract?accountid=10041

[9] S. Randall, Returnable containers simplify supply chain, Modern Materials Handling 53 (13) (1998) 56–57.
URL http://search.proquest.com.proxy.lib.chalmers.se/docview/236551129

[10] D. Maloney, Turning to returnables, Modern Materials Handling 55 (7) (2000) 40–47.
URL http://search.proquest.com.proxy.lib.chalmers.se/docview/2365794089

[11] D. Twede, Can you justify returnables?, Logistics Today 40 (4) (1999) 85–93.
URL http://search.proquest.com.proxy.lib.chalmers.se/docview/233560816

[12] T. F. Bewley, Advances in Economic Theory: Fifth World Congress, Cambridge University Press, Cambridge, 1987.

[13] Parberry, Lecture Notes on Algorithm Analysis and Computational Complexity.
URL http://larc.unt.edu/ian/books/free/lnoa.pdf

[14] A. Schrijver, Theory of Linear and Integer Programming, Wiley, Chichester, 1998.

[15] O. Goldreich, Computational complexity: A conceptual perspective. Illustrated edition, Cambridge University Press, Cambridge, 2008.

[16] S. Martello, P. Toth, Knapsack problems: Algorithms and computer implementations, Wiley, Chichester, 1990.

[17] K. Lumsden, Logistikens grunder, Studentlitteratur, Lund, 2006.

[18] L. S. Lasdon, Optimization Theory for Large Systems, Dover Publications, Mineola, New York, 2002.

[19] M. Galati, Decomposition methods for integer linear programming, Ph.D. thesis, Lehigh University (2010).

A

List of interviews

[I]	Mattias Axelsson	Global Packaging Manager
[II]	Chris Bunn	SLS North America
[III]	Sandro Carminati	Packaging Manager SLS Italy
[IV]	Caroline Descolas	Packaging Manager SLS France
[V]	Mike Fan	Packaging Manager SLS China
[VI]	Diego Gabadian	Packaging Manager SLS Uruguay
[VII]	Beatriz Harumi Katsuta	SLS Brazil
[VIII]	Annica Kjos	SLS Sweden
[IX]	Arthur Koh	Packaging Manager SLS Singapore
[X]	Christian Kukuljan	SLS Sweden
[XI]	Jan-Ove Olausson	Packaging Manager SLS Sweden
[XII]	Satish Paralikar	Packaging Manager SLS India
[XIII]	Katiuscia Silva	Packaging Manager SLS Brazil
[XIV]	Ben E Stefanavage	Packaging Manager SLS North America
[XV]	Andreas Tengberg	Contracting Manager
[XVI]	Magnus Wellsted	Business Support Manager
[XVII]	Tomas Winkler	Packaging Manager SLS Germany

B

AMPL

B.1 Model file of modified model

```
#----------#
#   SETS   #
#----------#

# Main sets #
#-----------#
set GSP_PRODUCTION;  # Nodes containing factories producing new GSPs.
set PALLET_POOLS;  # Pallet pools nodes.
set SLS_WH;  # Nodes for SLS's warehouses' locations.
set SKF_FACTORY;  # Locations for SKF's factories.
set EXTERNAL_FACTORY;  # Locations for SKF's suppliers factories, to which SKF sends GSPs.
set CUSTOMER_COUNTRY;  # Countries where SKF's customers are located.

# All nodes set #
#---------------#
set ALL_NODES  = GSP_PRODUCTION union PALLET_POOLS  union SLS_WH  union SKF_FACTORY
union EXTERNAL_FACTORY union CUSTOMER_COUNTRY;

# Sender and receiver sets #
#--------------------------#
# EXTERNAL_FACTORY and SLS_WH and SKF_FACTORY are goods sender.
set GOODS_SENDER  = SLS_WH  union EXTERNAL_FACTORY union SKF_FACTORY;
# PALLET_POOLS and GSP_PRODUCTION can send clean gsp to pallet pool.
set CLEAN_SENDER_TO_PALLET_POOL  = GSP_PRODUCTION  union PALLET_POOLS;
# SLS_WH and SKF_FACTORY and CUSTOMER_COUNTRY can send dirty GSP.
set DIRTY_SENDER  = SKF_FACTORY  union SLS_WH  union CUSTOMER_COUNTRY;
# SLS_WH and SKF_FACTORY and CUSTOMER_COUNTRY.
set GOODS_RECEIVER  = SLS_WH  union SKF_FACTORY  union CUSTOMER_COUNTRY;
# Nodes sending trucks: SLS_WH, SKF_FACTORY and PALLET_POOL.
set TRUCK_SENDER = ALL_NODES diff  CUSTOMER_COUNTRY diff  GSP_PRODUCTION  diff  EXTERNAL_FACTORY;
# Nodes receiving trucks: SKF_FACTORY and PALLET_POOL.
set TRUCK_RECEIVER = ALL_NODES diff  CUSTOMER_COUNTRY diff  GSP_PRODUCTION
diff  EXTERNAL_FACTORY diff  SLS_WH;
# Nodes sending dirty GSP parts to Pallet Pool.
set DIRTY_SENDER_TO_PALLET_POOL within {DIRTY_SENDER,PALLET_POOLS};
```

```
# Set of SKF_FACTORY and SLS_WH #
#-------------------------------#
set SKF_FACTORY_AND_SLS_WH = ALL_NODES diff GSP_PRODUCTION diff PALLET_POOLS
diff EXTERNAL_FACTORY diff CUSTOMER_COUNTRY;

# EMPTY_CLEAN demand sets #
#-------------------------#
# Except GSP_PRODUCTION and CUSTOMER_COUNTRY.
set EMPTY_CLEAN_DEMAND_NODES  = ALL_NODES  diff GSP_PRODUCTION diff CUSTOMER_COUNTRY;

# Sets of pairs with EMPTY_CLEAN #
#--------------------------------#
# All theoretical links between GSP_PRODUCTION and PALLET_POOLS.
set ALL_LINKS_GSP_PRODUCTION_TO_PALLET_POOL =  {GSP_PRODUCTION,  PALLET_POOLS};
# Existing links between GSP_PRODUCTION and PALLET_POOLS.
set EMPTY_CLEAN_FROM_GSP_PRODUCTION_TO_PALLET_POOL within  {GSP_PRODUCTION,  PALLET_POOLS};

# All theoretical links betweens PALLET_POOLS and SLS_WH.
set ALL_LINKS_PALLET_POOL_TO_SLS_WH = {PALLET_POOLS,SLS_WH};
# Existing links between PALLET_POOLS and SLS_WH.
set EMPTY_CLEAN_FROM_PALLET_POOL_TO_SLS_WH  within {PALLET_POOLS,  SLS_WH};

# All theoretical links betweens PALLET_POOLS and EXTERNAL_FACTORY.
set ALL_LINKS_PALLET_POOL_TO_EXTERNAL_FACTORY = {PALLET_POOLS,EXTERNAL_FACTORY};
# Existing links betweens PALLET_POOLS and EXTERNAL_FACTORY.
set EMPTY_CLEAN_FROM_PALLET_POOL_TO_EXTERNAL_FACTORY within {PALLET_POOLS,EXTERNAL_FACTORY};

# All theoretical links between PALLET_POOLS and SKF_FACTORY.
set ALL_LINKS_PALLET_POOL_TO_SKF_FACTORY = {PALLET_POOLS,  SKF_FACTORY};
# Existing links between PALLET_POOL and SKF_FACTORY.
set EMPTY_CLEAN_FROM_PALLET_POOL_TO_SKF_FACTORY within  {PALLET_POOLS,  SKF_FACTORY};

set EMPTY_CLEAN_FROM_PALLET_POOL_TO_EMPTY_CLEAN_DEMAND_NODES
within {PALLET_POOLS, EMPTY_CLEAN_DEMAND_NODES};
set EMPTY_CLEAN_FROM_CLEAN_SENDER_TO_PALLET_POOL
within {CLEAN_SENDER_TO_PALLET_POOL, PALLET_POOLS};
set EMPTY_CLEAN_FROM_PALLET_POOLS_TO_SLS_WH_AND_SKF_FACTORY_NOT_CLOSE_TO_PALLET_POOL
within {PALLET_POOLS, ALL_NODES};

# Set of pairs with EMPTY_DIRTY #
#-------------------------------#
# All theoretical links between CUSTOMER_COUNTRY and PALLET_POOLS.
set ALL_LINKS_CUSTOMER_COUNTRY_TO_PALLET_POOL =  {CUSTOMER_COUNTRY,  PALLET_POOLS};
# Existing links between CUSTOMER_COUNTRY and PALLET_POOLS.
set EMPTY_DIRTY_FROM_CUSTOMER_COUNTRY_TO_PALLET_POOL  within  {CUSTOMER_COUNTRY,  PALLET_POOLS};

# All theoretical links between SKF_FACTORY and PALLET_POOLS.
set ALL_LINKS_SKF_FACTORY_TO_PALLET_POOL =  {SKF_FACTORY,  PALLET_POOLS};
# Existing links between SKF_FACTORY and PALLET_POOLS.
set EMPTY_DIRTY_FROM_SKF_FACTORY_TO_PALLET_POOL  within  {SKF_FACTORY,  PALLET_POOLS};

# All theoretical links between SLS_WH and PALLET_POOLS.
set ALL_LINKS_SLS_WH_TO_PALLET_POOL =  {SLS_WH,  PALLET_POOLS};
# Existing links between SLS_WH and Pallet Pool.
set EMPTY_DIRTY_FROM_SLS_WH_TO_PALLET_POOL within  {SLS_WH,  PALLET_POOLS};

# Transporting set #
#------------------#
set EMPTY_CLEAN_DEMAND_NODES_ONLY_GET_TRUCKS  within {PALLET_POOLS, EMPTY_CLEAN_DEMAND_NODES};
set EMPTY_CLEAN_DEMAND_NODES_ONLY_GET_BUNDLES  within {PALLET_POOLS, EMPTY_CLEAN_DEMAND_NODES};
```

```
# Set of stockkeepers #
#---------------------#
set STOCKKEEPER = ALL_NODES diff GSP_PRODUCTION diff CUSTOMER_COUNTRY;

# INDEX #
#-------#
set STATUS; # Status on the pallet.
set TIME_INTERVAL_STOCK; # Time interval for stock.
set TIME_INTERVAL  within TIME_INTERVAL_STOCK; # Time-steps where the model is being recalculated at.
set TIME_INTERVAL_STOCK_CONTRAINT  within TIME_INTERVAL_STOCK; # Time-steps used in constraints;
set PARTS; # Bases, Collars, Lids.

# Set of stock keepers #
#----------------------#

#----------------#
#   PARAMETERS   #
#----------------#

param bigNumber; # Big number used in constraint for choosing a scenario.
param timeStart; # The first month in the period the model is running over.

# Cost, demand, full goods parameters #
#-------------------------------------#
# Cost transporting full container.
param costGspProductionToPalletPool {GSP_PRODUCTION, PALLET_POOLS, PARTS};
# Cost of transporting a full truck.
param costTruck {EMPTY_CLEAN_DEMAND_NODES_ONLY_GET_TRUCKS};
# Cost of transporting free parts.
param costBundle {EMPTY_CLEAN_DEMAND_NODES_ONLY_GET_BUNDLES, PARTS};
# Ordered goods between goods sender and goods receiver.
param fullGoods {GOODS_SENDER, GOODS_RECEIVER, TIME_INTERVAL_STOCK, PARTS} default 0;

# Return parameters #
#-------------------#
# The percentage of full goods that is returned as dirty to a pallet pool.
param returnRate {CUSTOMER_COUNTRY,PALLET_POOLS, TIME_INTERVAL, PARTS};
# Percentage of part that is not lost at node i in time interval t.
param scrapRate {DIRTY_SENDER, PALLET_POOLS, TIME_INTERVAL, PARTS};

# Stock parameters #
#------------------#
param stockInitial {STOCKKEEPER, PARTS} default 0; # The initial stock.
param safetyStockMinimum {STOCKKEEPER, PARTS}; # Minimum safety stock at a pallet pool
param safetyStockMaximum {STOCKKEEPER, PARTS}; # Maximum safety stock at a pallet pool

# Transport parameters #
#----------------------#
#Number of bundles that fills a full truck from GSP Production to PalletPool.
param numberOfPartsInTruckFromGspProduction {GSP_PRODUCTION, PALLET_POOLS, PARTS};
# Number of bundles in truck
param numberOfBundlesInTruck {TRUCK_SENDER, TRUCK_RECEIVER, PARTS};
# Number of parts in a bundle.
param numberOfPartsInBundle {PALLET_POOLS, PARTS};
# Lowest amount of parts that a GSP production unit will deliver to a PALLET_POOL.
param lowestLimitOfNewPartsDelivered {GSP_PRODUCTION, PARTS};

# Parameters for loading data #
#-----------------------------#
param symbCustomer  symbolic in CUSTOMER_COUNTRY;
param symbTime  symbolic in TIME_INTERVAL;
param symbParts  symbolic in PARTS;
```

```
param symbPalletPool  symbolic in PALLET_POOLS;
param symbGSPProduction  symbolic in GSP_PRODUCTION;
param symbGoodsSeneder  symbolic in GOODS_SENDER;
param symbGoodsReceiver  symbolic in GOODS_RECIEVER;
param symbStockkeeper  symbolic in STOCKKEEPER;
param symbAllNodes  symbolic in ALL_NODES;
param symbTruckSender symbolic in TRUCK_SENDER;
param symbTruckReceiver symbolic in TRUCK_RECEIVER;

#---------------#
#   VARIABLES   #
#---------------#
# Number of items of status s send from node i to j at time t.
var x {ALL_NODES, ALL_NODES, STATUS, TIME_INTERVAL, PARTS}  >= 0;
# Stock change of part p at time interval t.
var stockChange {STOCKKEEPER, TIME_INTERVAL, PARTS};
# The amount of stock a stockkeeper has.
var stock {STOCKKEEPER, TIME_INTERVAL_STOCK, PARTS} >= 0;
# Number of containers used.
var z {GSP_PRODUCTION, PALLET_POOLS, TIME_INTERVAL, PARTS}  integer  >= 0;
# Number of bundles used
var yBundle {ALL_NODES, ALL_NODES, TIME_INTERVAL, PARTS}  integer  >= 0;
# Number of full load used
var yTruck {ALL_NODES, ALL_NODES, TIME_INTERVAL, PARTS} integer  >= 0;

#----------------#
#   OBJECTIVES   #
#----------------#
minimize TotalCostObjective:
sum {(i, j) in EMPTY_CLEAN_FROM_GSP_PRODUCTION_TO_PALLET_POOL,  t in TIME_INTERVAL, p in PARTS}
costGspProductionToPalletPool[i,j,p] * numberOfPartsInTruckFromGspProduction[i,j,p] * z[i,j,t,p] +
sum {(i, j) in EMPTY_CLEAN_DEMAND_NODES_ONLY_GET_TRUCKS,  t in TIME_INTERVAL, p in PARTS}
costTruck[i,j] * yTruck[i,j,t,p] +
sum {(i, j) in EMPTY_CLEAN_DEMAND_NODES_ONLY_GET_BUNDLES,  t in TIME_INTERVAL, p in PARTS}
costBundle[i,j,p] * yBundle[i,j,t,p];

#-----------------#
#   CONSTRAINTS   #
#-----------------#

# Stock change #
#--------------#

# SKF_FACTORY. #
subject to settingStockChangeForSKF_FACTORY
{(i,j) in EMPTY_DIRTY_FROM_SKF_FACTORY_TO_PALLET_POOL , t in TIME_INTERVAL, p in PARTS}:
stockChange[i,t,p] = sum {gs in GOODS_SENDER} fullGoods[gs,i,t,p] +
sum {(pallet,i) in EMPTY_CLEAN_FROM_PALLET_POOL_TO_SKF_FACTORY} x[pallet, i, "EMPTY_CLEAN", t,p] -
sum {g in GOODS_RECIEVER} fullGoods[i,g,t,p] - x[i,j,"EMPTY_DIRTY",t,p];

# EXTERNAL_FACTORY. #
subject to settingStockChangeForEXTERNAL_FACTORY
{i in EXTERNAL_FACTORY, t in TIME_INTERVAL, p in PARTS}:
stockChange[i,t,p] = sum {(pallet,i) in EMPTY_CLEAN_FROM_PALLET_POOL_TO_EXTERNAL_FACTORY}
x[pallet,i,"EMPTY_CLEAN",t,p] - sum {g in GOODS_RECIEVER} fullGoods[i,g,t,p];

# SLS_WH. #
subject to settingStockChangeForSLS_WH
{(wh,thePallet) in EMPTY_DIRTY_FROM_SLS_WH_TO_PALLET_POOL, t in TIME_INTERVAL, p in PARTS}:
stockChange[wh,t,p] = sum {gs in GOODS_SENDER} fullGoods[gs,wh,t,p] +
sum {(pallet,wh) in EMPTY_CLEAN_FROM_PALLET_POOL_TO_SLS_WH} x[pallet, wh,"EMPTY_CLEAN",t,p] -
sum {gr in GOODS_RECIEVER} fullGoods[wh,gr,t,p] - x[wh,thePallet,"EMPTY_DIRTY",t,p];
```

```
# PALLET_POOLS. #
subject to settingStockChangeForPALLET_POOLS
{pallet in PALLET_POOLS, t in TIME_INTERVAL, p in PARTS}:
stockChange[pallet,t,p] = sum {(clean,pallet) in EMPTY_CLEAN_FROM_CLEAN_SENDER_TO_PALLET_POOL}
x[clean,pallet,"EMPTY_CLEAN",t,p] + sum {(dirty,pallet) in DIRTY_SENDER_TO_PALLET_POOL}
(scrapRate[dirty, pallet,t,p] * x[dirty,pallet,"EMPTY_DIRTY",t,p]) -
sum {(pallet,e) in EMPTY_CLEAN_FROM_PALLET_POOL_TO_EMPTY_CLEAN_DEMAND_NODES}
x[pallet,e,"EMPTY_CLEAN",t,p];


# Update stock and setting its limits. #
#--------------------------------------#

# Updates the stock, t=1. #
subject to updateStockInitial {i in STOCKKEEPER, p in PARTS}:
stock[i,timeStart+1,p] = stockInitial[i,p] + stockChange[i,timeStart,p];

# Updates the stock, t>1. #
subject to updateStock {i in STOCKKEEPER, t in TIME_INTERVAL_STOCK_CONTRAINT, p in PARTS}:
stock[i,t+1,p] = stock[i,t,p] + stockChange[i,t,p];

# Lowest safety stock limit. #
subject to settingSafetyStockMinimum {i in STOCKKEEPER, t in TIME_INTERVAL, p in PARTS}:
stock[i,t+1,p] >= safetyStockMinimum[i,p];

# Highest safety stock limit. #
subject to settingSafetyStockMaximum {i in STOCKKEEPER, t in TIME_INTERVAL, p in PARTS}:
stock[i,t+1,p] <= safetyStockMaximum[i,p];


# Setting EMPTY_DIRTY from Customer country #
#-------------------------------------------#

# Turn FULL_GOODS to dirty at CUSTOMER_COUNTRY. #
subject to fullGoodsToDirtyAtCUSTOMER_COUNTRY
{(c,p) in EMPTY_DIRTY_FROM_CUSTOMER_COUNTRY_TO_PALLET_POOL, t in TIME_INTERVAL, part in PARTS}:
returnRate[c,p,t,part] * sum {wh in SLS_WH} fullGoods[wh,c,t,part] = x[c,p,"EMPTY_DIRTY",t,part];


# Transport contraints from GSP_PRODUCTION #
#------------------------------------------#

# Number of gsp in a container is equal to the number of gsp in a bundle times number of bundles.
subject to numberOfGspInAContainer
{(gsp,pallet) in EMPTY_CLEAN_FROM_GSP_PRODUCTION_TO_PALLET_POOL, t in TIME_INTERVAL, p in PARTS}:
x[gsp,pallet,"EMPTY_CLEAN",t,p] =
numberOfPartsInTruckFromGspProduction[gsp,pallet,p] * z[gsp,pallet,t,p];

subject to lowestLimitOfNewPartsDeliveredFromGspProduction
{gsp in GSP_PRODUCTION, p in PARTS}:
sum {(gsp,pallet) in EMPTY_CLEAN_FROM_GSP_PRODUCTION_TO_PALLET_POOL, t in TIME_INTERVAL}
x[gsp, pallet, "EMPTY_CLEAN", t, p] >= lowestLimitOfNewPartsDelivered[gsp, p];


# Transport constraint for EMPTY_CLEAN from PALLET_POOLS #
#--------------------------------------------------------#

# To EMPTY_CLEAN_DEMAND_NODES which only get full loads. #
subject to transportingEmptyCleanFromPalletPoolTrucks
{(i,j) in EMPTY_CLEAN_DEMAND_NODES_ONLY_GET_TRUCKS, t in TIME_INTERVAL, p in PARTS}:
x[i,j,"EMPTY_CLEAN",t,p] =
```

```
numberOfPartsInBundle[i, p] * numberOfBundlesInTruck[i,j,p] * yTruck[i,j,t,p];


# To EMPTY_CLEAN_DEMAND_NODES which only get bundles. #
subject to transportingEmptyCleanFromPalletPoolBundles
{(i,j) in EMPTY_CLEAN_DEMAND_NODES_ONLY_GET_BUNDLES, t in TIME_INTERVAL, p in PARTS}:
x[i,j,"EMPTY_CLEAN",t,p] = numberOfPartsInBundle[i, p] * yBundle[i,j,t,p];
```

C

Questionnaire

Hi,
we are doing our master's thesis at SKF where we study the flows of GSPs around the world under supervision of Mattias Axelsson. The project's objective is to give SKF Logistics Services in the area of Packaging a better decision basis and to find areas of improvements for how global flows of GSPs can be handled. In order to do so we are creating a mathematical model which needs to be filled with data. Some data can of course be found in existing systems, but some is missing. Therefore we would like you to fill out the attached excel sheet with the data asked for in the questions below. If you find any question unclear or have other comments, don't hesitate to contact us. We would appreciate to receive your answers as soon as possible, **but no later than the 20th of April**.

Volume of returned GSPs

1. What is the monthly volume of returned bases, collars and lids from each market (country) stated during 2011 and the first three month during 2012?

Piles

2. What are the quantities in a pile of bases, collars and lids send from your Pallet Pool?

3. What are the dimensions (width, depth, height) in centimeter of each pile?

Flows of uncleaned GSPs

4. From which different SKF Factories and SLS Warehouses does your Pallet Pool receive uncleaned GSPs? Do also state in which countries these units are located.

5. If your Pallet Pool receives uncleaned GSPs from SKF Factories and SLS Warehouses abroad, do they always return pile-by-pile or do they return piles filling a full truck load?

Average scrap rate (not repairs)

6. What are the scrap rates for bases, collars and lids for each month during 2011 and the first three month during 2012?

Stock of empty and clean GSPs

7. Define an estimated absolute number for the upper and lower limit of your stock of clean bases, collars and lids during 2011 and the first three month during 2012. Do also define a +/- span for the variation of the estimated stock limits.

Data of empty and clean GSP flows

8. We need the historical monthly data during 2011 and the first three month during 2012 of the empty and clean GSP flows from your Pallet Pool to all current destinations, specified per base, collar and lid. Do also state in which countries these destinations are located and if it is a supplier to SKF Factories, a SKF Factory or a SLS Warehouse. If the destination is a supplier to SKF Factories please state, if you have any idea, which SKF Factories it is supplying.

Transport cost of empty and clean GSPs

9. What is the cost of sending a pile of bases, collars and lids to the stated domestic destinations? If it exists more than one physical location in each category, state the transport cost to the largest facility.

Thanks for your answers!
Annelie and Frida