Tampere University of Applied Sciences

Optimizing mixed box packing problem
with an in-house built application

Tu Nguyen

BACHELOR'S THESIS
May 2019
Degree Programme in International Business
Option of Green Supply Chain Management

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Degree Programme in International Business
Option of Green Supply Chain Management

Tu Nguyen
Optimizing mixed box packing problem with an in-house built application
Bachelor's thesis 50 pages, appendices 08 pages
May 2019

The three-dimensional knapsack problem aims to find the combination of given rectangles or boxes into a larger rectangle or box of fixed dimension. The optimization is an NP-Hard problem, which has no known algorithm can solve it in a polynomial time. This paper presents the case of knapsack problem in pallet combinatorial optimization at Automotive Aftermarket Division at Robert Bosch GmbH Japan. The division outsourced their packing process to a service provider called Provider. The palletization at Provider is done intuitively based on experiences. The author proposed to develop an in-house software that will determine the most optimal combination of boxes to pack in pallet box constraints to minimize waste of pallet capacity. A packing algorithm that considers up to 5 of volume, non-overlapping, box orientation, non-over packing and weight was proposed. The software was later developed based on the algorithm with Python language with GUI Tkinter. An additional Excel file with open source code from University of Bath was introduced to handle the visualization and animation of packing pattern.

Key words: bin-packing, optimization, automotive, knapsack

CONTENTS

1 INTRODUCTION 5
2 COMPANY BACKGROUND AND KNAPSACK PROBLEM 6
  2.1 Company background 6
  2.2 The Knapsack Packing Problem 9
    2.2.1 The construction of Part Number (PN) and Bill of Material (BOM) 9
    2.2.2 Packaging Master Data Project and the discovery of KPP 15
    2.2.3 General combinatorial optimization problem in the packaging line 17
    2.2.4 Knapsack Packing Problem description and visualization 20
3 LITERATURE REVIEW 23
  3.1 The global optimization approaches. 23
  3.2 Specified algorithms used in 3DPP 27
    3.2.1 Multi-pallet loading model 27
    3.2.2 The First Fit Decreasing (FFD) Algorithms 28
    3.2.3 A formulation for mixed-size box packing 28
    3.2.4 Large Area First Fit (LAFF) Algorithm 29
4 METHODOLOGY 30
  4.1 Phase 1- Elicitation 31
  4.2 Phase 2- Designing 32
    4.2.1 Software Architecture Pattern 33
    4.2.2 Software Process Model 34
    4.2.3 Interface Design 35
    4.2.4 Component Design 36
    4.2.5 Data Structure 37
    4.2.6 Algorithm Design 38
  4.3 Phase 3- Execution 39
    4.3.1 Developing Kombination 39
    4.3.2 Output, Testing and Acceptance 41
5 DISCUSSION 44
  5.1 Overall outcomes 44
  5.2 Evaluation 45
    5.2.1 Positive points 45
    5.2.2 Limitations and recomendations 46
  5.3 Company feedbacks 47
REFERENCES 49

APPENDICES ... 51
Appendix 1. New Grads Training Division Info Session + Workshop ... 51
Appendix 2. World Pallet Size ... 54

1 INTRODUCTION

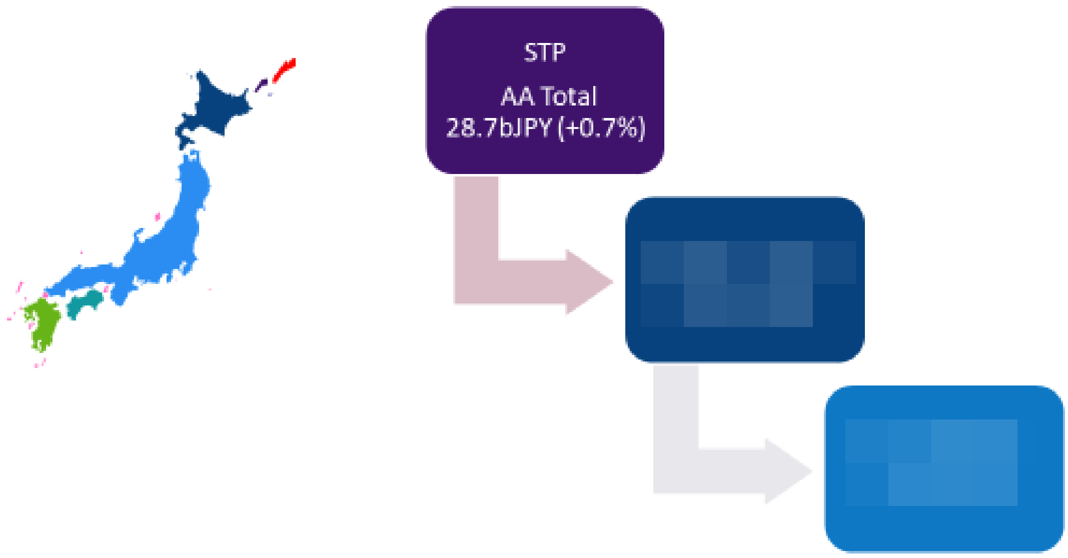
Knapsack Packing Problem (KPP), the main target of this paper is a specified case of Bin Packing Problem (BPP). The original Bin Packing Problem could be explained by given a set of n items of different weights and bins with capacity c, the problem solver needs to assign all n items into the minimum amount of bin. With the constraint of only one bin, KPP deals with finding the quantity of each item with given weight to put into one bin and maximize the capacity of that bin (Fukunaga & Korf 2007).

These two problems are niche cases of combinatorial optimization which found in many different fields especially resources allocation and packing. The solution to these problems has a wide range of application including: loading container/ bin/ cabin/ truck/ pallet, allocating data onto storages, tasks scheduling or project management. In the framework of this paper, the author takes into consideration the knapsack problem for loading pallet/bin in automotive industry. In other words, the research question is "How to reduce waste of capacity in packaging in industrial context". The author aims to provide a quantitative solution to KPP, following with an additional Excel CLP Solver to visualize the packing pattern for each combination. This solution aims to minimizing the waste of space in packing, provide better control and insight over packing process at warehouse and reduce packaging cost by minimizing wasted space.

2 COMPANY BACKGROUND AND KNAPSACK PROBLEM

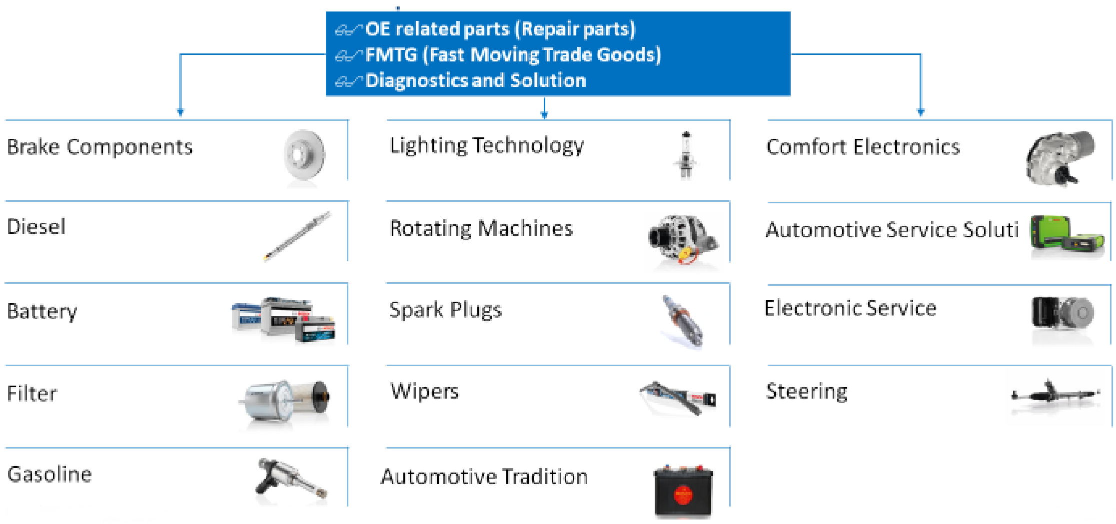
2.1 Company background

Robert Bosch GmbH, headquartered in Gerlingen, near Stuttgart, Germany is a world leading multinational engineering and electronics company. The group's business is divided into four core sectors: Mobility Solutions, Industrial Technology, Consumer Goods, and Energy and Building Technology (Bosch Deutschland 2018). The Automotive Aftermarket division, with 18,000 associates around the world, manages the provision, logistics and sale of automotive spare parts, workshop equipment and Bosch products for retrofitting. Its services also include technical customer service for automotive products and systems. Automotive Aftermarket Division Japan (AA-JP) generated a revenue of 28.7 million JPY (2016), equivalent to 22.5 thoudsand Euro and is constantly growing (Bosch Automotive2018). See appendix 1 for more details.



PICTURE 1. AA Japan - facts and figures (Bosch Automotive2018)

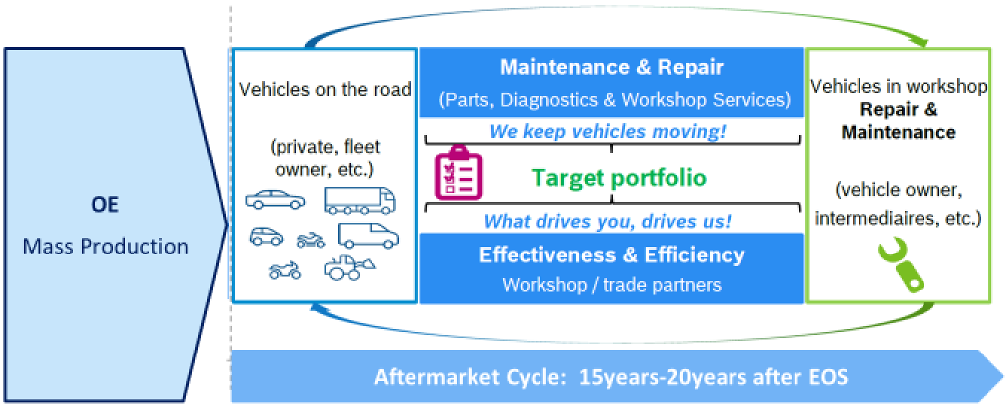
A combination of German pattern and Japanese craftmantship has made Automotive Parts from Japan favored products around the world.



PICTURE 2. AA product portfolio in Japan (AA Japan 2018)

Automotive Parts produced in Japan includes Automotive Aftermarket Parts (battery, wiper blades, AC filters), Testing Equipment (scan tool, battery tester, engine analyzers) circle around the world, especially to AA headquarter warehouse in Karlsruhe, Germany.

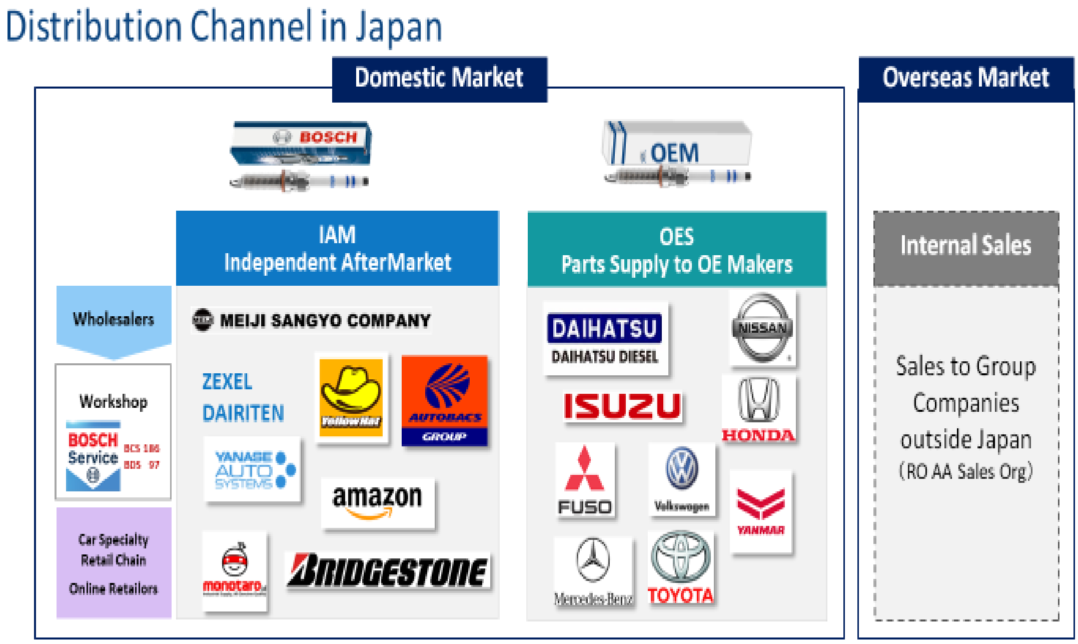
Looking at the product life cycle in the automotive industry, vehicles are born at Original Equipment (OE) Mass Production. A lot of Automotive Parts from Bosch are supplied to car-makers at this phase. The business belongs to another division rather than AA. After the moment of sales, also known as End of Sales (EOS), the ownership of the vehicle is transferred to the customer, there starts the product lifespan of 15 to 20 years.



PICTURE 3. Aftermarket product life cycle (Bosch Automotive 2018)

During this lifetime, the vehicles need to be maintained and repaired to keep up with legal, safety and efficiency standards (Roland Berger 2013). Many parts of the vehicle needed to be replaced frequently. For example: wiper blades need to be replaced every 12 months, not to mention seasonal change for snowy location; air filter needs to be changed at least every 3 years; brake pads and battery equally need to be changed every 4- 5 years (Dodson's Japanese Partsworld 2017) and so on. AA Products shown in picture 2 are sold to workshops, car-makers, retailers during this phase.

The products are circled into both oversea and domestic markets thought 2 different channels called Original Equipment Manufacturer (OEM) and Independent Aftermarket (IAM). In OEM, genuine car parts that Bosch produces for a car-maker are sold directly to workshops, retailers under the name, packaging, label of the car-makers. The IAM market, on the other hand, is independent car parts that are produced by Bosch, sold to workshops and retailers in Bosch packaging.



PICTURE 4. AA distribution channel in Japan (Bosch Automotive 2018)

The list of OEM customers includes Daihatsu, Customer Y, Isuzu, Honda, Fuso, Volkswagen, Mercedes-Benz, Yanmar and other oversea car-makers. For the IAM market, the company sells their products to Meiji, Zexel, Yellow Hat, Autobacs, Amazon, Yanase Auto Systems, Bridgestone and Monotaro. The trio Meiji, Autobacs and Yellow Hat are the three most popular retailers in the Japanese auto-parts market. One special case is Bosch Car Service. This customer is workshop but classified as a separate wholesale customer since it belongs to the corporation.

Take a BMW car as an example, after the customer purchases the vehicle, during the lifespan of that vehicle, when the customers need to replace a part, can choose the genuine part from BMW which is provided through OEM channel. But in case customers need to change the wiper blade, they can choose to pick a pair of wiper blade from any brand at the retails store, the Bosch wiper blades are provided through IAM channel.

2.2 The knapsack packing problem

The author worked at the company from May 2018 until April 2019 as an intern at Market Business Coordination Team. During this period, the author directly worked in two projects that serve an SAP project. They are Warehouse Management System (WMS) and Packaging Master Data (PMD) projects. Within PMD, there are three sub-projects: Master Data, Bill of Material Creation and Confidential. While working with these projects alongside with regular task of the team, the author learned about the supply chain in general, saw the KPP and convinced her manager to try a new approach in solving this problem. To understand the root of the problem and how the author came about with a solution for KPP, the following sections explain the basis of Part Number, the target outputs of each project mentioned above and where the problem lies along the chain.

2.2.1 The construction of Part Number (PN) and Bill of Material (BOM)

At grass-root level, data are collected, cleaned and controlled based on Part Number (PN). Each PN is a combination of Product Number and Index.
The Index itself is a combination of Country Code and Customer Code. In short, we can put the explanation of PN in equation:

$$PN = Product\ Number + \underbrace{(Country\ Code + Customer\ Code)}_{\text{Index}}$$

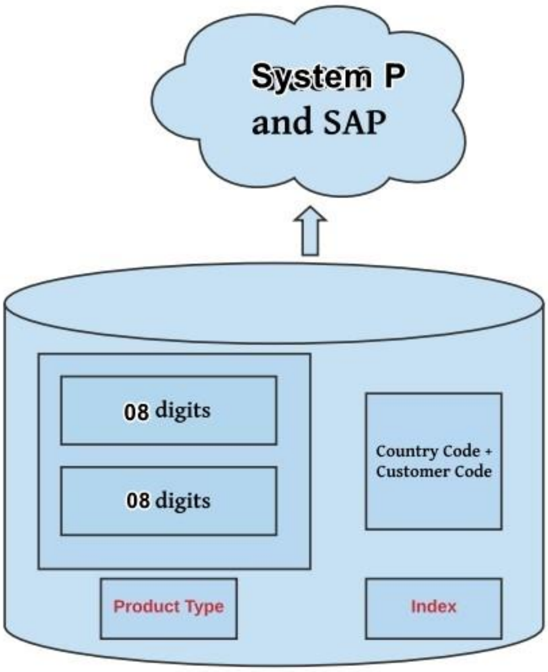
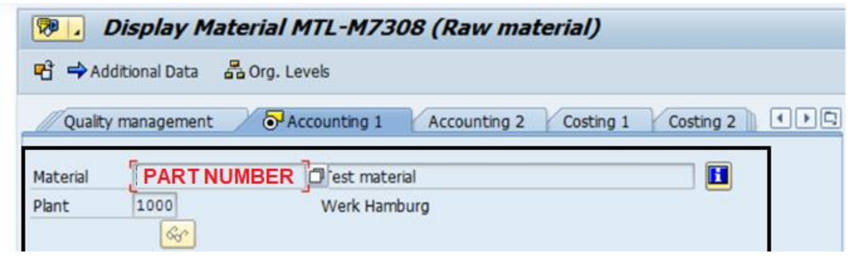


FIGURE 1. Structure of Part Number in the database

Each distinctive PN belongs to one product type and one customer in a specific country. Attached to each PN is several different types of information such as: Country of Origin (CoO), Production Date, Sales Channel, Customer Part Number etc. when handling works, for example, they are treated as Material Numbers in Material Management (MM) Module of SAP.



PICTURE 5. Part Number is treated as Material Number in MM Module, SAP

Each PN then being packed into different types of packaging based on its Product Type and Index. Product Type decides which type of packaging material is used for a certain PN while Index decides which type of label, pallet is used when delivering products.
For example, the auto-part in picture 6 belongs to Product Type: conventional injector pump (distributor head), an essential part of diesel engines.

PICTURE 6. Distributor injection pump (Bosch AA 2018)

This product is made of metal with a protective oily layer, a slow-moving product, therefore, it requires a special kind of packaging that prevents corrosion and oxidation. Each distributor head is wrapped with volatile corrosion inhibitor (VCI) paper, polyethylene (PE) bag, corrugated carton box level 1. VCI technology is considered one of the most advanced and cost friendly method to use for metal products, especially auto parts in ocean shipping (Chen, Xie, Yang & Yang 2015, 1).

The Index then will decide which kind of label to use, based on Customer Code and Country Code. For example, a distributor head produced has a 10 digits Product Number of F1234567 that need to be delivered to Customer X U.S.A, Customer X Japan and Customer Y Japan. The Index of three customers are listed in the table below:

TABLE 2. Customer Code, Country Code and Index

Customer	**Country**	**Index**
Customer X	Japan	562
Customer X	U.S.A	7ZN
Customer Y	Japan	4PQ

Indexes are taken for demonstration purpose only. Information is not real figures.

From the information above, we can tell that for the same distributor head, we have three different types of packaging. F1234567562 and F12345677ZN might have the same packaging, depends on requirements from customer in each location, but the labels are definitely different. F12345674PR on the other hand, have totally different packaging content from the other two because it goes to Customer Y. Another example, in picture 7, NISSAN GROUP Label for Genuine Parts made in Japan looks different from TOYOTA Label for Lexus Genuine Parts made in China, sold in North America Market.

PICTURE 7. NISSAN GROUP and TOYOTA label for GENUINE PARTS

The delivery destination denoted in Index also decide the type of pallet and sealing to use. When delivering to different continents, the company need to comply with local legal regulations about pallet sizes. Take Germany as an example, when delivering car parts from Japan to Germany, the consignments must be complied with regulations established by Verband der Automobilindustrie (VDA), German Association of the Automotive Industry. It includes transportation on VDA pallet (VDA 4530 2013) with or without pallet box. The VDA pallet has a different set of dimensions from other types of pallet or Japanese standard pallet in particular.

TABLE 3. Standard size of pallet used Germany and Japan in (mm)

Country	Width	Length
Japan	1100	1100
Germany (Europallet)	1200	800

Two pallets chosen for demonsation purpose. See appendix 2 and references for more details.

PICTURE 8. Two different types of pallet used for Customer X

PICTURE 9. Two different types of pallet used for Customer Y

PICTURE 10. Euro pallet meets VDA standard with EPAL mark (VDA 2018)

In picture 8 and 9, for some Product Types delivery to their plants in Japan, Customer X and Customer Y require returnable pallets (both metal and plastic type). These two pallets are among several different types of pallet that needs to be used for Customer X and Customer Y. Other types of pallet including tailored pallet for large and extra-large engine are used sometimes. In picture 10, when delivering to customers in Europe, including Customer X and Customer Y Europe, the company must use VDA Pallet. This wide variaty of pallet used in palletization hinders the difficulties in fulfilling the order without leaving a lot of empty spaces in the pallet box if the packaging agent does not possess adequate sorting and packing method.

According to the Index and BOM, corrugated box level 1 will be packed into level 2 box in a unit of 6, 12, 12 or 36 and then loaded onto pallet. All those components in the package include VCI paper, PE bag, corrugated box level 1, level 2, cushion (cardboard blank, air pillow, bubble wrap etc), label, pallet, seal together form the BOM that can be seen in picture 11. Each product leaves the production line has one particular PN attached to it and hence, being packed according to the respective BOM before being delivered to customer.

packaging data sheet

status bill of material: 0 - current data

part number	index	desig. code	Bosch designation	Lieferwerk

packaging level 1:
packaging level 2:
packaging level 3:
packaging level 4:

*Mindestablieferung oder Vielfaches
minimum delivery quantity or multiple
Quantité de livraison minimum ou un multiple (de cette quantité)
Cantidad de entrega mínima o múltiplos

level	position	Quantity	quantity unit text	pack-material no.	PaGOS-desig.	packing mat. status	length out	width out	height out
1	1		piece		box,	1		77	
1	2		piece		plastic bag		999		9999
1	3		piece		VCI			114	
1	4		piece		pp pap		270		1
2	5		piece		d box, tra	5		320	190
2	6		piece					538	1
1	7		piece		label				1
2	7		piece						1
3	7		piece					25	1
3	8		iece		skeleton container	7		1070	

comment	comment (in english)

SAMPLE

PICTURE 11. Bill of Material attached to one Part Number

Each packaging material also has its name denoted by a 10 digits number in the form of 8xxx-xxx-xxx. They are maintained in System P with dimensions, weight, description, material quality, drawing and ideally but not always unit cost. Dimensions and weight of the packaging material play a crucial role in the quantitative solution for KPP in the later part of this paper.

2.2.2 Packaging master data project and the discovery of KPP

As mentioned in the introduction at 2.2, the author directly worked in two projects at AA-JP during her 1-year internship. Through these projects, the author discovered the problem and determined to find a solution.

In order to work with the SAP project, the company needed to acquire all the information from different systems that were in use at that time (including both external and internal sources), cleansing and organizing the data for further proceed. PMD projects provide the Master Data directly for WMS project and SAP project. Within PMD, there are three smaller projects: Master Data Acquisition, BOM Creation and Confidential.

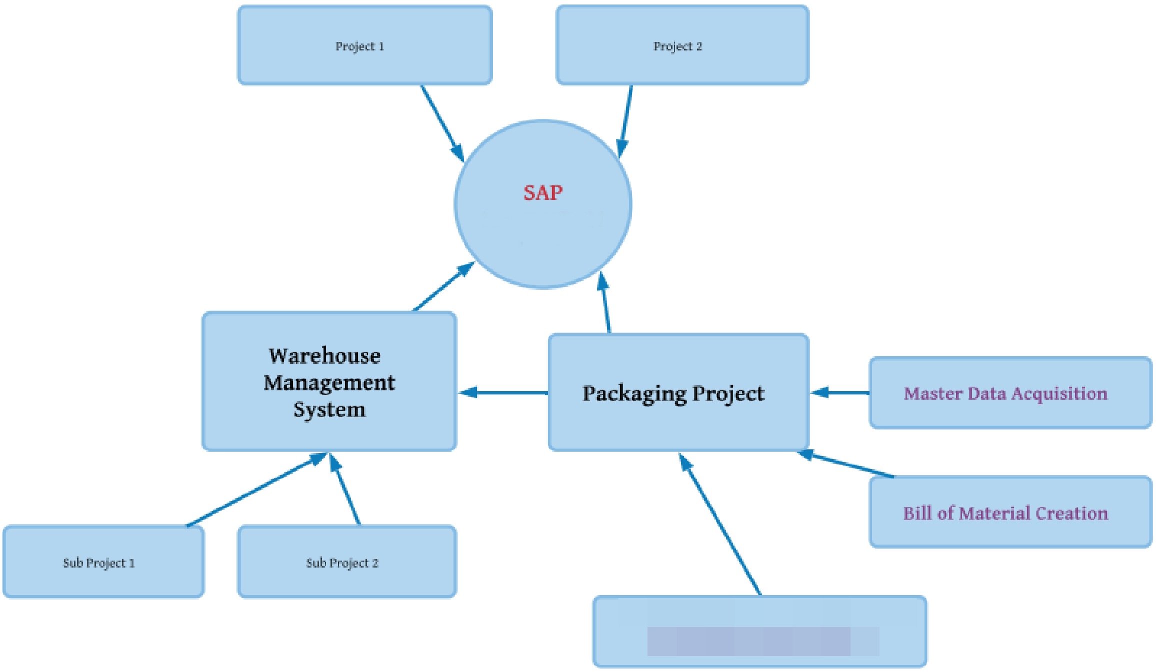


FIGURE 2: An overview of WMS and PMD projects

The company outsources its packaging process to a third-party service provider named Provider. Master Data Acquisition collected all the Master Data of BOM related to currently moving products from Provider, System P, SAP and Customers to retain needed data, abandon prohibited PN and its BOM. The dataset called Master Data then being used to create the BOM and mass-uploaded to System P and SAP in the sub-project BOM Creation. The two projects deal with approximately 50,000 PNs and their packaging material.

During the process, one problem unveiled was: "The palletization process at Provider is done based more on past experiences of the supervisor and packers." Even though Provider has been in the business for nearly 20 years, this intuitive approach with limited concise calculation of loading capacity and efficiency rate lead to unnecessary wastes and related issues such as:

- Sometimes, pallet is delivery not fully loaded (excluding cases where domestic and retail customers order less than pallet load).
- Pallet with mixed size boxes is not packed to the highest possible efficiency.
- Extra cushion needed to be inserted to keep the content of pallet stable. This is an additional cost.
- Due to empty space inside pallet box, products move during transportation and sometimes lead to damaged packaging. This issue leads to customer complaint, low customer satisfaction and potential additional cost for re-packing damaged box.

All the issues mentioned above can lead to the damaging of package and ultimately the product inside. For example, when a pallet is not fully loaded, there is empty space that causes cardboard boxes inside shaking, moving and crashing each other during transportation and handling. Cardboard boxes inside might be scratched, battered or damaged. In the worst scenario, customers might refuse to accept the consignment due to quality concern. The consignment will be delivered back to the packaging warehouse to be repacked with additional workload for both Provider packing team and Quality Control Team. In case of minor damages, customers can accept the consignment but request extra cardboard boxes to re-pack themselves before putting on display. This may seem like a natural and unharmful request from retail stores, however, when the cardboard boxes are printed with company logo or OEM customers logo in case of genuine parts, it might lead to serious legal concerns in the future if any problem occurs at end-customers usage. Either case, it is not acceptable to deliver customers scratched or battered cardboard boxes.

The extra cost carried with cushioning and empty space on each pallet might be not too high. However, the accumulated of extra unnecessary costs throughout the time is not healthy for both operational and financial performance of the entity.

On top all the additional costs and workload is a question of which team is held accountable for the damaged boxes so that it could be changed in the future. When tracking down to the root of the problem, it is often confused whether if it should be handled by BOM management team (MBL) or Logistics team (CLP). For all the possible consequences of poor fitting boxes to pallet, it is necessary to address and tackle this problem.

2.2.3 General combinatorial optimization problem in the packaging line

This section discusses the simplified flow of final products from plants to customers warehouses. It provides both the general picture of the supply chain and detailed insights of each work stations in the packaging process. Thus, it helps to define the most suitable stage to loop the quantitative solution in.

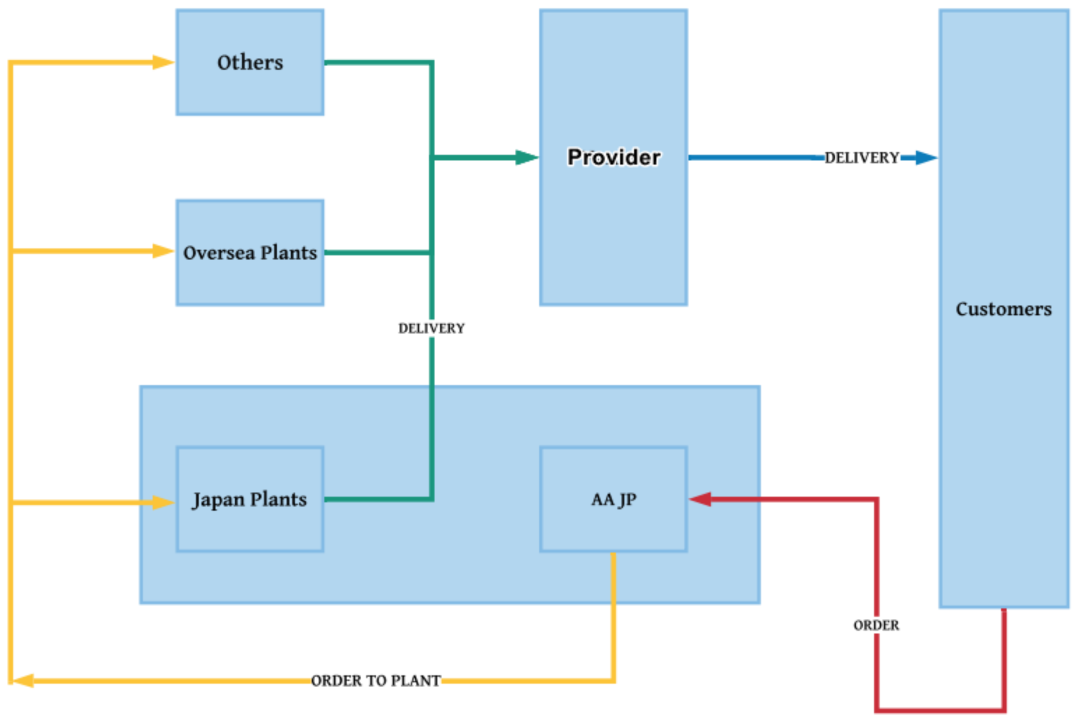


FIGURE 3: Simplified flow of final products from plants to customers.

Figure 3 describes the flow of final products from plants to customers warehouses in 4 different stage denoted with red, yellow, green and blue arrows. This figure does not consider detailed pre-sales activities, information flow, cash flow, demand planning and other operations. At red arrow, customers place order to the Sales Team and the order is created in SAP. Order then place to to plants and warehouses via yellow arrow. Plants and warehouse, after receiving the order, proceed to green arrow by sending products to packaging agent which is Provider. At Provider warehouse, products are packed and palletized according to their BOM. After that, products are delivered to customers warehouse via blue arrow.

Figure 4 below explains the simplified packing process from input to delivering.

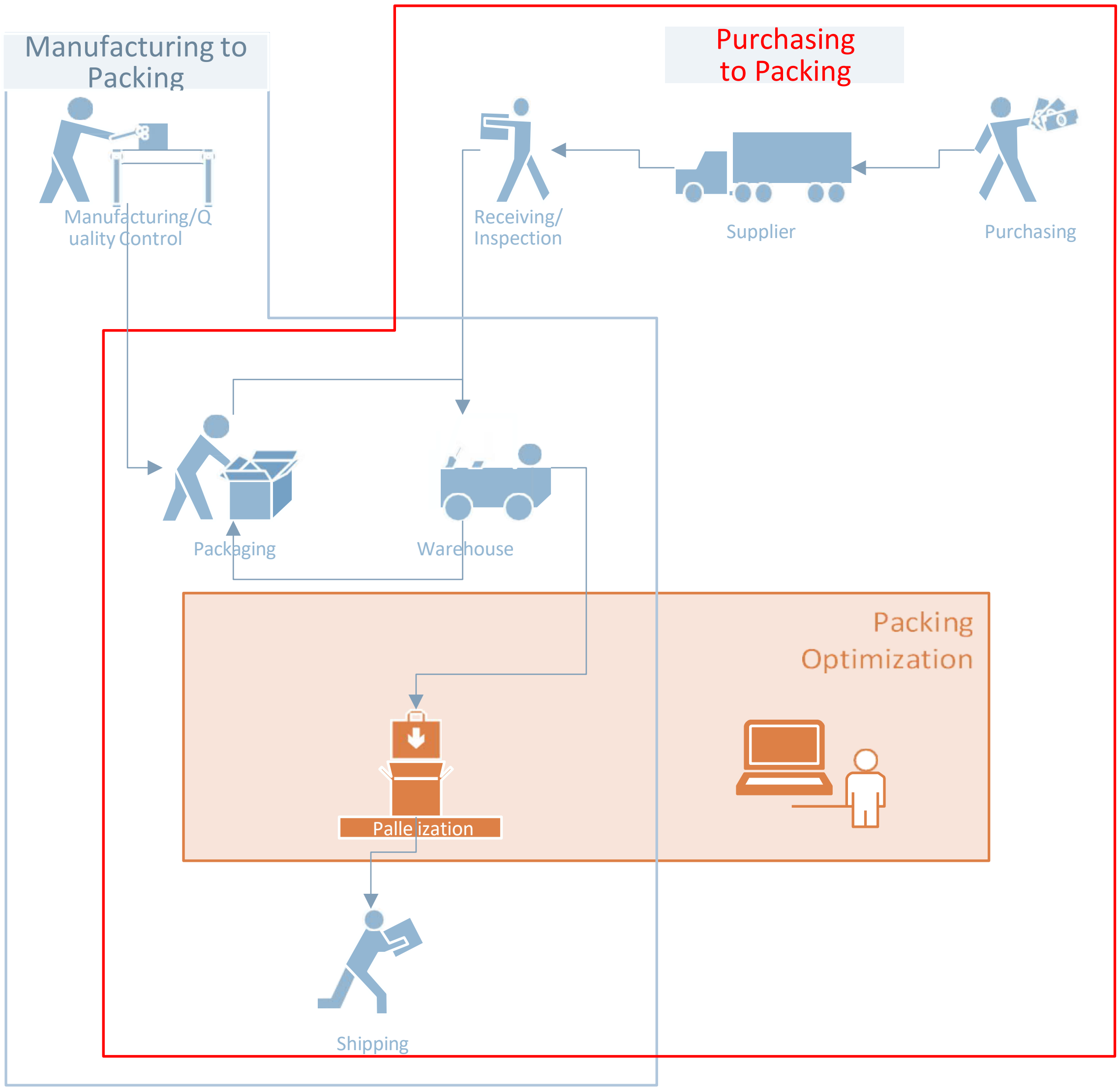


FIGURE 4. Simplified packing process from input to delivering

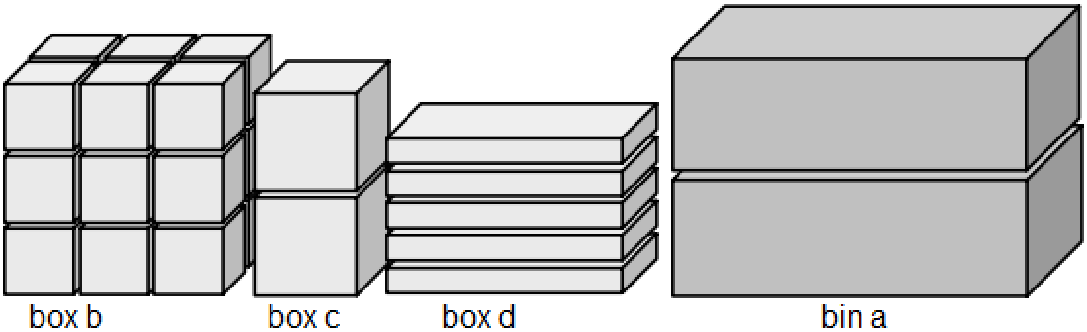
The general process was simplified to reflect only physical packing process. Other stations related to quality control, warehousing process, inventories and information flow are not shown in this figure.

There are three different boxes represent two processes and one focused operation in this figure. The blue lined area demonstrates the process from manufacturing to packing, storing in warehouse, repacking if needed and then palletizing and shipping to customers. The red lined area demonstrates the process of purchasing from supplier, receiving, sorting into warehouse, packing to palletizing and shipping to customers.

In general, after the warehouse in Provider receives products, they pack and deliver immediately in case of fast moving PN. In the case of slow moving PN, Provider may store them in the shelf with respective environment requirements until there is an order. Products then will be packed and delivered. There is a small loop between packing and warehousing before palletization. Depend on the nature of each PN and order, they will be looped out for palletization and delivery either directly after packing or from the warehouse. The two processes overlap from packing-warehousing phase until the very end. From the process, we see that solution to packing optimization can be added at palletization to maximize the used capacity of pallet box/bin without causing disruption for the whole process. The calculation can be done in the office and attached to the order picking form.

At the warehouse of Provider, orders are picked manually with paper form and bar code scanning machine. According to Global Innovation Index in 2018, Japan ranks number 13 as the most innovative country in the world (Cornell University, INSEAD & WIPO 2018, 15). Japan is also often known by the public as a world giant leader in advanced technology. While this notion is true, outside of high-tech field, Japanese people have a very good relationship with paper (washi in Japanese). This relationship dated long back into 610 AD, where paper was first introduced to Japan. Paper is used not only in daily activities but also in ritual, religious ceremonies and paper making is a means of living for many Japanese. Up until this day, Japan maintains the largest circulation of print newspapers in the world, and the second largest per capita (Salva 2018). Literature research combined with the authors closely observation show that, an ordinary Japanese worker at the warehouse are used to and enjoy using paper forms in their daily works. This is a subtle but very important feature to consider when developing and implementing a new method of doing the same task. To support the end-users, who are packing workers at the warehouse to fulfill their job with higher efficiency, problem solvers must think of not only overall benefits but also the habits, difficulties and strengths of end-users.

2.2.4 Knapsack Packing Problem description and visualization



PICTURE 10. A set of bin a and three types of box (Westerlund 2005)

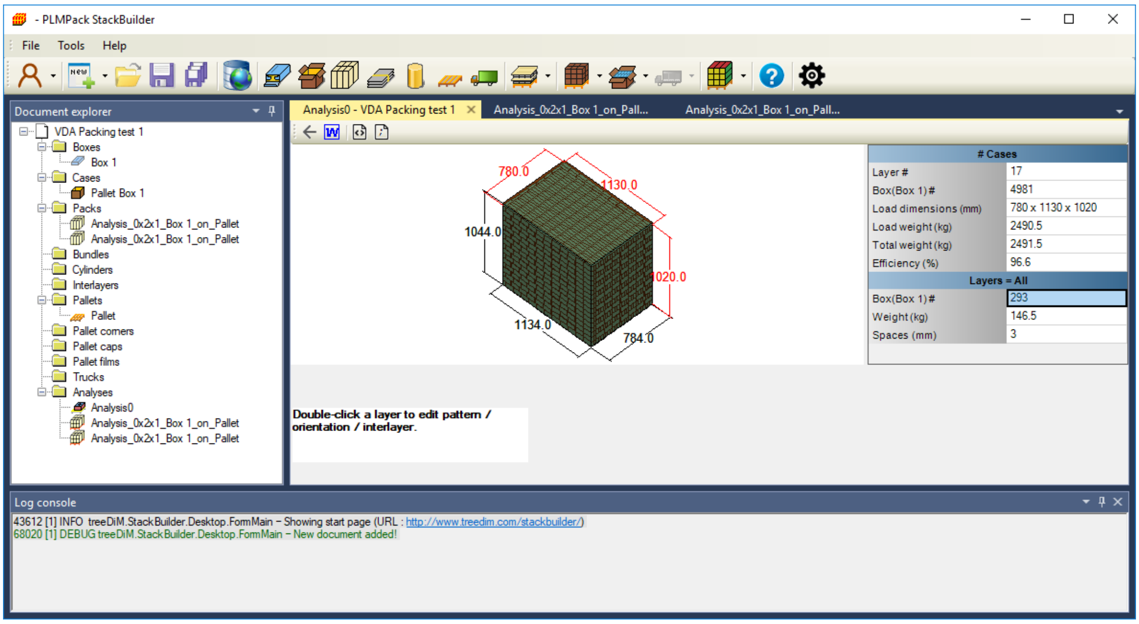
Given bin a in dark gray and set of mixed-size box b, c, and d. In reality, bin a could be any outer box, pallet box or container. The problem is to find the respective quantity of b, c, d that can be packed into bin a and minimize the wasted space of a.

The optimization is classified as an NP-Hard problem, which has no known algorithm can solve it in a polynomial time (Egeblad & Pisinger 2007, 1; Lin, Tsai & Chang 2011, 3-4). There are plenty of tools available on the market that can help tackle this issue. However, these tools are often very costly, technical oriented with complicated interface. For example, the author researched, prototyped and proposed an external solution called Stack Builder by treeDiM in July 2018. Stack Builder is an open source software that help to design and optimize the packing, palletizing (box/ pallet) and shipping items (pallets / truck) (treeDiM 2014). The software has more than 10 special features including:

- Case Palletization
- Packing optimization
- Calculate the optimal packaging system: article/box/pallet
- Search for the optimal packing solutions in your Database of boxes solutions
- Cylinders palletization
- Bundles palletization
- Add interlayers, corners, film,cap
- Databases of boxes solutions
- Filing truck

- Analysis reports.

Different size boxes on VDA pallet size were tested.



PICTURE 11: A trial test of PLMPack StackBuilder

Despite the promising potential and the highly desired feature of “efficiency (%), Stack Builder failed to meet the needs of the company due to several reasons:

- It does not support heterogeneous loading which means no mixed size boxes can be calculated.
- It has an overly complicated interface and many features that end-users do not need or are not able to use accurately.

The author decided to develop an in-house application as a long-term, low-cost, end-user friendly solution for this optimization problem.

As stated earlier, ordinary workers at the warehouse in Provider tend to enjoy using paper and stick to their own systems, it is not worthy to introduce an overly complicated with many technical highlights like Stack Builder. In term of user experiences design, an application that users must go through more than 5 steps to reach final page will not likely to work in long-term (Pierre 2018). In application development, most of the application developed nowadays are not by professional developers but people with expertise in the industry who needs application to support their needs (Ko, Abraham, Beckwith, Blackwell, Burnett, Erwig, Scaffidi, Lawrance, Lieberman, Myers, Rosson, Rothermal, Shaw

& Wiedenbeck 2011). In this case, people with expertise in packing are of course packaging engineers but also the packing workers in the warehouse. An application that does not cater to user habits and experiences is just another waste.

3 LITERATURE REVIEW

Packing optimization has been studied for over a century with application in various field. This paper analyzes various approach from mathematical model base to industry base to finally develop a distinctive set of algorithm that fulfills the need of the company. The literature review section is organized in the reversed pyramid structure which provides a general global view about combinatorial optimization problem, the Bin-Packing problem ranging from one dimension (1D) which deals with only weight constraint to three dimensions (3D) which takes both width, lengtht and height of the related boxes and bin into consideration. Some case studies from different countries and then particularly in industrial environment and the automotive industry are also presented.

3.1 The global optimization approaches.

Combinatorial optimization is an NP-Hard problem, which means non-deterministic polynomial-time hardness. In other words, with NP-Hard problem, the computational complexity of the solution does not depend only on the algorithm but largely on the size of the problem. Which also means that the computational time will increase exponentially with problem size that is determined by the number of variables, computer capacity. It is also worth noting that there is almost no absolute solution for a NP-Hard problem but many approaches using heuristic techniques to find the most acceptable solution in a reasonable amount of time (Lin et al. 2016, 4).

The simplest case of physical packing optimization involves only one parameter (weight). Therefore, we assume that both the boxes and bin are one-dimension bin packing (1DBP). Its goal is to pack n items of weight from $w1, w2, ..., wn$ to a bin of capacity c while fulfilling specific condition such as the fewest number of identical bins capacity c (Fukunaga & Korf 2007, 3). The application of 1DBP can also be found in classic data storage or memory allocation (Bein, Bein & Venigella 2011). The four simplest Bin Packing algorithms are Next Fit, First Fit, Worst Fit and Best Fit. In general, both of four are classified as Sequential Fit Algorithms. The similarity of these four algorithms is that they keep all given items in its exact order and try to pack them, one by one into the bin with different rules as shown in table 4.

TABLE 4. A summary of packing rules in Sequential Fit Algorithms (Malkevitch, 2004)

Type	Packing rule
Next Fit	Pack next item of the list into current bin if its fits. If not, seal the current bin (even not fully packed), and open a new bin to pack.
First Fit	Pack next item of the list into lowest numbered bin, if it does not fit, open a new bin to pack (without sealing other bins).
Best Fit	Pack the next item of the list into the bin that will leave the least empty space after packing that item in.
Worst Fit	Pack the next item of the list into the bin that will leave the largest empty space after packing that item in.

For example, pack 8, 5, 7, 6, 2, 4, 1 into bins of capacity 10, using both four types of sequential fit algorithms result in four different combinations. In figure 5, 6, 7, 8, the white space without number is the empty space inside the bin after completion of packing.

With the Next Fit Algorithm, 8 is put in bin 1. Next item in the list is 5, cannot be fit in bin 1, therefore, bin 1 is sealed and bin 2 is created to pack 5. Next item in the list is 7, also cannot be fit in bin 2, therefore, bin 2 is sealed and bin 3 is created to pack 7. The next item is 6, cannot be fit in bin 3, so bin 3 is sealed and bin 4 is created to pack 6. Next item in the list is 2, fit into bin 4. The last 2 items in the list are packed into bin 5 with the same principle.

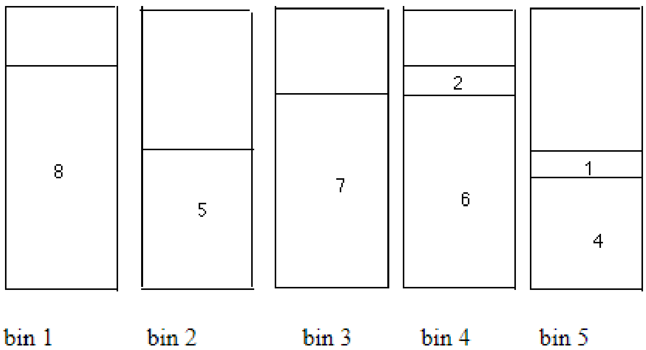


FIGURE 5: Visualization of Next Fit algorithm (Malkevitch, 2004)

With this algorithm, there are 5 bins used in the packing process with a lot of empty space in each bin.

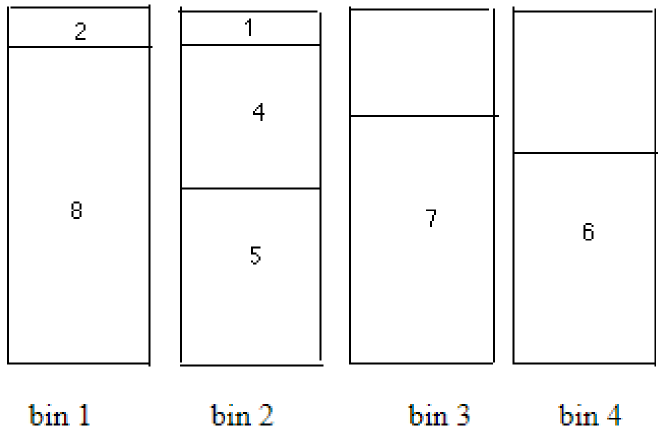


FIGURE 6: Visualization of First Fit algorithm (Malkevitch, 2004)

With the First Fit algorithm, 8 is packed into bin 1. Next item in the list is 5, cannot be fit in bin 1, therefore, bin 2 is created to pack 5 while bin 1 remains open. Similarly, 7 and 6 is packed into bin 3 and 4. The last three items are 2, 1 and 1 are sequentially packed into bin 1 and bin 2 as illustrated in figure 6. This method is clearly more efficient than the Next Fit method since it uses only 4 bins and bin 1, 2 are fully packed.

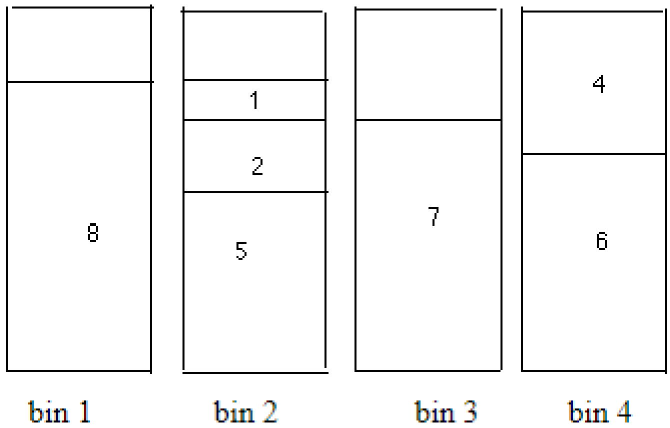


FIGURE 7: Visualization of Worst Fit algorithm (Malkevitch, 2004)

With the Worst Fit algorithm, after 8, 5, 7, 6 are packed into bin 1, 2, 3, 4 like in First Fit, item 2 is packed into bin 2, because after packing, it leave the largest empty space. Item 4 and 1 are packed into bin 4 and 2 respectively on the same principle.

With the Best Fit algorithm, 8, 5, 7, 6 are also packed into bin 1, 2, 3, 4 sequentially.

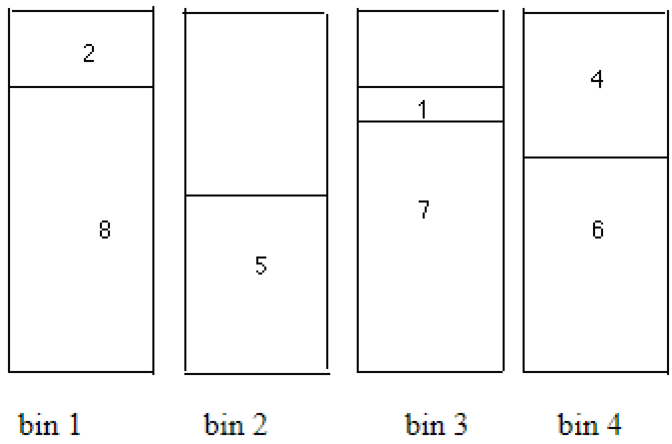


FIGURE 8: Visualization of Best Fit algorithm (Malkevitch, 2004)

With the Best Fit algorithm, 8, 5, 7, 6 are packed into bin 1, 2, 3, 4. After that, 2 is packed into bin 1 because it leaves the least empty space after packing, in this case, the empty space is equal to zero. Similarly, 4 and 1 are packed into bin 4 and bin 3. With the three methods First Fit, Worst Fit and Best Fit, only 4 bins are used to fully pack 7 items.

The next level of complexity in packing optimization is 2D packing problem (2DPP) that considers putting a set of *n* boxes with relative width and height into the minimum amount of fixed bin while ensuring that all boxes are not-overlapping and their edges are in parallel with the edge of the bin that contains them (Lodi 2000, 19). There is a lot of different greedy and hybrid type of algorithms used in solving 2DPP with constraints (bound), three most classical algorithms for 2DPP are derived from the famous Sequential Fit Algorithms of 1DBP. They are Next Fit Decreasing Height (NFDH), Best Fit Decreasing Height (BFDH), First Fit Decreasing Height (FFDH) (Smith 2001). The principle of these 3 algorithms are similar to the same of its name in 1DBP. The only difference is that the list of items in 2DPP is sorted in decreasing height order. Then items are packed according to First Fit, Next Fit and Best Fit rules. Since all the boxes are flattened into 2D, it does not consider the rotation of boxes inside the bin. Similarly, the total weight of bin and weight constraint of the bin are also not taken into consideration.

The advanced version of 2DPP is Three Dimension Packing Problem (3DPP). Of which, 3D-Knapsack Problem (3D-KP) and 3D-Open Dimension Rectangular Packing Problem (3D-ODRPP) have a wide range of application in industrial operation (Tsai, Wang & Lin 2013).

With 3D-KP approach, given a set of *n* mixed-shape boxes with respective width, length, height and bin with its fixed width, length and height. Each box (item) has its own profit value. The problem is to find a sub-set of boxes to be packed into the bin that maximizes the profit value of that packing combination. 3D-ODRPP as mentioned in its name, considers packing a set of *n* rectangular boxes of mixed size orthogonally inside a minimum rectangular bin. Boxes are obviously not allowed to overlap each other (Junqueira & Morabito 2016). A grid-based position formulation (Junqueira & Morabito 2016) installs the 3 dimensions coordinate at one corner of the bin (normally bottom left) with three axis (x,y,z). Each item/ box is placed into the bin according to its set of 3D coordinate. The 3D approach is highly practical in modern industrial processes such as cargo loading, pallet loading, truck loading, ship loading. In the framework of this paper, the author only considers 3D approach to pallet packing optimization at AA-JP

3.2 Specified algorithms used in 3DPP

This section discusses four algorithms used in 3DPP in four different studies. These algorithms have closes relationship to the case study at AA-JP.

3.2.1 Multi-pallet loading model (MPLP)

In MPLP, a consignment of different types of boxes is packed in multi-pallet. Given the dimensions and quantity of each box, the fixed dimension of the bin, the goal of this model is to minimizes the number of pallets that need to be used for the consignment with several constraints such as: order demands, placement position, weight condition, stability and others (Terno, Scheithauer, Sommerweiß & Riehme 2000). This model is considered highly practical especially in case of palletization in the warehouse with the aim to reduce transportation cost while enhancing the stability of the consignment. The solution must either meet the optimal condition or calculated in a given limitation of time.

3.2.2 The First Fit Decreasing (FFD) algorithm

In the First Fit Decreasing, list of boxes/ items is sorted in decreasing order and then packed into the bin based on First Fit rule similar to table 4. The algorithm was used by Beirão (2009) in expedition problem at INDASA, one of Europe's leading manufacturers of coated abrasive systems technology. In their study, researchers proposed two different approaches to solve their Container Loading Problem: the Layer Building Algorithm (LBA) and the Extreme Points (EP). The second approach led to two different algorithms: First Fit Decreasing and Ant Colony Optimization. At the final development phase, Beirão and his team decided to use the Extreme Point First Fit Decreasing algorithm which was a success in their case. Two instances were tested and efficiency were improved by 11.5% in the first instance and 16.6% in the second one.

In a similar paper at University of Bath in 2017, Doctor Gunes Erdogan introduced a VBA Macro in Excel that applies First Fit Decreasing Algorithms. Both two studies based on the condition that quantities of each type of box are known and the algorithm help to put them into designated bin.

3.2.3 A formulation for mixed-size box packing

Another approach to achieve optimal 3D unit allocation in a limited with up to four main constraints of weight, non-overlapping, box orientation and container constraint was introduced by Westerlund and Papageorgiou (2005).
Given:
- A set of *n* boxes of different rectangular sizes
- A set of k identical containers
- Cost data (fixed container costs) container, spatial allocation of boxes)

Determines:
-Which containers to use and detailed layout (which boxes in what direction)

The first author received financial support from TEKES, the National Technology Agency of Finland. The formulation was proven to be effective through two illustrative examples.

3.2.4 Large Area First Fit (LAFF) Algorithm

Another different approach to optimal 3D Packing Unit proposed and tested by Gürbüz et al. (2009) is the Large Area First Fit Algorithm. This algorithm requires users to input the number of different sized boxes and the dimensions (width, depth, height of each type of box). The algorithm sorts all boxes in the order of descending surface area, then put the box with largest surface area in the bin first while minimizing the height of occupied area. Boxes with the second largest surface follow in the sequence. The height of occupied area keeps increasing until it reaches the constraint of the outer bin. This algorithm assumes that the height of the bin is unlimited and all boxes are rotatable. The authors also consider developing other algorithm in the future considering the weight limitation of bin, weight distribution on pallet or the use of multiple bins.

In conclusion, based on the literature review and requirement from AA-JP task requirements, the author develop an application to find an optimal combination of boxes that fit into one fixed bin and maximizing the efficiency of that bin. The working process is presented in Methodology. In this paper, from this point onward, the two terms software and application are sometimes used interchangeably.

4 METHODOLOGY

The application is developed to calculate all the possible combination of a set of mixed size rectangular boxes to fill in the bin while minimizing the wasted capacity. The end-user then can choose which combination to use depend on the actual situation. It is built with Python language based on the basic of Software Engineering from Software Architecture to Software Architecture Pattern and Software Life Cycle Model (Orso 2018) which are described step by step in the following section.

The application was built based on the 7 steps architecture illustrated below.

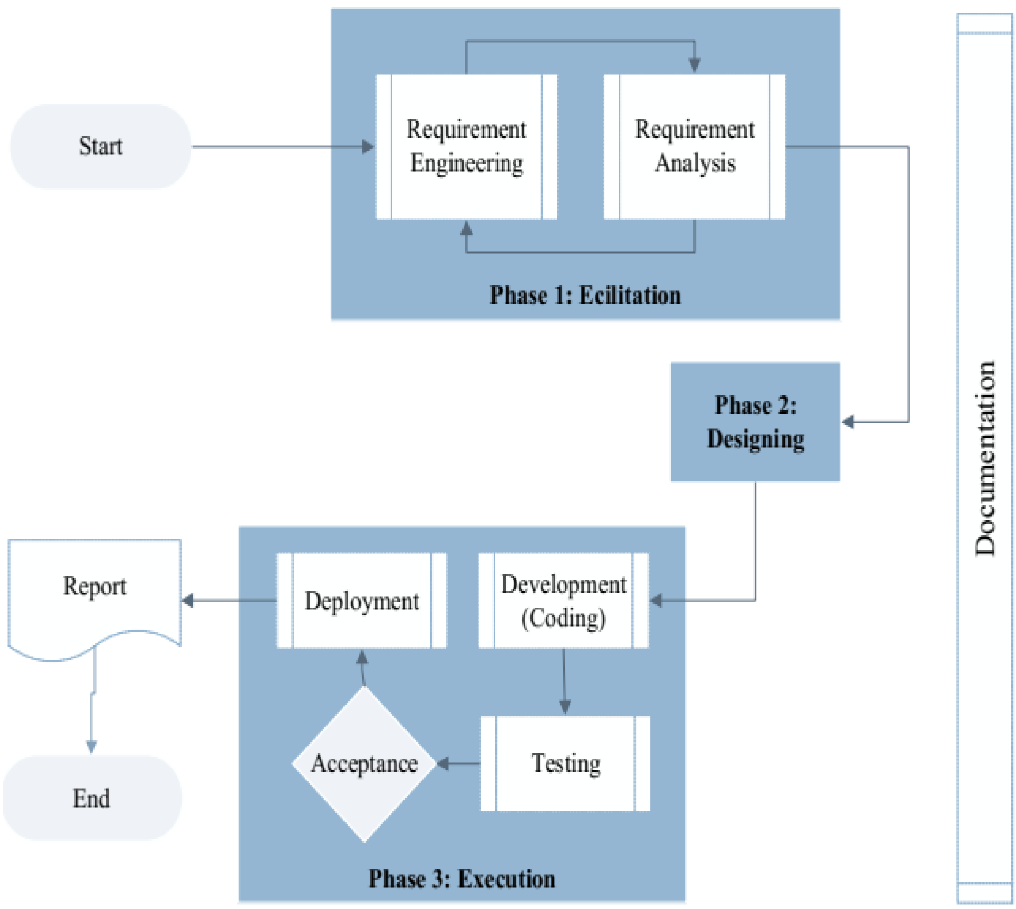


FIGURE 9. Software architecture

The process has three big phases: Elicitation to extract information about the problem and summary all requirements; Designing to prepare all the basic “ingredients” and an overview of how the application should look like; Execution to develop the application, test and deploy it.

4.1 Phase 1- Elicitation

The first phase of the process: Studying includes requirements engineering and analysis. The activities during this phase are also called requirements elicitation where author/ developer has various conversation with stakeholders to study the nature of the environment and requirements for the outputs of the project.

There are various methods could be used to extract information from stakeholders. In this case, the author used in-depth oriented interview to extract the necessary information from stake holders. Three focused meetings with 2 packaging specialists , 1 former logistics associate who is currently product manager of brake pads and hydraulic parts were conducted in the range of five months from October 2018 to discuss the topic. Four other interviews were conducted with one brake pads specialist, two wiper blades specialists, one battery specialist and two members of CLP team. In addition, the involvement in two different projects and daily tasks in System P and SAP allow the author to have full access to the pool of packaging materials data and BOM of different Part Numbers. The results of all the interviews and studies were summarized in table 4 and 5.

TABLE 5. Functional and non-functional requirements

Functional requirements	**Non-functional requirements**
• Input dimensions and weights of boxes and bin in Metric Unit • Input maximum: 3 types of boxes • Maximizing the occupied space • The total weight of all boxes does not excess weight constraint of the bin • Generating report	• Ability to extend to 5 and more boxes in the future • Help button and documentation for end-users

Table 5 is a summary of functional and non-functional requirements of application. In the non-functional requirements, end-users expect to increase the number of boxes to 5 and more in the future. However, because of the NP-Hard nature of the algorithm, when the number box increase, the required amount of time needed for calculation also increase. Additionally, having more than 5 types of boxes while packing at the warehouse is not necessarily an optimal practice of operation. Therefore, this requirement is left for further discussion.

TABLE 6. User requirements and system requirements

User requirements	System requirements
• User Interface in natural language (English) • Report generated in natural language (English) • Measurements in Metric System.	• User Interface in the form of button (no command-line) • Input slots for boxes and bin • All boxes are fully placed within the bin • No boxes overlapping each other • All boxes are rotatable • Maximizing the occupied space of bin • The total weight of all boxes does not excess weight constraint of the bin

Table 6 is a summary of all users and system requirements for the application. Among the products range of the company, battery is a special product line with acidic contents that make it non-rotatable. Therefore, this products line is excluded from the use of packing optimization software at the time of development. The system automatically assume that all boxes are rotatable.

4.2 Phase 2- Designing

Phase 2- Design includes of several different activities that result in different products described in the table 7. The detailed working process and outputs of each activities are explained in the following section.

TABLE 7. Design activities and its products

Design activities	**Design products**
Software architectural pattern	System structure
Software process model	Workflow process structure
Interface design	Interface specification
Component design	Component specification and or user journey
Data structure	Data structure specification
Algorithm design	Algorithm specification

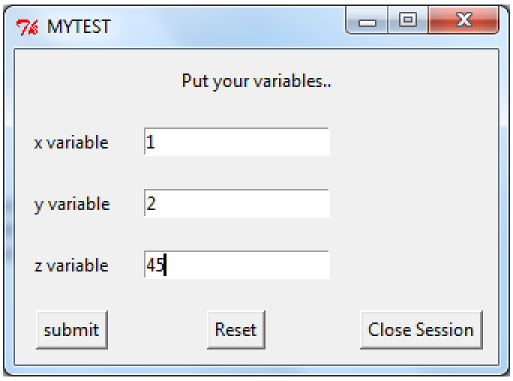
The author carried out these activities to design an application called Kombination. Activities were not carried in exactly order it appears in the table.

4.2.1 Software architecture pattern

The software could be built with command line interface. It requires end-users to type the order into a command line interface as shown in picture 12. However, this interface is difficult for end-users to communicate with the soft-ware if they do not know the basic rules and syntax of the coding language. As mentioned earlier, end-users are packing workers with limited proficiency in English and hardly know any coding language. From table 5 and 6, it is also a mandatory requirement that the users can communicate with the software easily via natural language, in this case, is simple English or Japanese.

PICTURE 12. A command line interface

The author decided to build the software with Observer Pattern using Event-Driven GUI. Observer Pattern is a pattern that one object called subject maintains a list of its dependent variables called observers that changes of the subject will lead to the changes of observers. For example, in the combination of three types of box that need to be put in the bin. The box with the largest surface area is also chosen as the first item to be put in the bin (as use of Large Area First Fit Algorithm). This largest type of box is the subject and the other three types are observers. GUI is Graphic User Interfaces which contains different icons or pictures that allow end-users to communicate with the machine without typing in command lines.



PICTURE 13. Basic GUI with TKinter written in Python

From picture 12 and 13, we can see that GUI is more user-friendly for end-users at almost any level of computer literacy. All of the labels, buttons and inputs are customizable. The design of GUI is event-driven with different command buttons that allow user to call in and execute different functions within the program.

4.2.2 Software process model

The software process model used for this case is the waterfall model which works in the way that each step must be completed before moving to the next one.

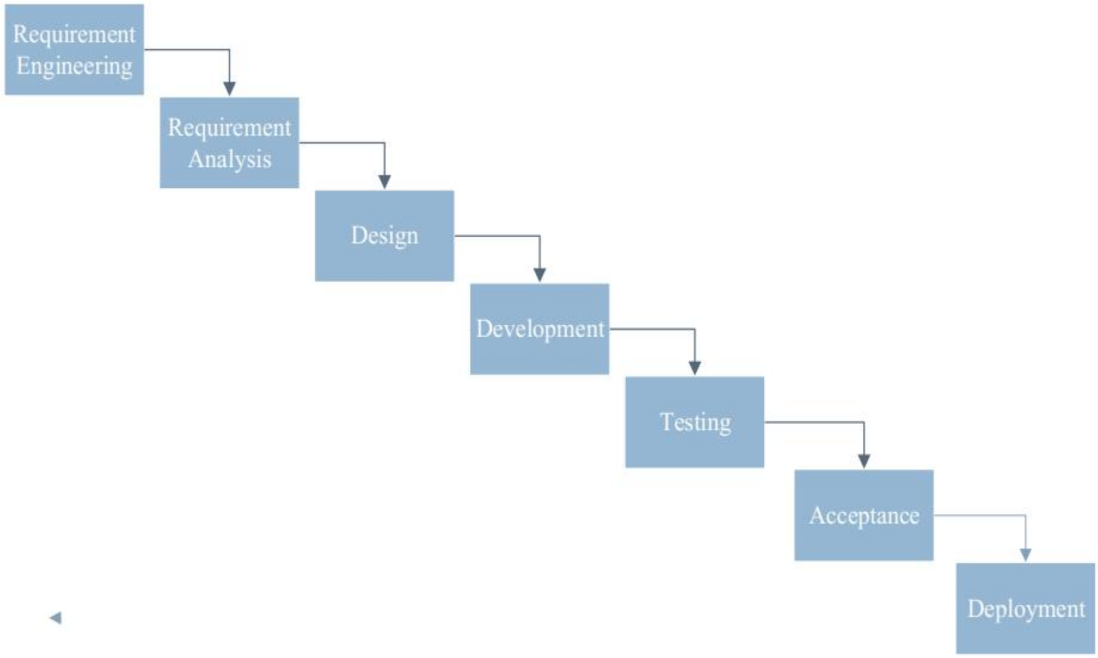
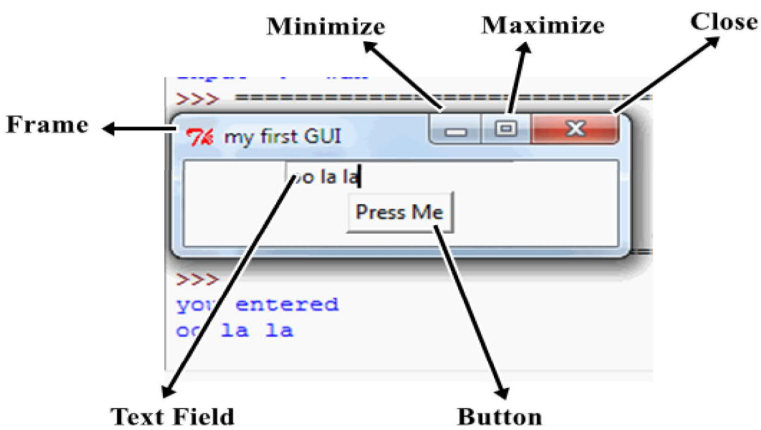


FIGURE 10. Waterfall process model

As requirement engineering, analysis and design were explained above, each step of the waterfall from development to deployment will be discussed in detail in the following sections

4.2.3 Interface design

The interface of the software consists of basic window, dialog box and button to execute command.



PCTURE 14. Example of interface

The GUI helps to create the window as can be seen in picture 14. Each window may contain one or several frames. In the example is window with only one frame. There are different components of the interface packed into this frame. The input of text field is designed so that user can enter the data. Later in the application, the input from text field will be converted into variable for calculation. After that, users can have choices of three buttons: execute, cancel and help to trigger the next event in the working process. Naturally, each window also contains minimize, maximize and close buttons.

4.2.4 Component design

The software contains of 4 different components.
The first component is the data entry component where user can enter information of the bin and boxes

TABLE 8. Data entry table

Bin	width	Length	height	weight constraint
Box 1	width	Length	height	weight
Box 2	width	Length	height	weight
Box 3	width	Length	height	weight

The data then enters second component which is a loop that running to find all combination of boxes that would fit into the bin. The third component displays the results of second component and user have choices to generate report or terminate the session. The fourth component generates report in form of Excel file according to the command from user. From this report, the user can then choose the most suitable combination according to each consignment. For example, the most optimal combination shows 20 box 1, 10 box 2, and 15 box 3, however, the consignment requires 22 box 1, then the user might choose combination that pack minimum 22 box 1.

The detailed user journey is demonstrated in figure 11. The journey is intuitive and simple to use through four steps: start the program, input data, click calculate and then export the result.

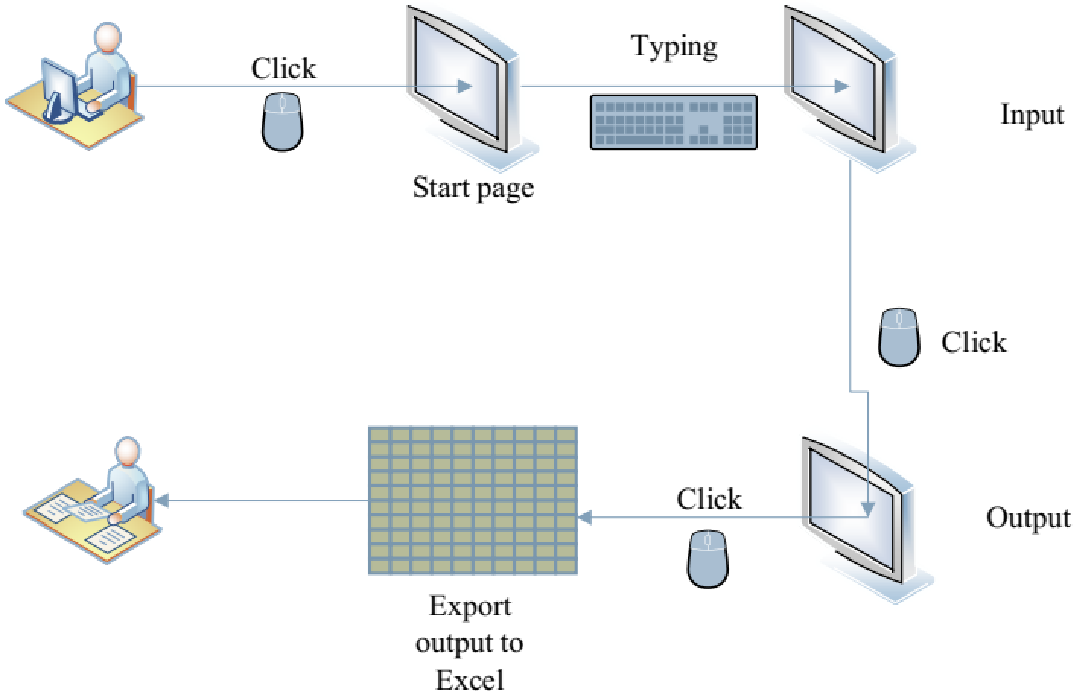
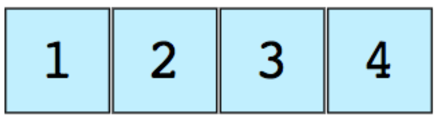


FIGURE 11. Visualization of user journey

This journey suggests that the interface of the software should contain three pages: input, calculation and result. This is an ideal number of pages since the best practice of User Experience shows that users are more likely to quit using the software if they must go through more than five steps to get to the final outputs (Pierre 2018).

4.2.5 Data structure

The input data is stored in the most popular and simple structure- array.



PICTURE 15. Data is stored in array structure

For example, when users enter the dimensions of 4 different types of boxes. The data set will be stored in an array from 1 to 4. Each slot also contains a sub-array for width, length, height and weight of each box type.

4.2.6 Algorithm design

The algorithm is built based on the Large Area First Fit algorithm and must fulfil the following constraints of the optimization:

1. All boxes are fully placed within the bin
2. No boxes overlapping each other
3. All boxes are rotatable
4. Minimizing the wasted space of bin
5. The total weight of all boxes does not excess weight constraint of the bin

Given 3 boxes with following dimensions:

TABLE 9. Naming of variables

Box name	Width	Length	Height	Weight	Quantity
Box 1	a1	b1	c1	w1	q1
Box 2	a2	b2	c2	w2	q2
Box 3	a3	b3	c3	w3	q3

The bin has dimensions of a4, b4, c4 and weight constraint of w4. The algorithm contains four steps:

Step 1: Calculating the capacity of boxes and bin:

$$v = a * b * c$$

The capacity of Box 1, Box 2, Box 3 and Bin are respectively *v1, v2, v3, v4*

Step 2: Calculating the surface areas of each boxes and sort them in descending order using if condition.

Step 3: Calculating the lower bound (*Lb*) and upper bound (*Ub*) of the bin. Lb and Ub are respective the minimum and maximum quantity of box with largest surface area that can be packed into the bin. Assuming that, box 1 is the one with largest surface area, we have:

$$\begin{cases} Lb = 1 \\ Ub = v4/v1 \end{cases}$$

with Lb, Ub ∈ N

Step 4: Calculate the number of *q1*, *q2* and *q3* with $Lb \leq q1 \leq Ub$ that fulfil the above 5 constraints. It is a multivariable linear system that fulfil the 5 constraints mentioned above.

Assuming that, box 1 is already sorted as the box with largest surface area and all independent variables are positive natural integers.

With $\begin{cases} q1 \in \mathrm{N} \\ q1 \in [\mathrm{Lb}, \mathrm{Ub}] \end{cases}$ We have:

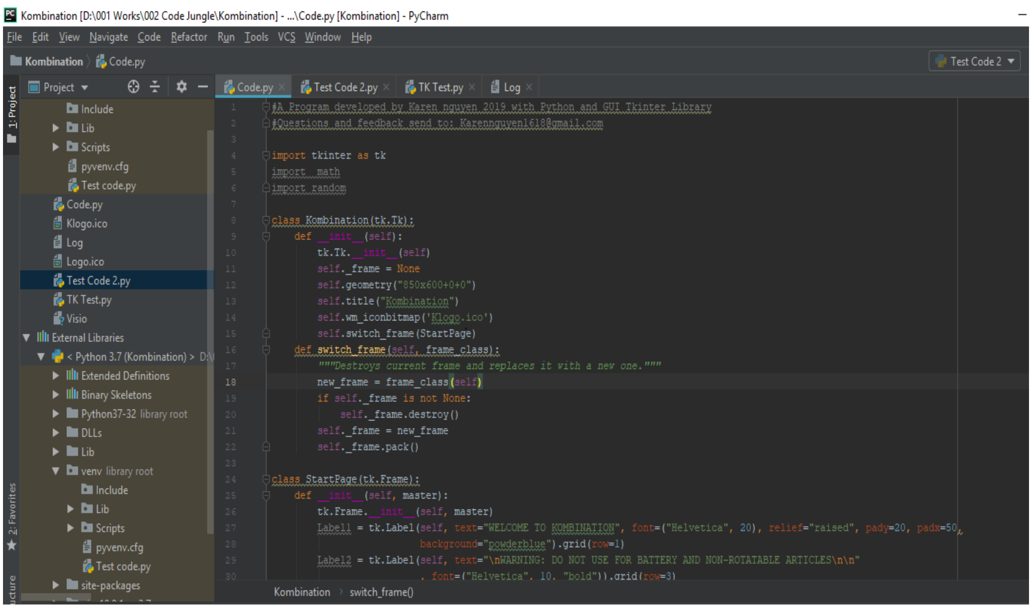
$$f(q1, q2) = \begin{cases} v1q1 + v2q2 + v3q3 \leq v4 \\ w1q1 + w2\,q2 + w3q3 \leq q4 \end{cases}$$

By not denoting the exact grid coordinate of each box in the bin, it allows boxes to rotate inside the bin. All weight and volume constraint are met and boxes must be placed inside the bin. The packing sequence and pattern are later handled in the Excel CLP Solver provided by University of Bath (Erdogan 2017).

4.3 Phase 3- Execution

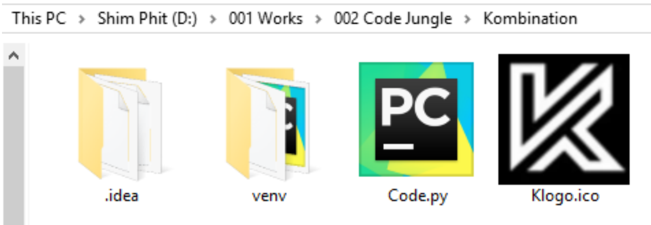
4.3.1 Developing Kombination

The software is developed with Python language using a GUI from Tkinter. Python is an object-oriented programming language and Tkinter is an open source GUI design for Python language with a build-in library that allows user to quickly call out any objects (usingpython 2018). The interface created with Tkinter has a simple design, clean and straight-forward looks.



PICTURE 15. Working screen of Pycharm

The application is called Kombination and a logo of letter K was designed by the author to replace the default Tk logo of Tkinter. To change the logo of the default window, a logo.ico file was created and placed into the same location with the code file (Critchlow 1997) as shown in picture 16. In fact, all the external elements that need to be included in the interface of the application must be stored in the same location as the .py file, so that the program can read it and follow the command.



PICTURE 16: The Kombination logo is stored in the same folder as code file.

The logic behind building an application with three pages is that we create a permanent window with one frame in it. Every time the end-user clicks a button to call a new function, it will destroy the current frame and create a new one (Vascellaro 2018). For example, this first class of the code create a permanent window that will close only when user click the close symbol (x) on the top right corner of the window. For example:

```
import tkinter as tk
import math

class Kombination(tk.Tk):
  def __init__(self):
    tk.Tk.__init__(self)
    self._frame = None
    self.geometry("850x600+0+0")
    self.title("Kombination")
    self.wm_iconbitmap('Klogo.ico')
    self.switch_frame(StartPage)
  def switch_frame(self, frame_class):
    #Destroy current frame and build a new one
    new_frame = frame_class(self)
    if self._frame is not None:
      self._frame.destroy()
    self._frame = new_frame
    self._frame.pack()
```

PICTURE 17: Example of the syntax and building basic window with frame with Tkinter

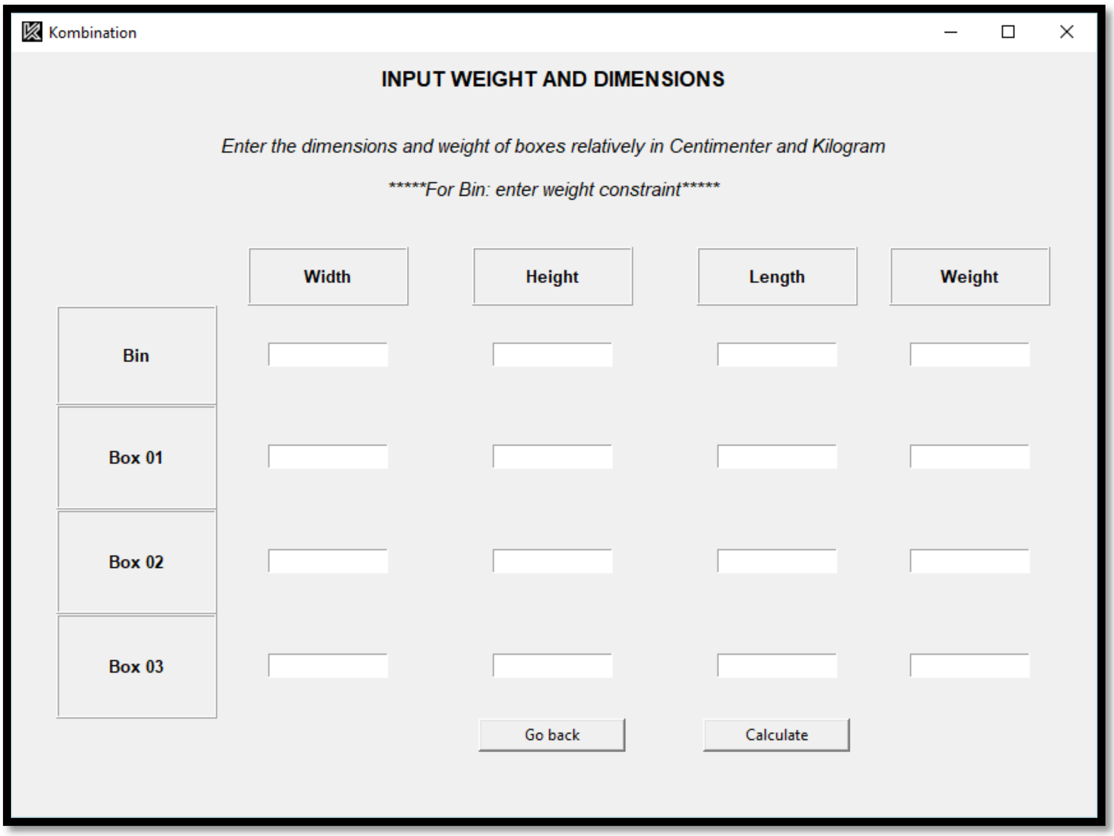
This piece of code imports two different Python libraries that are tkinter and math. Tkinter is the GUI library; and math library supports complicated mathematic calculation. The window is built with a frame, resolution 850x600 packed inside. The title is changed into Kombination and the original logo of Tk is replaced with Klogo. A function called switch frame is created so that when it is called, it will destroy the current frame and build a new one. Since the program is highly customizable, all English text can be replaced with its Japanese equivalent to fit the demand of end-users.

4.3.2 Output, Testing and Acceptance

The output of this study is a program called Kombination with the interface shown in picture 18, 19 and 20.

PICTURE 18. The start page of Kombination

The start page gives users an overview of the application, with simple welcome, a short introduction about its algorithm and a warning for not using with any battery products. When the users hit “continue”, the start page is killed and input page is created as shown in picture 19.



PICTURE 19. The input page of Kombination

PICTURE 15. Final page of Kombination

The layout of the input page is designed based on previous study. It is very simple and intuitive with input box, a “back” button arranged to the left hand-side and “calculate” button to the right hand-side of the frame. This arrangement is similar with a lot of popular software. This sense of similarity helps to put end-user at ease when being introduced to a new tool. After hitting the “calculate” button, the result is calculated and saved to the local computer, the users are navigated to final page where they can choose to terminate the section or start a new transaction. New transaction button navigates back to input page and start a new calculation.

Two instances of level 1 boxes into level 2 boxes were tested, proven to be effective. Then the software was used for small consignment of exporting samples

The department accepted the software along with another VBA Excel file to visualize the packing pattern of the combination calculated by Kombination. Kombination along with the Excel file is being used to recalculate the quantity of packaging materials used for each consignment at the end of each month for controlling purpose. Due to the complexity of the operation and the organization, the software cannot yet be used to directly intervene in the daily operation at industrial scale.

5 DISCUSSION

This section discusses the application of the software, limitations as well as feedback from both student and the company.

5.1 Overall outcomes

This study was conducted during the time of several different projects. The overlapping projects gave the authors a lot of opportunities to visit the plants, warehouses and work with specialists from Germany, Singapore and China. In particular, the author made these following trips to different location in the window of 10 months.

TABLE 10. A list of all the trips to warehouse and plants

Time	**Location**	**Participants**	**Content**
July 2018	Provider warehouse	Author (as intern) Section Manager	Confidential Warehouse tour
October 2018	A*** Plant	Author (as intern) Packaging Specialist Section Manager Packaging Manager	Confidential Plant tour Process Evaluation
October 2018	B***** Plant	Author (as intern) JP Packaging Team Sales Team	Information exchange Plant tour
March 2019	Provider Warehouse	Author (as intern) Packaging Specialists Section Manager	Confidential Warehouse Tour

The two final outputs of the study are the application called Kombination and the introduction of CLP Solver Worksheet. After introduced to the Market Business Department, the two applications were introduced to the Customer Logistics and Planning team- CLP.

This CLP team is in charge of logistics and planning activities. One specialist from the team closely monitors the packing quantity at the warehouse in Provider. The specialist visits the warehouse often to control the business process, the quantity of packaging materials used, costing and invoicing to AA-JP. The initial purpose of Kombination was to intervene before palletization, however due to the complexity of the operation, it is possibly used as a tool for cost controlling.

The research question was answered with an application that can be used by any end-user who possessed basic English skills and computer literacy. However, it could not fully support non-rotatable products such as battery and does not show any Help button. Further evaluation can be found in 5.2.

5.2 Evaluation

In general, the research was done based on previous studies in the field of operational optimization and engineering with a touch of business administration focuses on supply chain management. The author discovered the problem through daily works and approached the problem by reading literatures, first got to know about Bin Packing Problem, then Knapsack problem and decided to go along with her own work based on those foundations.

One noticeable difference is that, while previous studies mostly focused on finding the least amount of bin/ container to allocate a pre-fixed amount of different shape boxes. The author attempted to find the optimal quantity of different shape boxes to be put in a fixed dimension bin. This different certainly raised some manageable challenges during the research and was solved with further readings and some consultations from specialists.

5.2.1 Positive points

For the company, developing in-house software that tailored for specific needs of the operation is a new approach. The application is possibly used as an additional tool for controlling. A small improvement in the process of mass-production could result in significant reduction of production cost and even increase customer satisfaction with concrete and firm packaging.

On the other hand, during the process of this research, the author made various trips to both the plants and warehouses to learn about the manufacturing process, packaging process in attempt to study the problem and optimize the process. This knowledge is valuable and can be applied to not only automotive but other manufacturing industries as well. In addition to that, the author had a good amount of discussion with both packaging, logistics specialists and product specialist for brake pads, wiper blades, battery and so on to learn about the product features and customize the application according to their needs.

With a background study in Business Administration, the author was required to study basic knowledge in Software Engineering and gain a deeper understanding of Python language. The author attended two online courses, provided for free on edX.org about mentioned above topics.

TABLE 11. Online courses taken to accommodate the research.

Course	Provided by	Platform
Data Structures and Software Design	University of Pennsylvania - PennX	edX.org
Software Engineering Essentials	Technische Universität München	edX.org

5.2.2 Limitations and recommendations

Having two overlapping projects in the warehouse at the same time might provide a lot of opportunities to visit the physical warehouse, learning about the process and hands-on experiences on the field. However, it is also one of the main obstacles in the implementation of the study. During the development of the application, it was quite challenging to find a slot of time that this small project could fit in the schedule of the two big one. Due to the hierarchy and sophistication of a big organization, it was also challenging to find the right person in charge to connect between different functional areas and deal with the

IT Security requirement of the company. Overall, the application was developed. However, due to the limited capability of the author and the complexity of the problem, the outcomes were two separated working applications which are Kombination and the CLP Solver worksheet. This can lead to complication and confusion for end-users. And there was a limited number of testing instances. This limitation leads to a lack of validation on the efficiency of application. In the future study, the author should align the project with other operational activities, increase the number of types of box that can be packed, combine Kombination with the CLP Solver Worksheet to reduce complication for end-users and if possible, upgrade the user interface of Kombination.

5.3 Company feedback

In general, the feedback is positive about application with the possibility to maximize the use of space inside container, reduce packaging damages during transportation and reduce the packaging material cost. A few recommendations for future development were provided from end-users and product specialists.

The application has only three steps of operation (three pages) with buttons command and explanation in English. The software layout is rather simple and not very attractive in term of design. However, it is intuitive, easy to understand and follow. On the start page of the application, there is sort explanation and only one button. Even end-users without English knowledge could easily understand what to do. On the second page of the application, end-users know that they need to enter the dimensions of boxes. And the two buttons of “go back” and “calculate” are aligned respectively to the left and right side. This design is similar to other software and makes it more understandable. It is also a nice touch that the button and general instruction could be changed to Japanese if needed since the author has the source code.

The supplement of the application which is the CLP Solver Excel file from University of Bath is a good addition to the process. Kombination helps to calculate the quantity of each boxes and the CLP Solver helps to organize the packing pattern which in the end, help to compare with the cost that Provider charged for packaging. There are several points that could be improved in the future:

1. If technically possible, the CLP solver should be integrated into Kombination to reduce the complication for end-user. It is more convenient to have the visualization as one module of the same application than having it separately.
2. Both Kombination and the visualization should add an indication about loading direction. It is unclear whether if the container/bin is loaded from side or top.

Two other concerns raised from end-user that were solved by the author are:

1. Change the resolution of both application and visualization to bigger size so that the users could read the content easier.
2. Indicate which dimensions of the box are width, height, length. However, since the box are rotatable, it does not matter which dimension the user enters first.

During the process of creating the application, there was a lot of back and forth communication between the author and involved parties. Several features were changed after testing.

REFERENCES

Bein, D., Bein, W. & Venigella, S. 2011. Cloud storage and online bin packing." Intelligent Distributed Computing. Berlin. V. Springer, 63-68.

Beirão, J. 2009. Packing Problems in Industrial Environments: Application to the Expedition Problem at INDASA. Universidade Técnica de Lisboa. Instituto Superio Técnico. Doctoral thesis.

Bosch Automotive. 2018. New grads training Division Info Session- AA. Workshop. Intern Orientation Date on 28.03.2018. Tokyo.

Bosch Deutschland 2018. Die Bosch-Gruppe im Überblick. Read on 03.10.2018. https://www.bosch.de/unser-unternehmen/bosch-gruppe-weltweit/

Chen, C., Xie, J., Yang, F. & Yang, D. Packaging Solution Optimization of Automotive Parts and Its Ocean Shipping Test. International Conference on Artificial Intelligence and Industrial Engineering. 26-27.7.2015. Phuket.
Critchlow, R . 1997. Tk Command wm - Communicate with window manager. Read on 20.02.2019. http://www.tcl.tk/man/tcl8.4/TkCmd/wm.htm#M18

Dodson's Japanese Partsworld 2017. Which Car Parts Need To Be Replaced The Most Often?. Read on 02.03.2019. https://dodsonautospares.co.nz/car-parts-replacement/

Erdoğan, G. 2017. CLP Spreadsheet Solver. Released on 07.09.2017. Read on 02.03.2019. https://researchportal.bath.ac.uk/en/publications/clp-spreadsheet-solver

Egeblad, J. & Pisinger, D. 2009. Heuristic approaches for the two- and three-dimensional knapsack packing problem. Computers & Operations Research 36(4), 1026-1049.

Energizing The World With Innovation. 2018. The Global Innovation Index 2018: Energizing The World With Innovation. Cornell University, INSEAD & WIPO.

Fukunaga, A. & Korf, R. 2007. Bin Completion Algorithms for Multicontainer Packing, Knapsack, and Covering Problems. Journal of Artificial Intelligence Research 28, 393-429.

Gürbüz, M., Akyokuş, S., Emiroğlu, I. & Güran, A. 2009. An Efficient Algorithm for 3D Rectangular Box Packing. Applied Automatic Systems: Proceedings of Selected AAS

Junqueira, L. & Morabito, R. 2016. On solving three-dimensional open-dimension rectan-gular packing problems. Engineering Optimization 49(5), 733-745.

Ko, J., Abraham, R., Beckwith, L., Blackwell, A., Burnett, M., Erwig, M., Scaffidi, C., Lawrance, J., Lieberman, H., Mzers, B., Rosson, B., Rothermel, G., Shaw, M. & Wiedenbeck, S. 2011. The state of the art in end-user software engineering. ACM Computing Surveys, 43(3)

Lin, M., Tsai, J. & Chang, S. 2016. A superior linearization method for signomial discrete functions in solving three-dimensional open-dimension rectangular packing problems. Engineering Optimization, 49(5), 746-761.

Lodi, A. 2000. Algorithms for Two-Dimensional Bin Packing and Assignment Problems. Universita Degli Studi Di Bologna. Bologna. Doctoral's Thesis.

Malkevitch, J. 2004. Tidbit: Bin Packing. Released on 17.09.2014. Read on 17.02.2019. https://www.york.cuny.edu/~malk/tidbits/tidbit-bin-packing.html

Orso, A. CS 6300: Software Development Process (Software Engineering: Concepts and Practice). Read on 22.12.2018. http://www.omscs.gatech.edu/cs-6300-software-development-process

Pierre, S. 2019. Ideathon: Come with an idea, leave with a prototype. Prototyping and User Experiences Workshop. 19.01.2019. Tokyo: Le Wagon Tokyo Coding Bootcamp.

Roland Berger. 2013. Automotive Insights: Automotive Competence Center Client Magazine 2(1)

Salva, N. 2018. Why Is Japan Still So Attached to Paper?. The New York Times Style Magazine. Released on 19.11.2018. Read on 20.04.2019. https://www.nytimes.com/2018/11/19/t-magazine/japanese-paper-washi.html

Smith, H. 2001. Level Algorithms and Shelf Algorithms. Read on 03.01.2019. https://users.cs.cf.ac.uk/C.L.Mumford/heidi/Approaches.html

Terno, J., Scheithauer, G., Sommerweiß, U. & Riehme, J. 2000. An efficient approach for the multi-pallet loading problem. European Journal of Operational Research 123(2), 372-381.

treeDiM. 2014. Stack Builder. Read on 18.07.2018. https://www.treedim.com/stackbuilder/en/

Tsai, J., Wang, P. & Lin, M. 2011. An efficient deterministic optimization approach for rectangular packing problems. A Journal of Mathematical Programming and Operations Research: Optimization 62(7), 989-1002.

Tsai, J., Wang, P. & Lin, M. 2013. A global optimization approach for solving three-dimensional open dimension rectangular packing problems. A Journal of Mathematical Programming and Operations Research Optimization 64(12), 2601-2618.

Vascellaro, . 2018. Switch between two frames in tkinter. Released on 16.03.2018. Read on 20.02.2019. https://stackoverflow.com/questions/7546050/switch-between-two-frames-in-tkinter

VDA 4530. 2013. Disposable Small Load Carrier (EW-KLT) System 01. Berlin: Verband der Automobilindustrie.

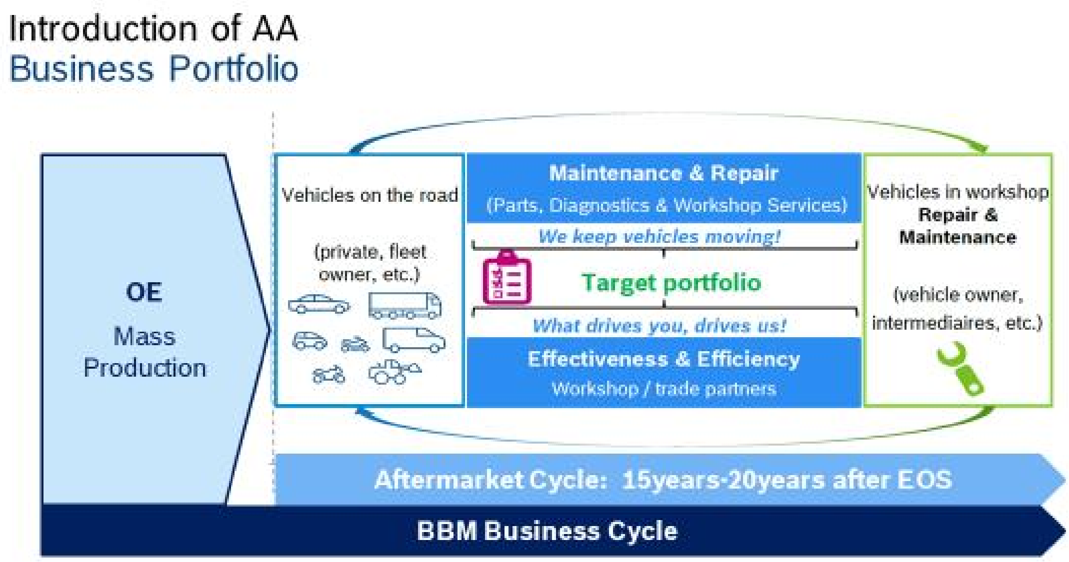
Westerlund, J., Papageorgiou, L.G. and Westerlund, T. 2005. A problem formulation for optimal mixed-sized box packing. European Symposium on Computer-Aided Process Engineering-15. Computer Aided Chemical Engineering 20, 913-918.

APPENDICES

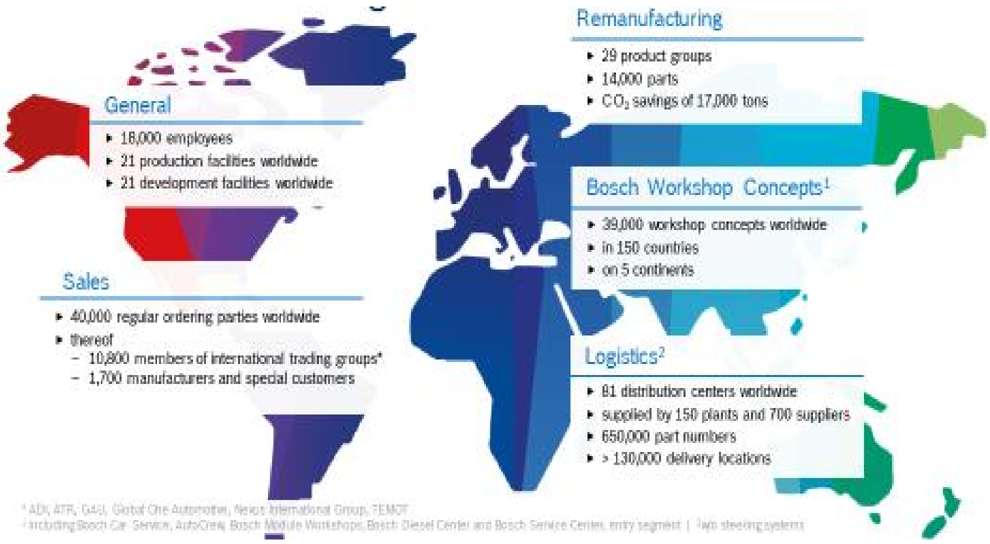
Confidential information excluded

Appendix 1. New Grads Training Division Info Session + Workshop

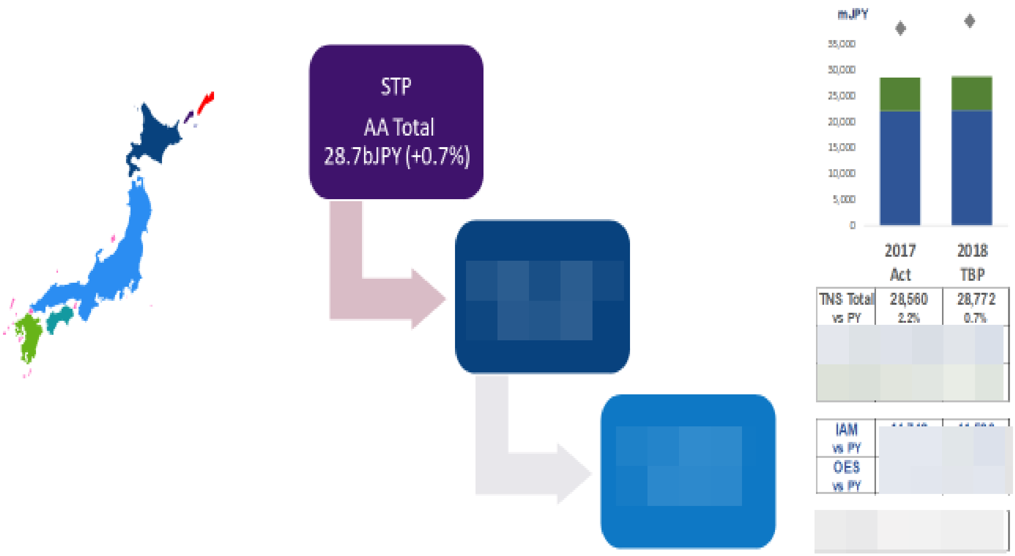
1(3)



2(3)



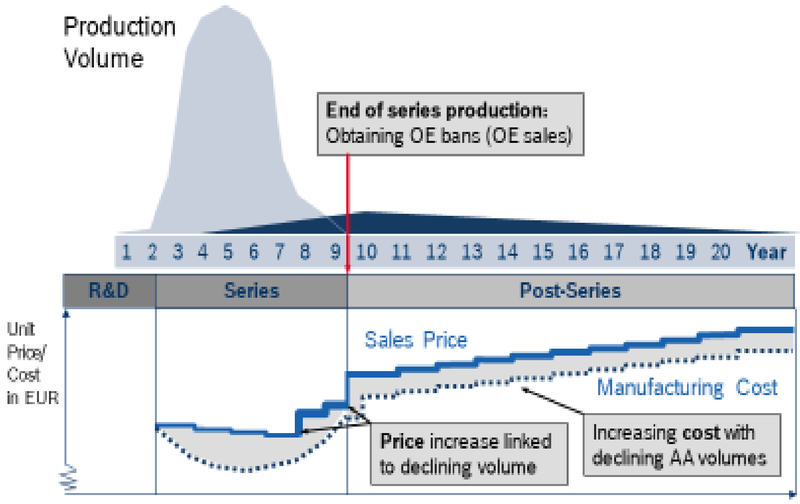
General
18,000 employees
21 production facilities worldwide
21 development facilities worldwide
Sales
40,000 regular ordering parties worldwide
thereof
– 10,800 members of international trading groups*
– 1,700 manufacturers and special customers
Remanufacturing
29 product groups
14,000 parts
CO_2 savings of 17,000 tons
Bosch Workshop Concepts[1]
39,000 workshop concepts worldwide
in 150 countries
on 5 continents
Logistics[2]
81 distribution centers worldwide
supplied by 150 plants and 700 suppliers
650,000 part numbers
> 130,000 delivery locations
* ADI, ATR, GAU, Global One Automotive, Nexus International Group, TEMOT
[1] Including Bosch Car Service, AutoCrew, Bosch Module Workshops, Bosch Diesel Center and Bosch Service Center, entry segment | [2] w/o steering systems

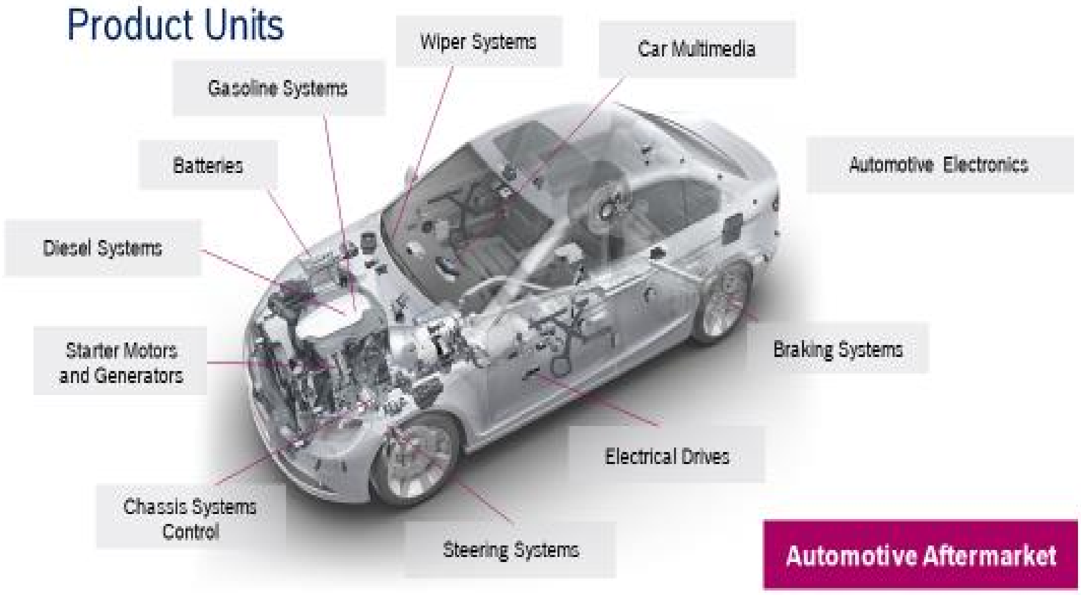


STP
AA Total
28.7bJPY (+0.7%)
mJPY
35,000
30,000
25,000
20,000
15,000
10,000
5,000
0
2017 Act
2018 TBP
TNS Total vs PY
28,560 2.2%
28,772 0.7%
IAM vs PY
OES vs PY

3(3)

Product Life Cycle Management





Appendix 2. World Pallet Size

North America Standard Pallet Dimensions

Dimensions, inches (W x L)	**Dimensions, mm(W x L)**	**Industries Using**
48 × 40	1219 × 1016	Grocery as well as many other uses
42 × 42	1067 × 1067	Telecommunications and Paint
48 × 48	1219 × 1219	Drums
40"× 48	1016 × 1219	Military and Cement
48 × 42	1219 × 1067	Chemicals and Beverage
40 × 40	1016 × 1016	Dairy
48 × 45	1219 × 1143	Automotive
44 × 44	1118 × 1118	Drums and Chemicals
36 × 36	914 × 914	Beverage
48 × 36	1219 × 914	Beverage, Shingles, and Packaged Paper
35 × 45.5	889 × 1156	Military 1/2 ISO Container
48 × 20	1219 × 508	Retail

Australian Standard Pallet Dimensions

Dimensions, inches (W × L)	**Dimensions, mm (W x L)**	**Region Most Used**
45.90 × 45.90	1165 x 1165	Australia, New Zealand


European Standard Pallet Dimensions?

Dimensions, inches (W x L)	Dimensions, mm(W x L)	EURO Pallet type
31.50 × 47.24	800 × 1,200	EUR, EUR 1
47.24 × 39.37	1,200 × 1,000	EUR 2
39.37 × 47.24	1,000 × 1,200	EUR 3
31.50 × 23.62	800 × 600	EUR 6


ISO Pallets Dimensions

The International Organization for Standardization (ISO) sanctions six pallet measurements, detailed in ISO Standard 6780: Flat pallets for intercontinental materials handling-principal dimensions and tolerances.

Dimensions, inches (W × L)	Dimensions, mm (W x L)	Region Most Used
48.00 × 40.00	1219 x 1016	North America
39.37 × 47.24	1000 x 1200	Europe, Asia
44.88 × 44.88	1140 x 1140	Australia
42.00 × 42.00	1067 x 1067	North America, Europe, Asia
43.30 × 43.30	1100 x 1100	Asia
31.50 × 47.24	800 x 600	Europe (fits many doorways)

Another Table with international pallet sizes in millimeters and inches:

Product name	**L x W x H in mm**	**L x W x H in inches**	**Major Countries Available**
Timber Block Pallet North American Standard (48 x 40 inches)	1219 x 1016 x 141	48 x 40 x 5.55	USA, Italy, New Zealand
Wooden Pallet	1200 x 1000 x 162	47.24 x 39.37 x 6.38	Most European countries, Latin America, United Kingdom, India, New Zea-land
Wooden & Metal Pallet	800 x 600 x 163	31.50 x 23.62 x 6.42	Most European countries, United Kingdom, South Af-rica
Wooden Pallet	800 x 600 x 166	31.50 x 23.62 x 6.53	France
Wooden Pallet - 3 Runner	1200 x 1200 x 154	47.24 x 47.24 x 6.06	India, China
Wooden Pallet - New Zealand standard	1200 x 1000 x 140	47.24 x 39.37 x 5.51	New Zealand
Perimeter Wooden Pallet	1200 x 1000 x 154	47.24 x 39.37 x 6.06	India, China
Timber Half Pallet	1220 x 508 x 128	47.24 x 20 x 5.04	Canada
Bearer Pallet	1200 x 1200 x 150	47.24 x 47.24 x 5.90	Namibia, South Africa
Bearer Pallet	1500 x 1200 x 176	59 x 47.24 x 6.13	Namibia, South Africa
Wooden Pallet	1200 x 800 x 144	47.24 x 31.50 x 5.67	Most European countries, United Kingdom, South Af-rica, India
Wooden Pallet - Australian Standard	1165 x 1165 x 150	45.87 x 45.87 x 5.90	Australia, New Zealand
Pallet Mercosur	1200 x 1000 x 145	47.24 x 39.37 x 5.71	USA, Argentina
Wooden Pallet	1000 x 600 x 162	39.37 x 23.62 x 6.38	France
Plastic Dolly	600 x 400 x 173	23.62 x 15.75 x 6.81	Austria, Switzerland

			4(4)
Plastic Pallet - New Zealand standard	1219 x 1016 x 144	48 x 40 x 5.67	New Zealand
Plastic Display Pallet	600 x 400 x 145	23.62 x 15.75 x 5.71	Most European countries
Plastic Display Pallet	1200 x 800 x 160	47.24 x 31.50 x 6.30	Belgium, Denmark, Finland, France, Germany, Ireland, Netherlands, Portugal, Spain, Sweden, United Kingdom
Blue Shield Plastic Pallet	1200 x 1000 x 150	47.24 x 39.37 x 5.90	China, Thailand, South Africa
Plastic Pallet - 3 Runner	1200 x 1000 x 150	47.24 x 39.37 x 5.90	China
Plastic Pallet	1165 x 1165 x 150	45.87 x 45.87 x 5.90	Australia
Automotive Pallet - US Standard	1166 x 1242 x 152	45.87 x 48.90 x 5.98	USA, Canada, Mexico
Automotive Half Pallet	100 x 600 x 144	3.93 x 23.62 x 5.67	Most European Countries, Brazil, Argentina
Automotive Pallet	1200 x 1000 x 144	47.24 x 39.37 x 5.67	Most European Countries, Brazil, Argentina