



**Optimizing transportation cost with joint ordering:
an application on a Mexican distribution company**

Arturo Mercado Guijarro

409994

Master thesis

Rotterdam, 2015

Abstract

A cyclic ordering strategy is selected to minimize transport costs for a Mexican distribution company specialized in industrial lubricants. Simulations we made first following the modified algorithm of Goyal. Several scenarios changing major and minor setup cost were compared with a method from Chopra & Goyal that agglomerates around the most ordered product. Goyal algorithm proved to generate lower average costs every time. Following this procedure other restrictions related to the company operations were introduced. First a method from Porras & Dekker is applied to include minimum order quantities. Then to include truck capacity restrictions Goyal algorithm is applied and the ordering is changed to improve the average percentage of filling from the trucks. Finally a non-cyclic strategy is used to compare the results and show the limitations from cyclic strategies.

Table of Contents

1. Problem definition.....	1
2. Background Information	3
System description and Data extraction.....	3
Market Description	5
Presentation of the products and Demands	7
3. Literature review	9
4. Data	11
5. Models	13
General Concepts and problem formulation	13
Joint replenishment model.....	14
Modified algorithm of Goyal	15
JRP with Minimum Order Quantity Restriction.....	15
Algorithm Joint Replenishment Problem with MOQ.....	17
Joint Replenishment Problem adjusting to most ordered product.....	18
Joint Replenishment Model with Truck Capacity	19
6. Calculation of economic parameters.....	23
7. Results	25
7.1 Examples to represent all models (4 products)	25
7.2 Simulation to represent company scenario (20 products)	31
7.3 Improving results with a non-cyclic policy.....	33
8. Conclusions, discussion and limitations.....	35
Reference list	36
Appendix	37

1. Problem definition

A Mexican distribution company Merdiz needs to optimize its distribution policy. The company has a portfolio specialized in industrial and transport lubricants. The basic parameters that characterize the market and operations are:

- Portfolio is composed of 83 products with roughly 950 pallets ordered per year.
- There is a minimum ordering quantity of 1 pallet per product.
- There is a truck capacity of 24 pallets.
- Products are sold in 6 types of presentations: Drums, buckets, IBC, bottles, kegs and boxes.

A model is needed that will take the parameters of interest from the operation and can deliver an optimal strategy that could be used as a guideline to be updated into the company system. This guideline will be in terms of scheduling of trucks and product ordering.

Several scenarios will be tried to understand the effects of changes in the economic parameters and the limitations in including a cyclic strategy for the company.

Relevance to the subject and motivation

The Joint replenishment problem is a complex theoretical mathematical problem with a direct application to logistic companies. The company becomes more difficult as we include other restrictions like truck capacities and minimum order quantities. The application leads to immediate cost savings if the economics parameter are measured properly and the models are applied carefully.

Now the company has a system which has been updated and followed for the last two years for the purposes of financial reports and taxes. However there is not a guideline for improving truck efficiency or holding cost, basically the company follows an individual approach for the products and the trucks are filled as much as possible. Having a guideline that follows the status for each product with a joint approach would lead to a strategy that will minimize transport cost and will increase help to convert the system as a tool to enhance productivity in the logistics of the company too.

Models to approach the problem

Two papers were found to compare the basic joint replenishment problem. Later another paper is used to include the minimum order quantity and finally an approach is proposed to take care of the truck capacity problem.

The classic approach for the Joint Replenishment Problem using a cyclic policy is found on Goyal (1974), which published an iteration that later was modified by van Eijs (1993). The method uses an algorithm in which two bounds are found for the optimal ordering schedule. From there different combinations for the individual product ordering are tried until an optimum is reached.

Another approach is tested to compare it with the Goyal algorithm. This method is obtained from Chopra & Meindl (2012). It is a simpler method that could be compared to the basic JRP. It uses the same formula to obtain the optimal waiting time between orders. However it does not follow an exhaustive algorithm that tries all combinations of product ordering until finding an optimum. What it does is aggregates orders for smaller demands around the most ordered product to reduce costs.

Those last methods were compared and the modified algorithm from Goyal find a better solution each time. However the savings and distribution saving are very small, most of them less than 1% and the reason

is both method use the same formula for the waiting time between orders. However the comparison helps to show the relation between the parameters used and introduce the problem to extend it later with more complex restrictions.

We introduce a method from Porras & Dekker (2008) that uses MOQ, this approach uses the algorithm from Goyal but includes a restriction for the product ordering and analyses the jumps in the total cost generated by this restriction. The limitation with this model is that we have to relax the results to include mixed pallets, given that even if there is a minimum ordering, once the order surpasses the restriction the results can give orders that are not integers or multiples of the minimum ordering. That is a more complex restriction that we will not analyze in this report but will be interesting to have for further research.

Finally an approach that includes the truck capacity is proposed. This method uses the definition of average truck to correct the total cost (Oudenes, 2015). Using the Goyal algorithm again and assuming the possibility of mixed pallets the individual optimal product schedule is fixed and the time between orders is variated. The objective is maximizing the average filling percentage of the trucks moving the size of the orders. The method delivered savings in total costs compared to Goyal algorithm for the scenarios tested.

Evaluating measures and economic parameters

The JRP analyzed for this report uses a cyclic policy that assumes an equal ordering for each product each time. The variables are in terms of how much will be ordered and when it will be ordered. Once selected the optimal strategy we expect to have an ordering schedule will have a cyclic pattern. That cyclic pattern will be used to measure the total cost based in the next three definitions.

- Holding Cost: This cost is linear depending on the average of orders, this definition depends on the assumption of a constant known demand.
- Major Setup Costs: This is the number of times a truck is required. However it involves as well administrative cost like setting up the orders together and updating the data on the system.
- Minor Setup Costs: This is the cost for getting a certain product in the truck. It does not depend on how much is loaded on the truck but if the product is loaded or not. Each product would have a different minor cost depending on their size. It will be measured as the labor cost involved with the loading/unloading of the product when the truck arrives.

For testing all the models and test the relations between the variables and the parameters we use an approach with 4 products from the company. All the products have a certain type of presentation as we said before, so we took the total demand for IBC, buckets, drums and the rest of the products. The parameters are obtained through weighted averages, so that they recommend

Once we test all models we wanted to use an approach that is a closer guideline to the company scenario, so we used the 19 most demanded products and a 20th product that will aggregate the rest of the product demand. That is made to include orders that are greater that one pallet which is the company operation restriction, and allow to find out a better redistribution on the trucks. We find that the method proposed optimizing the truck filling leads to interesting savings compared with Goyal algorithm.

Finally the results are compared with a non-cyclical policy. What this method does is distributing the trucks in several orders, instead of having several trucks ordered at a time. What was found is that this method gives and improvement over Goyal algorithm, but not over the Goyal algorithm with the filling percentage of trucks optimized. This application gives an idea of cyclic strategies limitations and non-cyclic strategies complexity.

2. Background Information

System description and Data extraction.

The company Merdiz has a website that updates information over level of stock, sells and orders. The system however was started in 2013 and that is why there is not a possibility of extracting older information. Once log-in was allowed the data had to be accessed product by product. The information came to the level of transaction and it had to be sorted to be useful. The basic filters that had to be changed where:

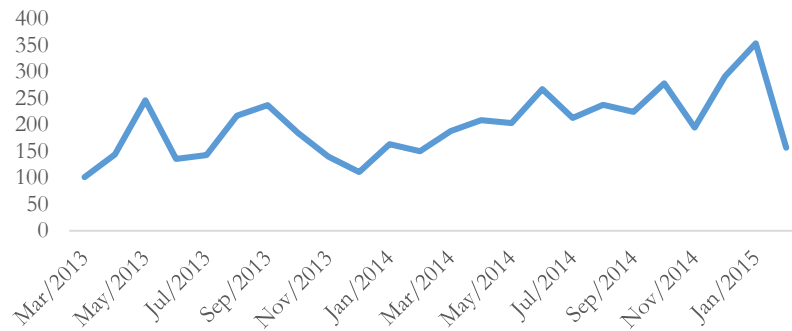
- Type of operation: the transaction could be sell, an order, or an update in stock created by cancelation, lost merchandise or theft. The information needed from the products was the demand, then only the sells where selected.
- Category of product: The transactions came in individual packages, so the data had to be setup in pallets according to the type of product selected.
- Date of operation: On this case we just use the total demand during the 2 years of analysis however the date will help us to show how the demand behaves in the next figures 2 and 3.

Figure 1. Update from Company System

Fecha	Docto.	Tipo de Mov.	Entrada	Salida	Cant. por Costear Promedio	Existencia	Costo Unitario
18/03/15	TM-1603007	Traspaso	0	-1	0.0	66	\$610.10
31/03/15	A-02137	Factura	0	-1	0.0	67	\$621.28
21/01/15	A-02107	Factura	0	-2	0.0	68	\$637.64
09/01/15	A-02076	Factura	0	-1	0.0	70	\$646.17
17/12/14	A-02018	Factura	0	-1	0.0	71	\$651.28
03/12/14	TM-1412002	Traspaso	0	-1	0.0	24	\$619.01
03/12/14	A-01993	Factura	0	-1	0.0	25	\$627.49
02/12/14	A-01988	Factura	0	-1	0.0	26	\$627.49
01/12/14	PM-6763	Entrada	48	0	0.0	72	\$510.10
19/11/14	A-01950	Factura	0	-1	0.0	27	\$627.49
13/11/14	A-01935	Factura	0	-1	0.0	28	\$673.49
13/11/14	A-01933	Factura	0	-3	0.0	32	\$573.49
13/11/14	A-01934	Factura	0	-3	0.0	29	\$573.49
13/11/14	A-01932	Factura	0	-3	0.0	35	\$573.49
03/11/14	A-01890	Factura	0	-1	0.0	38	\$646.17
23/10/14	A-01856	Factura	0	-1	0.0	39	\$573.49
23/10/14	A-01854	Factura	0	-3	0.0	43	\$573.49
23/10/14	A-01855	Factura	0	-3	0.0	40	\$573.49
23/10/14	A-01853	Factura	0	-3	0.0	46	\$573.49
23/10/14	PR2559	Entrada	48	0	0.0	49	\$510.10
07/10/14	A-01774	Factura	0	-4	0.0	1	\$551.13
30/05/15	A-21761	Entrada	6	-1	0.0	6	\$651.98

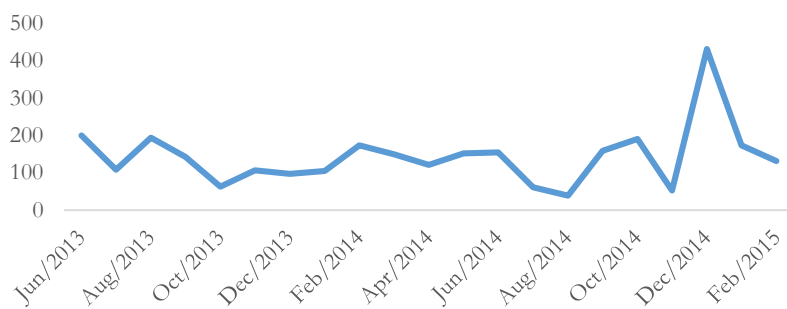
On figure 2 we can observe that there are several peaks on the demand, and there is not a cyclical behavior leaning towards specific months. Talking to the managers of the company they commented that the demand is depending on a few big clients and the competence in the branch is creating big uncertainty because there are no long term contracts. That is not only bad news for the company, but also for the research because it makes forecasting of the demand more difficult. On this study we are taking a total constant known demand and that generates input for the models. However much still can be done in terms of forecasting for application on this company on further research.

Figure 2. Demand from Merdiz 2013-2015 (€1000)



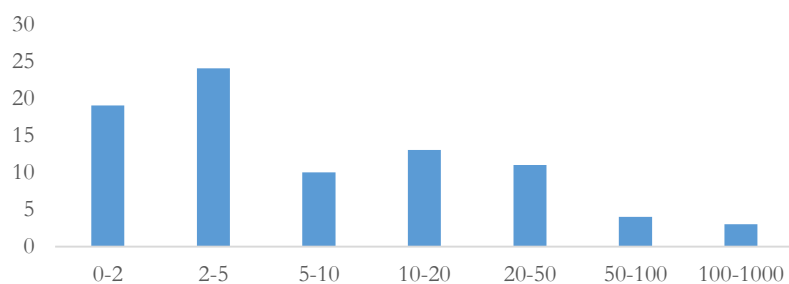
On the other side the ordering is much more flat, we can see a peak on the beginning of this year because there was a big deal with a new client. However the company have only limited resources every month and it has to select according to the risk of an uncertain demand.

Figure 3. Ordering from Merdiz (1000€)



When the demand of the product is compiled, filtered and converted into pallets. It can be seen that the demand is centered in a few products with high demand and for the rest there is a very irregular demand. A monthly demand and standard deviation is measured to get the coefficient of variance $cv = \sigma/\mu$. The graph shows that there is big uncertainty for most of the products involved.

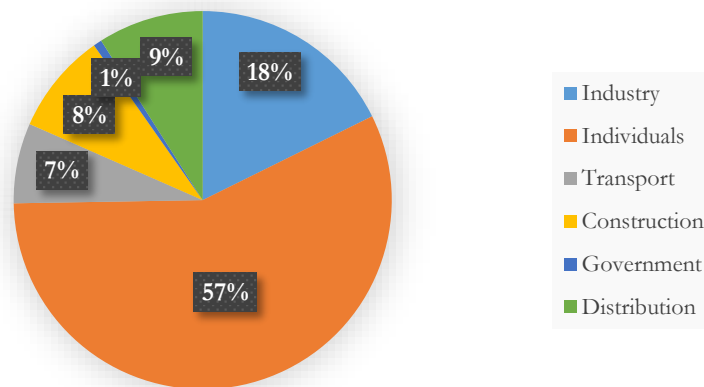
Figure 4. Number of Products by Coefficient of Variance



Market Description

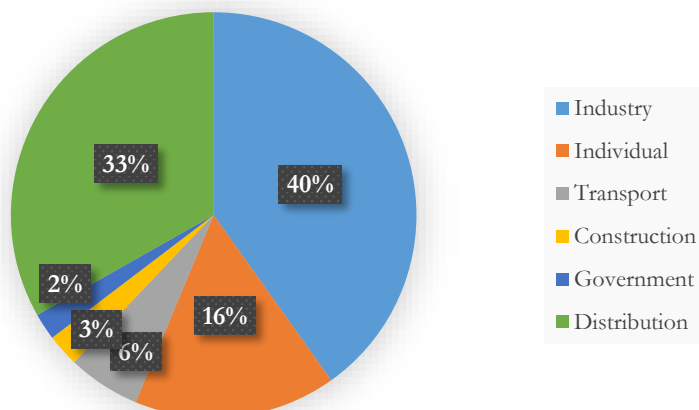
We can understand why the demand of the company has such variance by looking at the customers. Most of the market is based on industrial manufacturers that require lubricants for their processes, there are also transport or distribution companies, which have a fleet of trucks that need to be maintained. Construction and government companies need special lubricants for construction machinery. The rest of the companies are dedicated to distribution to smaller companies or individual that do not get a discount on volume. For the next graph we see that there are a lot of sells done by individuals. However this is only the number of customers, but the percentages change when we look at the selling.

Figure 5. Distribution of type of customers



When we take a closer look to the sells number we can see that even if all the costumers are individuals, the market is own by the distribution and industry companies that sell the lubricants to smaller costumers or to industries with enough scale to be direct consumers. All the other markets look small in comparison and they are used to diminish risk because close to 50% of all sells are divided in approximately 10 companies.

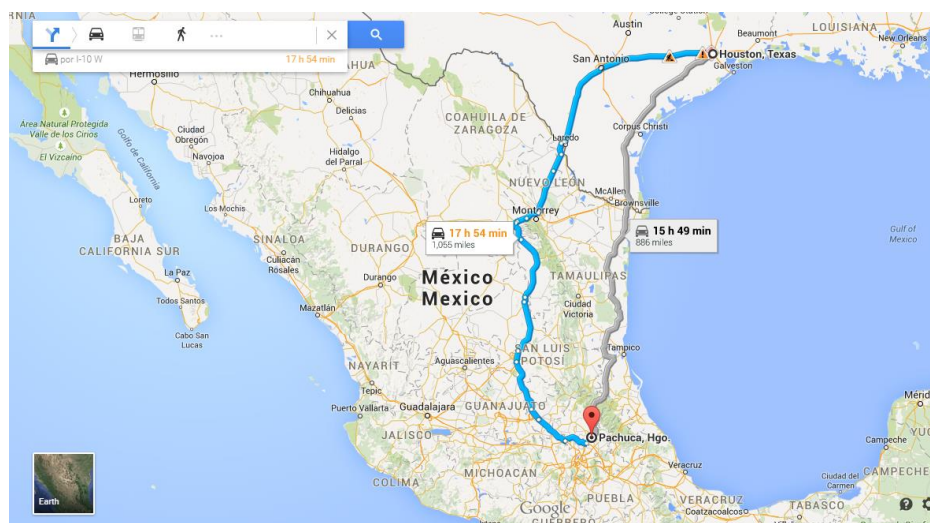
Figure 6. Percentage of sells by type of customers



Another interesting fact from the company is how they are being supplied. They have a contract with Shell and the supply is done from Houston, Texas while the warehouses are located in the center of Mexico. There is a lag in the ordering period of 5 to 7 days. There are two ways to pass the border, one is on the side of Laredo and the other on Brownsville, depending on how busy the customs agents are on each side. A truck would take around 2-3 days to do the trip. However the load always stays between 2 to 4 days on customs depending on how busy the season is or if there was a problem with the regulation in terms of import taxes or loading and packaging.

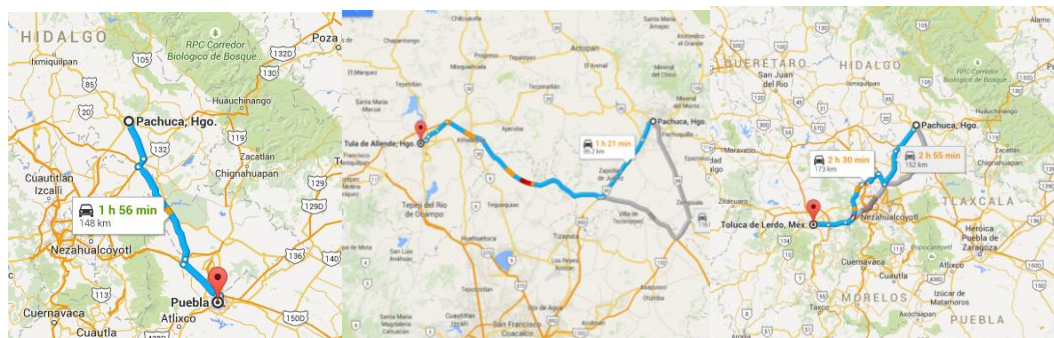
In the border the trucks need to be switched. In United States the regulation for supplying this type of products only allows to have 24 pallets in the truck. On Mexico that regulation is more lax but the import is done from Houston so the truck capacity is 24 and later will be used as input for the models.

Figure 7. Route from the Supplier to the Warehouse



Once the service is done in Pachuca, the data is uploaded on the system and the product could stay in that city for direct service with costumers or it can be sent to other warehouses in other states in Mexico. So far those locations are in Puebla, Tula and Toluca. These warehouses are shown on the next figure to give an idea of the extension of the market. Shell makes arrangements with the distributors to obtain the distribution of a certain territory. The company cannot distribute to Mexico City because that area is assigned to a different distributor. However deals between distributors can be made to increase market recognition and joint sells.

Figure 8. Routes from the central warehouse to the selling points



Presentation of the products and Demands

As we already established there is a minimum order quantity of 1 pallet for all the products. However the products come in different presentations that vary in weight, maneuverability and volume. All of them are unloaded with a forklift but once they are requested from a customer they are unpacked and wrapped together. That means different operation times and manpower that involves different labor cost. That information will be used to define the minor ordering cost, needed as input for the simulations.

Figure 9. Presentation of Products.

Packaging available



Pails 19 liters



Drums 208 liters



IBC 1040 liters



Bottles 1 liter



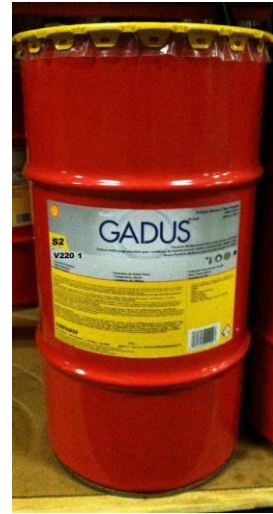
Bottles 4 liters



Box (Depending on Product)



Keg 50 liters



3. Literature review

Khouja and Goyal (2008) made a summary of this literature on the last 20 years on Joint replenishment problem. It is especially helpful as guideline for this review, because the paper applied from this report is also from Goyal.

The joint replenishment problem is generally and objective function to minimize costs while satisfying demand. This function is composed by two parts:

- Setup or ordering costs: For this case we divide this cost into major setup cost and minor setup cost. The first being the fixed cost for ordering a truck service and the second the individual cost of setting up a product inside the truck.
- Holding costs: Involves the opportunity cost of holding the product on inventory while it is sold. For this case this cost is linear.

A first thing that we have to specify when we calculate with a JRP problem is what kind of inventory policy is followed. A periodic review in which we assign a vector \mathbf{k} that specifies turns for a constant ordering. Or a continuous review that is called a can policy. For this policy there is a minimum level product replenishment and a can level. So that if a minimum level is reached, all the products with levels between the minimum and can level are ordered up to level, so we can have different orders every period (Johansen and Melchior, 2003). For this case the company uses a periodic review and the papers applied use the same policy.

There are first two ways of dealing with a joint distribution problem in terms of the objective function:

Indirect group strategy (IGS): The most straightforward solution to this problem is to use of an EOQ for every product. That would lead to take the transport cost as individual instead of shared. It is obvious that this strategy is non-optimal and will not be followed on this report.

Direct group strategy (DGS): The other possibility is taking all of the products in account in the objective function, which will give different subsets of products for each shipment. Aggregating small orders with larger ones that need to be flowing more often. These are the joint methods that we will compare on the study.

The first interpretation of the DGS concept uses the EOQ as the IGS. First finding the most ordered product, then the rest of the products are allocated around that product with a vector \mathbf{k} . Once done that, the vector of optimal frequency relative to the most ordered product is calculated taking into account its cost and the minor and holding cost related to the vector \mathbf{k} (Chopra & Meindl, 2012). This method is used on the report to compare it with the rest of the models.

The traditional approach for this problem was developed by Goyal (1974). In this paper an iteration is developed that takes as variables the length of time until each truck is ordered T and a vector of variables \mathbf{k} that assign how much each product has to wait to get into the truck. First the formulas that optimize the objective function are obtained for each variable. Now, given that the formula for the cost is monotone decreasing in \mathbf{k} , a maximum value of T is taken when the vector \mathbf{k} is all one. Then a lower bound is generated and following an algorithm a finite number of sets are compared until the optimum is found.

This general algorithm was later revised by several papers for using different bounds that would come faster to a result or that will include all possible scenarios. Van Eijs (1993) improved this algorithm to include

scenarios in which the minimum $k_i > 1$. That is when there are orders of trucks that go empty. This corrected algorithm is the one used on the report.

Several restriction and special applications are developed from this algorithm. What is especially interesting for us is Porras & Dekker (2006), which developed an algorithm based on van Eijs revision that includes minimum order quantities. The problem with this restriction is that it generates some jumps in the minimum total cost function. They resolve using a quadratic equation to get both bounds in which one side includes the total cost with the T that has to be defined, and the other side is the total cost with T using a vector of $\mathbf{k} = \mathbf{1}$ and a vector of \mathbf{k} that includes the minimum order restrictions.

Another paper was needed that takes care of the capacity restriction from the models. Some papers were spotted like Sindhuchau et al (2005) or Minner & Silver (2005). However they use dynamic models that are too complex to develop for the limited time of this research. Another simpler approach is used by Hoque (2006) that takes individual restrictions on capacity and investment. However this paper is not taken for the report because the applications involve problems with the capacity in the warehouse and the real problem with the company is the truck capacities.

A simpler approach is used in the report, first the average trucks used according to capacity are used to correct the total cost (Oudenes, 2015). This approach uses the vector \mathbf{k} from Goyal algorithm and changes T to optimize the filling of the trucks. This method proved to reduce the average cost once costs are measured properly.

4. Data

The data is obtained from a Mexican distribution company named Merdiz specialized on industrial and transport lubricants. The portfolio consist of 250 products, with 134 products with positive demand and 83 with at least 1 pallet of demand. As we already stablished the minimum ordering quantity is 1 pallet.

We assume prices and transport costs are stable during this period to facilitate calculation. The prices are expressed in pallets and are updated to the month march 2015 without any discounts. As all other parameters prices are expressed in pallets.

The system in the business keeps track of any orders and supplies in the last 2 years. On this regard we have 2 years of information on sales from March 2013 to March 2015. This is the general information from the total demand.

Table 1. General overview from data

Period of Study	March 2013-March 2015
Total value of demand during the period	4,048,794 euros
Amount of pallet demand	1946
Number of products	134

The system from the company has an individual update of demand, which means all parameters from our study have to be converted to pallets. Each product has a standardized size according to their application. That involves having different pallet setting. On the table below we can see that relation. This information will be used later to find the minor cost per product on section 6.

Table 2. Set-up of pallets.

Unit	Liters per product	Products per Pallet	Layers in Pallet	Products per Layer
Drum	200	4	1	4
Pail	19	42	3	14
Tote	1040	30	1	1
12 Bottles	11.35	48	4	12
Box	-	100	4	0
IBC	1000	1	1	1
Keg	50	9	2	4

As it was shown on figure 6, most of the clients are industrial, transport or distribution companies. Those companies demand bigger sizes for their customers. On the next table we can see how relatively big are the demands for drums, IBC and buckets compared to the smaller presentations. When we arrive to the calculation of the parameters, we will see that first we will use 4 products and then 20 products to calculate the results. For getting the economic parameters as we will see in further detail in section 6, first we are summing the demands from the types and using weighted averages from the demand. We use 4 products to test all models that represent drums, pails, IBC and the rest of the products. The reason for doing that is the small percentage represented by the rest of the products as we can see on table 3. Those small

demands are handled together to get the results and then the results will be used as a guideline, from which the rest of the products will be handled individually.

Table 3. Percentage of total value of demand

Units	% of Total Value
Drum	77.41%
Pail	7.49%
12 Bottles	0.30%
Box	0.17%
IBC	14.43%
Keg	0.19%

Another reason to have a fewer sample of products that represent the total demand is the variation between the demands, not only between the types of products but between products as well. We can observe that the demand is centered in a few big sized products.

Table 4. Distribution of products according to value of total demand.

Number of Products in the Range	Total Value of Product Demand
37	0 to 1000 €
13	1000 to 3000 €
6	3000 to 5000 €
12	5000 to 10000 €
8	10000 to 50000 €
6	50000 to 100000 €
2	more than 100000 €

Table 5. Distribution of products by number of pallets

Number of Products in the Range	Total demand in pallets 2013-2015
44	1 to 5
10	6 to 10
11	11 to 20
7	20 to 50
5	50 to 100
6	more than 100

5. Models

The models will be based on the basic approach of the joint replenishment problem model made by Goyal (1976) and its revision by van Eijs (1993). To compare we use a different approach from Chopra & Meindl (2012) that relies on increase efficiency by conglomerating products around the most ordered product. Later we take an application of Goyal algorithm with the case of minimum order quantity, using the paper of Porras & Dekker (2008). Finally the major setup cost is corrected using the average trucks in a cycle. To include an analysis for truck capacity restriction.

Assumptions

- Holding costs are linear
- There is not backlog
- Demand is constant and known
- Initial inventory is assumed to be 0
- There are no price discounts
- Mixed pallets are allowed

Parameters for the Model.

S_o	major ordering cost
D_i	total demand rate in pallets for item i per year
s_i	minor ordering cost
h_i	holding cost in per year

General Concepts and problem formulation

In both models the basic two variables are the length of time between each order T_o and the multiplier k_i that generates vector \mathbf{k} and describes the length of time between each product order T_i .

$$T_i = k_i T_o \quad (1.1)$$

From this follows that the ordering for each product will be.

$$Q_i = T_i D_i \quad (1.2)$$

To know exactly what units this variables represent we proceed with an example. If the unit period we are using is 1 year then we have:

$T_o = 1/4$ of a year, $D_i = 8$ pallets per year and $k_i = 2$ for a product then:

$1/T_o = 4$ orders per year, $1/T_i =$ Product i will be ordered 2 times per year, $Q_i = 4$ pallets per order.

We always start distribution in period 0 for all products, then the ordering would be [4,0,4,0]. It is worth saying that the cycle of the product will be [4,0,4]. The cycle is when the ordering repeats itself. However, now we are just giving a general overview of the variables. In the problem with truck capacity restriction a more detailed ordering schedule example will be included with the products from the company.

T_o is always a fraction of the period of time we are studying, so for expressing it in days.

$T_o * (365 \text{ days}) = 91.25$ days between orders.

Now once we understand the general concept of the variables, we formulate the general objective function that will be used to handle the joint replenishment problem. It includes the minor and major setup cost and a linear holding cost. All of those changed by two variables: T_o and the vector of k_i which we call \mathbf{k} . This equation will be referenced several times during the report. Given that the problems part from the same equation, and assign additional restrictions like minimum order quantities or truck capacity. This function is defined as the average cost per time unit given.

$$\text{Min } TC_{T_o, k_1, \dots, k_n} = \frac{1}{T_o} \left(S_o + \sum_1^n \frac{S_i}{k_i} \right) + T_o \frac{1}{2} \sum_1^n h_i D_i k_i \quad (1.3)$$

Joint replenishment model

From Goyal (1976) we take that (1.3) can be minimized. If we take a fixed (k_1, \dots, k_n) then the basic cycle that optimizes TC is:

$$T^* = \sqrt{\frac{2 \left(S_o + \sum_1^n \frac{S_i}{k_i} \right)}{\sum_1^n D_i h_i k_i}} \quad (2.1)$$

Once T^* is substituted in (1.3) the optimal average total cost is found. This will be the outcome for comparing different models and the changes between parameters.

$$TC^* = \sqrt{2 \left(S_o + \sum_1^n \frac{S_i}{k_i} \right) \left(\sum_1^n D_i h_i k_i \right)} \quad (2.2)$$

Goyal (1976) proved that for a fixed T, the variable cost of item i is minimized by selecting an integer $k_i(T)$ which satisfies:

$$k_i(T)(k_i(T) - 1) < \frac{2S_i/D_i h_i}{T^2} \leq k_i(T)(k_i(T) + 1) \quad (2.3)$$

Originally this procedure (Goyal, 1974) obtains a maximal and minimal values of T for which a delimited number of scenarios has to be tested to find the optimum. From Van Eijs (1993) this two boundaries are corrected to include all scenarios, including $k_1 > 1$ for all i .

The formula of T^* in (2.1) is monotone decreasing in (k_1, \dots, k_n) so

T_{max} occurs in T^* with $(k_1, \dots, k_n) = (1, \dots, 1)$ then:

$$T_{max} = \sqrt{\frac{2(S_o + \sum_1^n S_i)}{\sum_1^n D_i h_i}} \quad (2.4)$$

According to Andres and Emmons (1974), if we take TC^* from (2.2) the minimum value of T should be

$$T_{min} = \frac{2S_o}{TC^*}$$

Yet in the first approach we only have an approximation of TC^* using (2.3) and (2.4). Once we have the boundaries in which the optimal values can be found. We proceed to follow an iterative procedure that will select a k_j on each iteration to be summed as $k_j = k_j + 1$. Then T and TC^* are calculated again and the iteration repeats until an optimum is reached. The procedure is the next:

Modified algorithm of Goyal

Step 1: Initialization

- a) $(k_1^*, \dots, k_n^*) := (1, \dots, 1)$
Use (2.2) to get $TC^* := TC(1, \dots, 1)$
- b) Set $T_{max} := TC^* / \sum_1^n D_i h_i$
Determine $(k_1, \dots, k_n) := (k_1(T_{max}), \dots, k_n(T_{max}))$ with (2.3)
Determine $TC(k_1, \dots, k_n)$ with (2.2)
- c) If $TC(k_1, \dots, k_n) < TC^*$, then
 $(k_1^*, \dots, k_n^*) := (k_1, \dots, k_n)$
 $TC^* := TC(k_1, \dots, k_n)$
Go back to step 1(b)
Else go to step 1(d)
- d) Set $T_{min} := 2A/TC^*$
Set $T_{ch} := \sqrt{2s_i / (D_i h_i k_i (k_i + 1))}$. This is the value that denotes the product that will change k_i for $k_i + 1$

Step 2: Set $T := \max_i T_{ch}(i)$

- If $T \leq T_{min}$, then go to Step 4
Else go to Step 3

Step 3: Evaluation of the cost in the new order

- Set $p := \arg \max_i T_{ch}(i)$ (where p is the item that change from k_i to $k_i + 1$)
Set $k_p := k_p + 1$ and Set $T_{ch}(p) = \sqrt{2s_p / (D_p h_p k_p (k_p + 1))}$
Calculate $TC(k_1, \dots, k_n)$ with (2.2)
If $TC(k_1, \dots, k_n) < TC^*$ then
 $(k_1^*, \dots, k_n^*) := (k_1, \dots, k_n)$
 $TC^* := TC(k_1, \dots, k_n)$
 $T_{min} := 2A/TC^*$
Go back to Step 2

Step 4. Termination of the algorithm

The optimal strategy is $(T^*; k_1^*, \dots, k_n^*)$ and TC^* . Final T^* is obtained from (2.1) with vector \mathbf{k} .

JRP with Minimum Order Quantity Restriction

This formulation is obtained from Porras & Dekker (2008). In this paper the original total cost function is changed with a correction factor $\Delta(\mathbf{k})$. Which refers to the fraction of non-empty replenishments arising to correct when the smallest $k_i > 1$.

So we will take equation (1.3) than defines the general JRP. However on Porras & Dekker (2008) they do not use minor setup cost for the algorithm, because they work with full containers. So the cost per product is only expressed on the holding cost. We will follow their concept even if it will reflect on the outcome from the model. The minor cost for the company on this report are very low given the small labor cost in Mexico so the comparison with and without MOQ will still be valid.

$$TC(T, \mathbf{k}) = \frac{A\Delta(\mathbf{k})}{T} + \frac{1}{2}T \sum_1^n h_i k_i D_i \quad (3.1)$$

This correction factor functions with the idea that there are setup cost that will not be charged because of the gaps created when $k_i > 1$. Then those gaps are counted using the principle of inclusion-exclusion. Adding a series that includes each product, pairs, triples and so on. The operator lcm denotes the least common multiple of all those combinations.

$$\Delta(\mathbf{k}) = \sum_1^n (-1)^{i+1} \sum_{\{\alpha \subset \{1, \dots, n\}; |\alpha|=i\}} (\text{lcm}(k_{\alpha_1}, \dots, k_{\alpha_i}))^{-1}$$

$$\Delta(\mathbf{k}) = \sum_1^n \frac{1}{k_i} - \sum_{(i,j) \subseteq \{1, \dots, n\}} \frac{1}{(\text{lcm}(k_i, k_j))} + \sum_{(i,j,k) \subseteq \{1, \dots, n\}} \frac{1}{(\text{lcm}(k_i, k_j, k_k))} - \dots + (-1)^{n+1} \frac{1}{(\text{lcm}(k_1, \dots, k_n))} \quad (3.2)$$

Porras & Dekker (2008) found proof that if at least one of the $k_i = 1$ then $\Delta(\mathbf{k}) = 1$. This concept is quickly commented for its implications but will not be used on the model given that almost all cases have at least one product with $k_i = 1$. The idea of having a correction factor is especially useful when there are MOQ, given that the iteration arrives to high values for the vector \mathbf{k} . This actually happened for 2 scenarios on this report that include extreme measures on low major setup cost and very high MOQ. However when we have at least one $k_i = 1$ the results are the same with or without correction factor.

We formulate the problem with the minimum order quantity restriction (MOQ)

$$\begin{aligned} \text{Min } TC(T, \mathbf{k}) &= \frac{A}{T} + \frac{1}{2}T \sum_1^n h_i k_i D_i & (3.3) \\ \text{s. t. } &k_i D_i T \geq \text{MOQ}_i \text{ for all } i \\ &T > 0 \\ &k_i \geq 1 \text{ integers for } i = 1, \dots, n \end{aligned}$$

Now for a fixed vector $\mathbf{k} = (k_1, \dots, k_n)$, the T value minimizing TC is:

$$T^* = \sqrt{\frac{2A}{\sum_1^n h_i k_i D_i}} \quad (3.4)$$

For a fixed T, the optimal value of \mathbf{k} is:

$$k_i(T) = \left\lceil \frac{\text{MOQ}_i}{D_i T} \right\rceil \text{ for } i = 1, \dots, n \quad (3.5)$$

With this values Porras & Dekker (2008) generate the first approach to the bounds using T^* with a vector of $(\mathbf{k}) = (1, \dots, 1)$

$$T_{unc}^* = \sqrt{\frac{2A}{\sum_1^n h_i D_i}} \quad (3.6)$$

Then a vector $\mathbf{k}^{(0)}$ is defined with that value T_{unc}^*

$$\mathbf{k}_i^{(0)} = \left\lceil \frac{\text{MOQ}_i}{D_i T_{unc}^*} \right\rceil \text{ for } i = 1, \dots, n \quad (3.7)$$

Now for generating the bounds a quadratic function with the total cost function is set for getting both limits. First TC_{low} is defined as the cost curve with vector $(\mathbf{k}) = (1, \dots, 1)$. This value has to be equal to the total cost that contains the MOQ restrictions, the solution from that equation will be the bounds:

$$TC_{low}(T) = TC(T_{unc}^*, \mathbf{k}^{(0)})$$

The same expression is shown extended below.

$$\frac{A}{T} + \frac{1}{2}T \sum_1^n h_i k_i D_i = \frac{A}{T_{unc}^*} + \frac{1}{2}T_{unc}^* \sum_1^n h_i \left\lfloor \frac{MOQ_i}{D_i T_{unc}^*} \right\rfloor D_i \quad (3.8)$$

Where the lower value of T will be T_{low}^0 and the upper bound T_{upp}^0 .

The same procedures that we find in Goyal and van Eijs are followed for the iteration, just that for this case the definition of the bounds and the optimal k_i and T are changed by the minimal order quantity, we follow the notation of the paper. But the concept will be the same, finding two bounds that will contain the optimal value and make those bounds closer until an optimum is found.

Algorithm Joint Replenishment Problem with MOQ

Step 0. Evaluate T_{low}^0 and T_{upp}^0 determine $\mathbf{k}^{(0)}$ with (3.7)

$$\text{Set } T^{(0)} = \min_i \{T_i^{(0)}\}$$

Where

$$T_i^{(0)} = \begin{cases} \frac{MOQ_i}{D_j} \frac{1}{k_j^{(0)} - 1} & \text{if } k_j^{(0)} > 1, \\ \infty & \text{if } k_j^{(0)} = 1 \end{cases}$$

$$\text{Set } TC_{min}^{(0)} = \infty \text{ and } n = 1$$

Step 1. For $\mathbf{k}^{(n-1)}$ determine $T^{(n)}$ from:

$$T^{(n)} = \max_j \left\{ \frac{MOQ_j}{D_j k_j^{(n-1)}} \right\} \text{ and set: } J^{(n)} = \left\{ i: \max_{i=1, \dots, n} \left\{ \frac{MOQ_i}{D_i k_j^{(n-1)}} \right\} \right\}$$

Evaluate T_{n-1}^* using (3.4) with $\mathbf{k}^{(n-1)}$

$$\text{Set } TC_{min}^{(n)} = \begin{cases} \min \{TC_{min}^{(n-1)}, TC(T_{n-1}^*, \mathbf{k}^{(n-1)})\} & \text{if } T_{n-1}^* \in [T^{(n)}, T^{(n-1)}] \\ \min \{TC_{min}^{(n-1)}, TC(T^{(n)}, \mathbf{k}^{(n-1)}), TC(T^{(n)}, \mathbf{k}^{(n)})\} & \text{otherwise} \end{cases}$$

Where

$$k_j^{(n)} = \begin{cases} k_j^{(n-1)} + 1 & \text{for } j \in J^{(n)} \\ k_j^{(n-1)} & \text{otherwise} \end{cases}$$

Step 2. If $T^{(n)} \leq T_{low}^0$ STOP and set $TC_{min} = TC_{min}^{(n)}$. Otherwise set $n = n + 1$ and Go to Step 1

End of algorithm

Joint Replenishment Problem adjusting to most ordered product

For this method first all products are calculated separately to look for the most ordered product. The individual orders are taken from the economic order quantity formula (EOQ) from Chopra & Meindl (2012). However the cost have to be changed, instead of just ordering cost we have the minor and major setup cost per product.

$$S_i^* = s_i + S_o \quad (4.1)$$

Then EOQ formula comes from solving the simplest replenishment problem that includes only holding cost and ordering costs.

$$TC_i = \left(\frac{D_i}{Q_i}\right) (S_i^*) + \left(\frac{Q_i}{2}\right) h_i \quad \forall i$$

We find the optimal order quantity by taking first order derivatives.

$$Q_i^* = \sqrt{\frac{2D_i S_i^*}{h_i}} \quad \forall i$$

Finally for the ordering frequency for each product, we just divide the demand by the optimal quantity. The notation $1/T_i$ was chosen instead of just n_i to be comparable with the last models shown.

$$\frac{1}{T_i} = \frac{D_i}{Q_i^*} = \sqrt{\frac{D_i h_i}{2S_i^*}} \quad \forall i \quad (4.2)$$

Now, the most ordered product is selected to be a reference for the other products which will be ordered with it.

$$\frac{1}{T} = \frac{1}{T_i^*} = \max \left\{ \frac{1}{T_i}, i = 1, \dots, n \right\} \quad (4.3)$$

The ordering from the rest of the product is calculated again with a minimum boundary that includes only the minor setup cost.

$$\frac{1}{T_i} = \sqrt{\frac{D_i h_i}{2S_i}} \quad \forall i \quad (4.4)$$

A multiple k_i will be referencing the relative ordering. The most ordered product will get $k_i^*=1$, given that is ordered every time. The rest of the products will follow this formula.

$$k_i = \left\lceil \frac{1/T}{1/T_i} \right\rceil \quad \forall i \quad (4.5)$$

Now to get the final ordering cycle or the ordering from the most ordered product. We use equation (2.1) which is obtained from Goyal (1974).

$$\frac{1}{T_o} = \sqrt{\frac{\sum_1^n h_i k_i D_i}{2(S_o + \sum_1^n k_i)}} \quad (4.6)$$

Finally the ordering per each product is corrected following $1/T_o$.

$$\frac{1}{T_i} = \left(\frac{1}{T_o k_i} \right) \forall i \quad (4.7)$$

Now we have a simpler procedure for solving JRP. If we can observe the essence of the method is following also an iterative process. It gives a maximum and minimum boundaries for the product ordering and finishes with Goyal formula. However it will not give the optimal because the results are only corrected once.

Joint Replenishment Model with Truck Capacity

The problem once we incorporate truck capacity is that more than one truck is needed to refill stock, then the cost is not properly measured because the major setup cost is just the fixed cost of one truck. Oudenes (2015) uses the average trucks used per order \mathbf{B} . Then those extra trucks are included in the input of the model. If we recall the model from equation (1.3) with this change we have equation (5.1).

$$\text{Min } TC_{T_o, k_1, \dots, k_n} = \frac{1}{T_o} \left(BS_o + \sum_1^n S_i / k_i \right) + T_o \frac{1}{2} \sum_1^n h_i D_i k_i \quad (5.1)$$

The same algorithm from Van Eijs (1993) will be used to get the output and compare the cost with the new major cost. However for getting the results we need to know how to get the truck schedule. The way to do it is taking each cycle. That means the pattern that shows when all the orders have been delivered. It is a cycle because that pattern will repeat itself during the period chosen, which in this case will be 2 years. First we will take a specific result from Goyal that will be shown in the next section. We will take the setting shown below for the input. The variable w is the number of workers are needed for the shift on the truck unloading, which will affect the minor setup cost.

$$S_o = 500 \quad w = 5$$

Once we run the algorithm the result for the vector \mathbf{k} is:

Table 6. Outcome \mathbf{k}

k1	k2	k3	k4
1	2	1	6

For getting the length of the cycle we have to take the least common multiple (lcm) of the vector \mathbf{k} . That means for this case we will have a pattern of distribution that will repeat itself until the demand is met. This pattern will give us the average of trucks. First, we show in which periods there will be an order from a product with a binary matrix.

Table 7. Decision ordering schedule

Product \ Period	0	1	2	3	4	5	6
1	1	1	1	1	1	1	1
2	1	0	1	0	1	0	1
3	1	1	1	1	1	1	1
4	1	0	0	0	0	0	1

From this distribution schedule we need to get the ordering following equation (1.2), the numbers for each product are.

Table 8. Ordering quantity per product

Product	D	T	k	O
1	1,539	0.025	1	38.5
2	170	0.025	1	4.3
3	223	0.025	1	5.6
4	14	0.025	4	1.4

In table 9 we see the decision matrix multiplied by the ordering matrix for each product, this will be the pallet ordering schedule for the cycle.

Table 9. Pallet ordering schedule

Product\Period	0	1	2	3	4	5	6
1	38.5	38.5	38.5	38.5	38.5	38.5	38.5
2	4.3	0	4.3	0	4.3	0	4.3
3	5.6	5.6	5.6	5.6	5.6	5.6	5.6
4	1.4	0	0	0	0	0	1.4
Total	49.8	44.1	48.4	44.1	48.4	44.1	49.8

We sum the number of pallets for each order and adjust the number of trucks according to the capacity. For this case if we take a truck capacity of 24 pallets, and the average from the cycle is 2.5 trucks, which come from the number of trucks ordered. That will be the value for B to adjust the major setup cost. To define this variable B is the average number of trucks over all cycles.

Table 10. Truck schedule

Product\Period	0	1	2	3	4	5	6
Number of trucks needed	2.07	1.84	2.01	1.84	2.01	1.84	2.07
Number of trucks ordered	3	2	3	2	3	2	3

So far we have the parameter that will help us to correct the total cost using equation (5.1). This parameter however cannot be used inside Goyal algorithm because $B(Cap; T; k_1, \dots, k_n)$ depends on T and vector k , as well as the capacity of the truck. Then if we want to use it in Goyal algorithm we will be disrupting equation (2.1) that leads to an optimal T . Several iterations were tried and even if the T^* is replaced by T_{ch} or T_{min} the results are an overall increase in cost. And that is given due to the fact that by adding truck capacity, if we use any B to correct cost, then the cost will increase and that will lead to less orders with more pallets each time. That will increase the number of trucks but it does anything to improve the truck filling efficiency. That is why the next procedure is proposed in which the average filling percentage of the trucks should be maximized modifying the final T .

So we will run the normal Goyal algorithm and in Step 4 instead of selecting T^* from equation (2.1). We follow the next simple procedure, first we define.

$O_{it}(k_i)$	This variable indicates if a certain product has an order in a certain ordering period. This is a binary variable depending on k_i
D_i	Demands in pallets per year
T	Time in between orders, this will be the variable to define in the next method
T^*	Optimal time between orders chosen by Goyal algorithm
(k_1^*, \dots, k_n^*)	Optimal vector \mathbf{k} obtained from Goyal algorithm
<i>Capacity</i>	The capacity of trucks in pallets
<i>lcm</i>	Least common multiple from vector (k_1, \dots, k_n)

Now we first give a formal definition of B . We had already shown manually how the variable is constructed, so for getting the formal definition, we need the average in a cycle of the rounded up numbers of the pallets ordered in every period divided by the capacity of the trucks. We have that a cycle is the period in which at least all products are ordered twice. As we see we have $lcm(k_1, \dots, k_n) + 1$ because we start at period 0, where all the products are ordered.

$$B = \frac{\sum_{t=0}^{lcm} \left(\frac{\sum_{i=1}^n O_{it} D_i T k_i}{Capacity} \right)}{lcm(k_1, \dots, k_n) + 1} \quad (5.2)$$

Now what we want to improve is the efficiency in the truck filling, so for getting that we need to get the average of the division between the pallets ordered and the pallet capacity from the trucks ordered.

$$Fill\%Trucks = \frac{\sum_{t=0}^{lcm} \left(\frac{\frac{\sum_{i=1}^n O_{it} D_i T k_i}{\left(\frac{\sum_{i=1}^n O_{it} D_i T k_i}{Capacity} \right) Capacity}}{lcm(k_1, \dots, k_n) + 1} \right)}{lcm(k_1, \dots, k_n) + 1} \quad (5.3)$$

If we take $Fill\%Trucks$ as the objective function, we have that we will maximize the filling percentage of the trucks in a cycle. The variable to optimize the objective function will be T . The key restriction will be that the total average cost has to be bigger than the one with the optimal variables obtained with the Goyal algorithm. The results from this problem are solved with Excel Solver.

$$\max_T \frac{\sum_{t=0}^{lcm} \left(\frac{\frac{\sum_{i=1}^n O_{it} D_i T k_i}{\left(\frac{\sum_{i=1}^n O_{it} D_i T k_i}{Capacity} \right) Capacity}}{lcm(k_1, \dots, k_n) + 1} \right)}$$

$$\text{s.t.} \quad TCR(T; k_1^*, \dots, k_n^*) \leq TCR(T^*; k_1^*, \dots, k_n^*), \quad k_j \geq 1, \quad D_j \geq 0, \quad O_{it} = \{0,1\}, \quad Capacity \geq 1$$

This function is not continuous so the problem is difficult to maximize. However the formal definition of the problem is complete and the results show improvements in cost while increasing filling efficiency. The

lcm has to be also restricted, if it is larger than the number of orders in a time period, then the number of orders will be used to obtain the averages for B and $Fill \%Trucks$.

6. Calculation of economic parameters

In the result section we use the data first from 4 products to show clearly the vector \mathbf{k} and all the procedures explained before (Goyal, MOQ and truck capacity). Later we expand these results for 20 products to include the Goyal algorithm with truck capacity restriction. We use the next methods to obtain the economic parameters.

Parameters per product

Extraction of parameters for 4 products

The products that are selected are not specific but 4 types of products: drums, buckets, IBC and the rest types with smaller demands. We define the next variables to show the formal method to obtain the parameters.

i	Product i
j	Type of product j according to product i
D_{ij}	Annual demand in pallets per product with a certain type of product.
p_{ij}	Price per pallet of a certain product with a certain type of product
s_{ij}	Minor setup cost per pallet of a certain product with a certain type of product

A new demand is proposed that will be the total demand for each type of product

$$D_j = \sum_{i=1}^n D_{ij} \quad \text{for each } j$$

The price is calculated with a weighted average according to the demand of each type

$$p_j = \sum_{i=1}^n \frac{D_{ij} p_{ij}}{D_j} \quad \text{for each } j$$

The minor setup cost is calculated in the same way as the price

$$s_j = \sum_{i=1}^n \frac{D_{ij} s_{ij}}{D_j} \quad \text{for each } j$$

On Appendix 1 we can see the product parameters once we used this procedure for the 4 types of products.

Extraction of parameters for 20 products

For the last case on the result section we use 20 products instead of 4. The case shown is Goyal method with truck capacity restriction. The reason for using 20 products instead of 83 is that when we use the 83 products the results for most of the products are less than 1 pallet which does not reflect the operation of the company that needs at least 1 pallet. So what is done instead is select the 19 products with the greatest annual demand per pallet with their respective demands and prices. For the 20th product an aggregation is made with the rest of the products with smaller demand. The same procedure as before is used with a weighted average depending on demand. For practical purposes if we want to get the complete truck schedule with 83 products, we can manage the rest of the products individually with a guideline already set. On appendix 2 we can observe the data we used for obtaining the results.

Minor Setup Cost

We assume that the minor setup cost each product has is related to the wage cost of unloading the product and setting it up for the customers. We need to define how much time in minutes that operation takes. The idea behind it is that the size of the product will involve different handlings. When a pallet is unloaded it always will be done with a forklift, for all products that will be the same. However the handling of the product when is opened and setup for distribution will take different times according to its size and number. There is also a number of workers required during the operation that will change the cost.

<i>Avg_i</i>	Average time in minutes for handling a unit of product
<i>Pallet_i</i>	Number of products that fit in a pallet
<i>Wage</i>	Wage per minute of a warehouse worker in the company
<i>Workers</i>	Amount of workers used during the operation

Then the minor cost will be defined as.

$$S_i = Avg_i * Pallet_i * Wage * Workers$$

The next table shows how the calculation done depending on the kind of presentation the product has.

Table 11. Minor setup cost setting

Unit	Liters	Packages per Pallet	Average Time Unit (Minutes)	Wage per minute €	Workers needed during operation	Minor Cost Pallet €
Drum	200	4	16	0.06	1	3.94
Pail	19	42	6	0.06	1	15.53
IBC	1000	1	20	0.06	1	1.23
Rest	-	-	16	0.06	1	11.69

General parameters

The general parameters changed to test the models will be the major and minor setup costs, minimum ordering quantities and capacities of the trucks. All the rest of the parameters will remain constant. In the next table we show the ranges on which we will test the different scenarios for the economic parameter and the justification why those values are reasonable.

Table 12. Description of general economic parameters

Parameter	Justification	Value range
<i>S_o</i>	The average value per truck the company faces is around 610 euros	{50, 250, 500, 750, 1000}
<i>r</i>	The interest rate requested to Mexican companies for loans	{16%}
<i>Workers</i>	The number of workers in operation changes proportionally the minor setup cost. For a normal operation 5 workers are available	{1, 3, 5, 7, 10}
<i>Capacity</i>	The capacity of the trucks from the company are 24 pallets and on some exceptions smaller trucks are used	{5, 10, 20, 24}
<i>MOQ</i>	The minimum ordering quantity from the provider is 1 pallet per product	{1, 3, 5, 7, 10}

7. Results

7.1 Examples to represent all models (4 products)

As we explained in previous sections in the first part of the results we took 4 products that represent the 4 different categories of products: Drums, buckets, IBC and the rest. The reason is to have a setting that will allow us to test all models and be more transparent showing the vector \mathbf{k} . The calculation of parameters for the 4 products can be reviewed in section 6 and the data used for the results is on Appendix 1. On table 13 we can observe the results from Goyal on the left side and on the right side from Chopra. The two parameters changed are the major setup cost and the amount of workers needed in the unloading operation (this variable affects minor setup cost, refer to section 6). By observing the results on costs from Goyal algorithm are always equal or slightly less than the one obtained with the Chopra & Meindl (2012), and the annual orders are almost the same.

However percent changes are really small. The reason for that is both use the same formulas for adjusting T , and in this case the minor costs are relatively small so a change in vector \mathbf{k} does not reflect in a large change in total costs.

Even if the results are tied for both methods we can see that the next effects are observed while changing the parameters:

- When the major setup cost increases and we fix the minor setup cost. The time between orders increases and the number of order decreases. Also k_i increases which means the products will be ordered together more often given that number of orders decrease.
- When the minor setup cost increases and we fix the major setup cost. k_i increases and that means some products will be delayed with a higher order. Also the number of orders will decrease given that now is more expensive to load products on the truck.

Table 13. Results 4 products. Changing major and minor setup costs

So	w(Si)	Goyal						Chopra & Meindl						% ΔTRC
		Orders	TRC	k1	k2	k3	k4	Orders	TRC	k1	k2	k3	k4	
50	1	53.2	6,900	1	2	1	6	53.2	6,900	1	2	1	6	0.00%
	3	48.6	8,262	1	3	1	9	48.6	8,262	1	3	1	9	0.00%
	5	43.3	9,323	1	3	1	10	46.2	9,350	1	4	1	10	0.29%
	7	42.6	10,246	1	4	1	12	42.6	10,246	1	4	1	12	0.00%
	10	38.3	11,456	1	4	1	13	38.3	11,456	1	4	1	13	0.00%
250	1	24.6	13,504	1	1	1	3	24.6	13,504	1	1	1	3	0.00%
	3	22.8	14,645	1	1	1	4	24.7	14,690	1	2	1	4	0.31%
	5	23.8	15,429	1	2	1	6	23.8	15,429	1	2	1	6	0.00%
	7	22.8	16,122	1	2	1	6	22.8	16,122	1	2	1	6	0.00%
	10	21.6	17,098	1	2	1	7	23.2	17,154	1	3	1	7	0.33%
500	1	17.7	18,639	1	1	1	2	17.7	18,639	1	1	1	2	0.00%
	3	17.0	19,521	1	1	1	3	17.0	19,521	1	1	1	3	0.00%
	5	16.4	20,327	1	1	1	4	17.7	20,496	1	2	1	4	0.83%

	7	17.3	21,042	1	2	1	5	17.3	21,042	1	2	1	5	0.00%
	10	16.8	21,821	1	2	1	6	16.8	21,821	1	2	1	6	0.00%
750	1	14.6	22,635	1	1	1	2	14.6	22,635	1	1	1	2	0.00%
	3	14.2	23,390	1	1	1	3	14.2	23,390	1	1	1	3	0.00%
	5	13.8	24,078	1	1	1	3	13.8	24,078	1	1	1	3	0.00%
	7	13.5	24,736	1	1	1	4	14.5	24,988	1	2	1	4	1.02%
	10	14.2	25,662	1	2	1	5	14.2	25,662	1	2	1	5	0.00%
1000	1	12.6	26,014	1	1	1	1	12.7	26,025	1	1	1	2	0.04%
	3	12.4	26,690	1	1	1	2	12.4	26,704	1	1	1	3	0.05%
	5	12.2	27,309	1	1	1	3	12.2	27,309	1	1	1	3	0.00%
	7	11.9	27,901	1	1	1	3	12.0	27,910	1	1	1	4	0.03%
	10	11.6	28,746	1	1	1	4	12.5	28,986	1	2	1	4	0.83%

In the next table we observe that the slight changes in vector \mathbf{k} between models does not reflect much in the cost and they correct themselves with the holding costs. We include the average CPU time to compare it with the model from Porras & Dekker, now we will see this times are relatively small for all scenarios. We can see the next observations as the parameters are changed:

- If the parameter for major setup cost increases fixing the minor setup cost parameter. Then the total minor setup cost decreases. The reason for that is the vector \mathbf{k} decreases and number of orders as well. That means bigger and less frequent orders are made for most of the products.
- If the parameter for minor setup cost increases fixing the parameter for major setup cost, the holding cost increases. Given that less orders are made for some products, the orders are getting bigger on average and that increases holding costs.

Table 14. Breakdown of average total costs on table 13

So	w(Si)	Goyal						Chopra & Meindl						% ΔTRC
		TRC	Major setup cost	Minor setup cost	Holding cost	T^*	Average CPU time	TRC	Major setup cost	Minor setup cost	Holding cost	T^*	Average CPU time	
50	1	6,900	2,659	792	3,450	0.019	0.67	6,900	2,659	792	3,450	0.019	-	0.00%
	3	8,262	2,431	1,700	4,131	0.021	0.96	8,262	2,431	1,700	4,131	0.021	-	0.00%
	5	9,323	2,166	2,495	4,662	0.023	0.89	9,350	2,311	2,364	4,675	0.022	-	0.29%
	7	10,246	2,131	2,992	5,123	0.023	1.17	10,246	2,131	2,992	5,123	0.023	-	0.00%
	10	11,456	1,915	3,813	5,728	0.026	1.33	11,456	1,915	3,813	5,728	0.026	-	0.00%
250	1	13,504	6,147	605	6,752	0.041	0.33	13,504	6,147	605	6,752	0.041	-	0.00%
	3	14,645	5,705	1,617	7,323	0.044	0.36	14,690	6,171	1,175	7,345	0.041	-	0.31%
	5	15,429	5,945	1,770	7,715	0.042	0.56	15,429	5,945	1,770	7,715	0.042	-	0.00%
	7	16,122	5,689	2,372	8,061	0.044	0.48	16,122	5,689	2,372	8,061	0.044	-	0.00%
	10	17,098	5,396	3,153	8,549	0.046	0.60	17,154	5,792	2,785	8,577	0.043	-	0.33%

500	1	18,639	8,850	470	9,320	0.056	0.31	18,639	8,850	470	9,320	0.056	-	0.00%
	3	19,521	8,505	1,255	9,760	0.059	0.36	19,521	8,505	1,255	9,760	0.059	-	0.00%
	5	20,327	8,221	1,942	10,163	0.061	0.39	20,496	8,845	1,403	10,248	0.057	-	0.83%
	7	21,042	8,667	1,854	10,521	0.058	0.70	21,042	8,667	1,854	10,521	0.058	-	0.00%
	10	21,821	8,407	2,503	10,910	0.059	0.58	21,821	8,407	2,503	10,910	0.059	-	0.00%
750	1	22,635	10,931	387	11,318	0.069	0.35	22,635	10,931	387	11,318	0.069	-	0.00%
	3	23,390	10,647	1,048	11,695	0.070	0.30	23,390	10,647	1,048	11,695	0.070	-	0.00%
	5	24,078	10,343	1,696	12,039	0.073	0.36	24,078	10,343	1,696	12,039	0.073	-	0.00%
	7	24,736	10,133	2,234	12,368	0.074	0.33	24,988	10,883	1,611	12,494	0.069	-	1.02%
	10	25,662	10,660	2,171	12,831	0.070	0.71	25,662	10,660	2,171	12,831	0.070	-	0.00%
1000	1	26,014	12,599	408	13,007	0.079	0.25	26,025	12,676	337	13,013	0.079	-	0.04%
	3	26,690	12,360	984	13,345	0.081	0.30	26,704	12,434	918	13,352	0.080	-	0.05%
	5	27,309	12,159	1,496	13,655	0.082	0.29	27,309	12,159	1,496	13,655	0.082	-	0.00%
	7	27,901	11,901	2,049	13,950	0.084	0.40	27,910	11,975	1,980	13,955	0.084	-	0.03%
	10	28,746	11,626	2,747	14,373	0.086	0.36	28,986	12,509	1,984	14,493	0.080	-	0.83%

Results minimum order quantities (4 products)

Once we tested the efficiency of the Goyal algorithm, we now test the application with MOQ. For this case Porras & Dekker (2008) method is used, which does not use minor setup cost in their algorithm. However, when the average total cost is calculated, we take the resulting \mathbf{k} vector and use equation (2.2) of total average cost that does include minor setup cost. So when we calculate with Goyal algorithm we take a level of $Workers = 5$ to define the minor setup cost for the input in Goyal algorithm, but also to compare the average total cost.

We variate the major setup cost and MOQ. As we can see from 1 pallet there is already an increase in the average cost compared to the same scenarios run with the normal Goyal algorithm. The cost increases in great measure as the MOQ increases. However, for the case that we are studying which is $MOQ = 1$ pallet, the distribution does not increase the cost a lot for a normal level of major setup cost. Porras & Dekker method gives an optimal strategy for this case without increasing cost in great measure.

Table 15. Results for 4 products with MOQ. Changing major setup cost and MOQ

So	w(Si)	MOQ	Goyal							Porras & Dekker					% Δ TRC
			Orders	TRC	k1	k2	k3	k4	Orders	TRC	k1	k2	k3	k4	
50	5	1	43.3	9,323	1	3	1	10	59.1	12,342	1	1	1	11	32.4%
		3	43.3	9,323	1	3	1	10	81.1	11,501	1	4	4	49	23.4%
		5	43.3	9,323	1	3	1	10	102.3	13,275	1	9	7	98	42.4%
		7	43.3	9,323	1	3	1	10	130.9	15,230	2	14	10	160	63.4%
		10	43.3	9,323	1	3	1	10	170.0	18,766	3	24	18	285	101.3%
250	5	1	23.8	15,429	1	2	1	6	25.9	15,957	1	1	1	5	3.4%

	5	3	23.8	15,429	1	2	1	6	27.8	15,805	1	2	1	15	2.4%
	5	5	23.8	15,429	1	2	1	6	33.1	18,180	1	3	3	33	17.8%
	5	7	23.8	15,429	1	2	1	6	37.1	19,983	1	5	4	51	29.5%
	5	10	23.8	15,429	1	2	1	6	43.5	23,110	1	8	6	87	49.8%
500	5	1	16.4	20,327	1	1	1	4	18.2	20,464	1	1	1	3	0.7%
	5	3	16.4	20,327	1	1	1	4	18.6	20,670	1	1	1	10	1.7%
	5	5	16.4	20,327	1	1	1	4	21.1	22,433	1	2	2	20	10.4%
	5	7	16.4	20,327	1	1	1	4	23.4	24,553	1	3	3	33	20.8%
	5	10	16.4	20,327	1	1	1	4	26.3	27,305	1	5	4	52	34.3%
750	5	1	13.8	24,078	1	1	1	3	14.9	24,148	1	1	1	3	0.3%
	5	3	13.8	24,078	1	1	1	3	15.1	24,353	1	1	1	8	1.1%
	5	5	13.8	24,078	1	1	1	3	16.0	25,107	1	2	1	14	4.3%
	5	7	13.8	24,078	1	1	1	3	17.3	27,108	1	2	2	23	12.6%
	5	10	13.8	24,078	1	1	1	3	19.8	30,563	1	4	3	39	26.9%
1000	5	1	12.2	27,309	1	1	1	3	12.8	27,391	1	1	1	2	0.3%
	5	3	12.2	27,309	1	1	1	3	13.1	27,563	1	1	1	7	0.9%
	5	5	12.2	27,309	1	1	1	3	13.3	27,950	1	1	1	12	2.3%
	5	7	12.2	27,309	1	1	1	3	14.9	30,763	1	2	2	20	12.6%
	5	10	12.2	27,309	1	1	1	3	16.5	33,906	1	3	3	33	24.2%

As we said before the total cost are compared with the equation (1.3) that describes the basic JRP, the total cost are compared with the optimal T and vector \mathbf{k} from each method. What we see is on one side an increase in MOQ increases the number of orders, but also increases k_i , which means the orders are less frequent for some products. So the only way to see if the orders are bigger or smaller is with the holding cost. We can see that the holding cost are getting bigger while MOQ increases and that means the orders are getting bigger, so in other words orders are less frequent. On the same line, the minor setup cost decrease because some products now are ordered less frequently, but the major cost are up because the ordering overall increases.

Table 16. Breakdown of average total costs on table 15

So	w(Si)	MOQ	Goyal					Porras & Dekker					% Δ TRC		
			TRC	Major setup cost	Minor setup cost	Holding cost	T^*	Average CPU time	TRC	Major setup cost	Minor setup cost	Holding cost		T^*	Average CPU time
50	5	1	9,323	2,166	2,495	4,662	0.023	0.71	12,342	2,955	6,432	2,955	0.017	0.49	32.4%
	5	3	9,323	2,166	2,495	4,662	0.023	0.66	11,501	4,054	3,394	4,054	0.012	3.61	23.4%
	5	5	9,323	2,166	2,495	4,662	0.023	0.71	13,275	5,113	3,049	5,113	0.010	8.78	42.4%
	5	7	9,323	2,166	2,495	4,662	0.023	0.71	15,230	6,543	2,144	6,543	0.008	14.48	63.4%
	5	10	9,323	2,166	2,495	4,662	0.023	0.71	18,766	8,502	1,761	8,502	0.006	25.12	101.3%

250	5	1	15,429	5,945	1,770	7,715	0.042	0.44	15,957	6,484	2,988	6,484	0.039	0.30	3.4%
	5	3	15,429	5,945	1,770	7,715	0.042	0.45	15,805	6,949	1,907	6,949	0.036	0.71	2.4%
	5	5	15,429	5,945	1,770	7,715	0.042	0.46	18,180	8,272	1,635	8,272	0.030	2.21	17.8%
	5	7	15,429	5,945	1,770	7,715	0.042	0.46	19,983	9,287	1,409	9,287	0.027	3.96	29.5%
	5	10	15,429	5,945	1,770	7,715	0.042	0.47	23,110	10,878	1,354	10,878	0.023	8.00	49.8%
500	5	1	20,327	8,221	1,942	10,163	0.061	0.30	20,464	9,111	2,241	9,111	0.055	0.25	0.7%
	5	3	20,327	8,221	1,942	10,163	0.061	0.30	20,670	9,316	2,038	9,316	0.054	0.43	1.7%
	5	5	20,327	8,221	1,942	10,163	0.061	0.28	22,433	10,536	1,360	10,536	0.047	1.15	10.4%
	5	7	20,327	8,221	1,942	10,163	0.061	0.38	24,553	11,699	1,156	11,699	0.043	2.33	20.8%
	5	10	20,327	8,221	1,942	10,163	0.061	0.38	27,305	13,154	997	13,154	0.038	4.01	34.3%
750	5	1	24,078	10,343	1,696	12,039	0.073	0.44	24,148	11,159	1,830	11,159	0.067	0.42	0.3%
	5	3	24,078	10,343	1,696	12,039	0.073	0.43	24,353	11,339	1,675	11,339	0.066	0.50	1.1%
	5	5	24,078	10,343	1,696	12,039	0.073	0.50	25,107	12,002	1,102	12,002	0.062	0.99	4.3%
	5	7	24,078	10,343	1,696	12,039	0.073	0.43	27,108	12,998	1,112	12,998	0.058	1.93	12.6%
	5	10	24,078	10,343	1,696	12,039	0.073	0.40	30,563	14,859	846	14,859	0.050	3.87	26.9%
1000	5	1	27,309	12,159	1,496	13,655	0.082	0.61	27,391	12,843	1,705	12,843	0.078	0.30	0.3%
	5	3	27,309	12,159	1,496	13,655	0.082	0.41	27,563	13,051	1,460	13,051	0.077	0.50	0.9%
	5	5	27,309	12,159	1,496	13,655	0.082	0.43	27,950	13,256	1,437	13,256	0.075	0.67	2.3%
	5	7	27,309	12,159	1,496	13,655	0.082	0.34	30,763	14,901	962	14,901	0.067	1.50	12.6%
	5	10	27,309	12,159	1,496	13,655	0.082	0.36	33,906	16,544	818	16,544	0.060	2.85	24.2%

Results truck capacity restriction (4 products)

Now we take the final comparison that is including the truck capacity by correcting the major setup cost. In table 17 we compared scenarios with a constant level $Worker = 5$ to fix the minor setup cost. The major setup cost and capacity of the trucks are changed. The method starts on the left side, we take the first example $Capacity = 5$ and $S_o = 50$ to explain the results.

By using that input we obtain that 50% k_j are equal to one, which means half of the products are ordered every period. With the optimal value of T we find the number of order is 43.3 with equation (1.2). We calculate the average number of trucks B_1 with equation (5.2), and then we have the total cost correcting by the number of trucks 18,756.

Now with the same k vector we select with solver a new T that maximizes the average filling percentage from the trucks. That new T gives a value of 35.2 orders and a corrected $B_2 = 5.8$. Now, as we remember the method does not change the value of the k vector, just the number of orders that defines the number of trucks. The new cost shows improvements in cost, with a filling capacity improved. The same procedure is repeated for the rest of the scenarios with similar results. We see that the results improve as the capacity is larger, given that the misused space in trucks becomes larger.

We observe that the difference between B_1 and B_2 is very small, the reason is even if we are changing the value of T , the formula of B is based on rounded numbers and jumps on average are difficult to make, however the decrease in ordering by increasing the truck filling is decreasing the cost.

There is not a regular relation between the major setup cost parameter and the savings produced by the method because the ordering will lead to an arbitrary setting for the filling of the trucks depending on the truck capacity. The method will only correct for the number of orders when there is enough space in the trucks for doing that.

Table 17. Results 4 products with truck capacity restriction. Changing truck capacity and major setup cost

Capacity	So	Goyal					Goyal adjusting for T with B					% Δ TRC
		% Fill Trucks	Avg trucks B_1	Orders	TRC S_o	TRC $B_1 S_o$	% $k = 1$	% Fill Trucks	Avg trucks B_2	Orders	TRC $B_2 S_o$	
5	50	84%	5.4	43.3	9,323	18,756	50%	97%	5.8	35.2	17,927	4.6%
	250	94%	8.9	23.8	15,429	62,138	50%	98%	9.1	22.0	60,262	3.1%
	500	96%	12.4	16.4	20,327	114,046	75%	98%	13.0	15.3	112,043	1.8%
	750	98%	14.5	13.8	24,078	163,708	75%	98%	14.5	13.8	163,659	0.0%
	1000	97%	16.5	12.2	27,309	215,774	75%	98%	16.5	12.1	214,279	0.7%
10	50	75%	3.0	43.3	9,323	13,655	50%	84%	3.4	34.7	13,637	0.1%
	250	91%	4.6	23.8	15,429	36,661	50%	98%	4.6	22.0	35,119	4.4%
	500	93%	6.4	16.4	20,327	64,720	75%	95%	6.4	16.1	63,757	1.5%
	750	95%	7.5	13.8	24,078	91,307	75%	95%	7.5	13.8	91,284	0.0%
	1000	95%	8.5	12.2	27,309	118,502	75%	95%	8.5	12.1	117,779	0.6%
20	50	57%	2.0	43.3	9,323	11,489	50%	71%	2.0	34.7	11,289	1.8%
	250	83%	2.6	23.8	15,429	24,771	50%	89%	2.6	22.0	24,119	2.7%
	500	89%	3.4	16.4	20,327	40,057	75%	91%	3.4	16.1	39,632	1.1%
	750	88%	4.0	13.8	24,078	55,107	75%	99%	4.0	12.3	51,962	6.1%
	1000	90%	4.5	12.2	27,309	69,866	75%	91%	4.5	12.1	69,529	0.5%
24	50	75%	1.4	43.3	9,323	10,092	50%	85%	1.4	36.7	10,222	-1.3%
	250	87%	2.0	23.8	15,429	21,374	50%	91%	2.0	22.8	21,132	1.1%
	500	82%	3.0	16.4	20,327	36,769	75%	98%	3.0	13.8	34,433	6.8%
	750	98%	3.0	13.8	24,078	44,764	75%	99%	3.0	13.7	44,621	0.3%
	1000	84%	4.0	12.2	27,309	63,786	75%	99%	4.0	10.3	58,512	9.0%

We observe that for every case the correction in B leads to a more efficient filling of the trucks which expresses in less orders that at the same time lowers the major and minor setup cost. We have to remember that the same vector k is used, then the difference here only comes from the number of orders. That is the reason why we do not have a change in holding cost across capacities. The formula for holding cost is $0.5T \sum_1^n D_i k_i$. Then the capacity of the trucks really does not change the holding costs because the orders

stays the same just the number of trucks changes when we have a different capacity. Now from the left side to the right we see an increase in holding cost and that is generated by the method that decreases orders and that leads to bigger orders.

Table 18. Breakdown of average total costs on table 17

Cap. Pallet	So	Goyal						Goyal adjusting for T with B				% Δ TRC
		TRC	Major setup cost	Minor setup cost	Holding cost	T^*	Average CPU time	TRC	Major setup cost	Minor setup cost	Holding cost	
5	50	18,756	11,599	2,495	4,662	0.023	0.89	17,927	10,163	2,028	5,737	4.6%
	250	62,138	52,653	1,770	7,715	0.042	0.56	60,262	50,286	1,638	8,338	3.1%
	500	114,046	101,941	1,942	10,163	0.061	0.39	112,043	99,300	1,805	10,938	1.8%
	750	163,708	149,973	1,696	12,039	0.073	0.36	163,659	149,920	1,696	12,043	0.0%
	1000	215,774	200,624	1,496	13,655	0.082	0.29	214,279	199,031	1,484	13,764	0.7%
10	50	13,655	6,498	2,495	4,662	0.023	0.89	13,637	5,814	1,996	5,827	0.1%
	250	36,661	27,176	1,770	7,715	0.042	0.56	35,119	25,143	1,638	8,338	4.4%
	500	64,720	52,615	1,942	10,163	0.061	0.39	63,757	51,467	1,900	10,390	1.5%
	750	91,307	77,572	1,696	12,039	0.073	0.36	91,284	77,545	1,696	12,043	0.0%
	1000	118,502	103,352	1,496	13,655	0.082	0.29	117,779	102,531	1,484	13,764	0.6%
20	50	11,489	4,332	2,495	4,662	0.023	0.89	11,289	3,466	1,996	5,827	1.8%
	250	24,771	15,286	1,770	7,715	0.042	0.56	24,119	14,143	1,638	8,338	2.7%
	500	40,057	27,951	1,942	10,163	0.061	0.39	39,632	27,342	1,900	10,390	1.1%
	750	55,107	41,372	1,696	12,039	0.073	0.36	51,962	36,975	1,516	13,471	6.1%
	1000	69,866	54,716	1,496	13,655	0.082	0.29	69,529	54,281	1,484	13,764	0.5%
24	50	10,092	2,935	2,495	4,662	0.023	0.89	10,222	2,602	2,112	5,508	-1.3%
	250	21,374	11,889	1,770	7,715	0.042	0.56	21,132	11,375	1,693	8,064	1.1%
	500	36,769	24,663	1,942	10,163	0.061	0.39	34,433	20,688	1,629	12,116	6.8%
	750	44,764	31,029	1,696	12,039	0.073	0.36	44,621	30,813	1,684	12,124	0.3%
	1000	63,786	48,636	1,496	13,655	0.082	0.29	58,512	41,083	1,263	16,165	9.0%

7.2 Simulation to represent company scenario (20 products)

Now we apply the same method to include truck capacity including 20 products from the company instead of 4 to approximate a real scenario. As we explained in section 6, the most demanded products were selected and then the products with low demand were aggregated in the 20th product with weighted parameters. We have the same total demand, however we take the first 19 most ordered product and the rest of the products are aggregated in one, which account for 16% of total demand.

The reason we take 20 products instead of 83 is that with 83 products the results show orders that are less than a pallet and that is a restriction from the company.

The problem could be solved also with Porras & Dekker method by applying a minimum order quantity for all 83 products. However that leads to a minimum $k_j > 1$ and then a correction factor has to be used. It was decided that given that 84% of the demand is in the first 19 products, the results would be reasonable using 20 product aggregating the rest of the products in one with weighted parameters. Later for an application the rest of the products can be handled individually. In appendix 2 the parameter used are shown and on section 6 the method to acquire the parameters is explained on further detail.

Now in table 19 we take the real company operation restrictions, with a truck capacity of 24 pallets and a level of 5 workers per operation. The only changing parameter will be the major setup cost. We see a saving of 6.36% of the average cost per year when the cost is 750 euros, which is a cost similar to the 610 euros the company is facing per truck.

As before we see some interesting savings but they are less than in the case for 4 products. In general we see the same trend as in the last example, an increase in the filling from the trucks, a decrease in the orders and there is no relation between the major setup cost and the percentage of saving.

Table 19. Results for 20 products with truck capacity restriction. Changing major cost

So	Goyal with input S_o						Goyal with input				% Δ TRC
	TRC S_o	TRC $B_1 S_o$	% Fill Trucks	Avg trucks B_1	Orders	% $k = 1$	TRC $B_2 S_o$	% Fill Trucks	Avg trucks B_2	Orders	
50	16,643	18,055	77%	2.1	25.3	50%	18,117	81%	2.2	22.9	-0.35%
250	20,740	29,558	78%	3.0	17.6	65%	29,320	90%	3.3	14.1	0.80%
500	24,716	40,938	89%	3.3	14.2	75%	40,765	90%	3.3	14.0	0.42%
750	28,033	55,853	84%	4.0	12.4	75%	52,515	96%	4.0	10.8	6.36%
1000	30,913	63,725	94%	4.0	10.9	90%	62,289	98%	4.0	10.4	2.31%

On the breakdown of cost we see that the holding cost increase because the orders are now getting bigger. The minor and major setup cost on the other hand are decreasing because of the less frequent ordering.

Table 20. Breakdown of average total cost on table 19

So	Goyal with input S_o						Goyal with input				% Δ TRC
	TRC	Major setup cost	Minor setup cost	Holding cost	t^*	Average CPU time	TRC	Major setup cost	Minor setup cost	Holding cost	
50	18,055	2,679	7,055	8,321	0.039	1.92	18,117	2,529	6,363	9,225	-0.35%
250	29,558	13,227	5,961	10,370	0.057	0.54	29,320	11,590	4,769	12,962	0.80%
500	40,938	23,319	5,261	12,358	0.07	0.47	40,765	23,068	5,204	12,492	0.42%
750	55,853	37,094	4,743	14,016	0.081	0.84	52,515	32,281	4,127	16,106	6.36%
1000	63,725	43,750	4,519	15,456	0.091	0.41	62,289	41,792	4,317	16,180	2.31%

7.3 Improving results with a non-cyclic policy

Now we propose a different method with a non-cyclic strategy to distribute the truck schedule and compare it with the scenarios on the last table to test the results. We choose the next parameters for the case of 20 products.

$$S_o = 750 \quad Workers = 5 \quad Capacity = 24$$

From those parameters we can see on table 21 the vector \mathbf{k} and T from Goyal algorithm expressed in a truck schedule, we can observe that there is an ordering of 4 trucks each order in a cycle, which leads to a high holding cost. Now the idea proposed is the truck schedule will be manipulated to have 1 truck each time. The most ordered products will have a lower holding cost because they will be ordered less. And the least ordered products will be adjusted to have more than one pallet, so that the schedule follows the operation restriction, and the total average cost will decrease by reducing minor setup cost.

Table 21. Distribution with all trucks in an order

Product	K	Orders	Orders in a cycle			Total ordered in cycle
			0	1	2	
1	1	18.8	18.8	18.8	18.8	56.3
2	1	6.4	6.4	6.4	6.4	19.2
3	1	5.8	5.8	5.8	5.8	17.5
4	1	5.0	5.0	5.0	5.0	15.0
5	1	4.2	4.2	4.2	4.2	12.6
6	1	4.0	4.0	4.0	4.0	11.9
7	1	3.4	3.4	3.4	3.4	10.2
8	1	3.1	3.1	3.1	3.1	9.2
9	1	2.7	2.7	2.7	2.7	8.2
10	1	2.5	2.5	2.5	2.5	7.5
11	2	3.1	3.1		3.1	6.1
12	1	1.5	1.5	1.5	1.5	4.4
13	1	1.4	1.4	1.4	1.4	4.1
14	1	1.3	1.3	1.3	1.3	3.9
15	1	1.1	1.1	1.1	1.1	3.2
16	2	1.8	1.8		1.8	3.6
17	2	1.6	1.6		1.6	3.2
18	2	1.6	1.6		1.6	3.2
19	2	1.6	1.6		1.6	3.2
20	1	12.8	12.8	12.8	12.8	38.3
Truck Used			4	4	4	12
Avrg%Fill			87.0%	76.9%	87.0%	

On table 22 we observe the setting for the same cycle. Now the first order in table 21 correspond to orders 0 to 3 from table 22, the second order in table 21 correspond to orders 4 to 7 and so on. The truck schedule now correspond to 1 truck per order. The new schedule is manipulated manually to fit in the trucks. This scenario was selected on purpose with a vector \mathbf{k} with values of 1 or 2. That makes much easier the manipulation because the cycle is small, however that manipulation becomes exponentially intricate as the lcm from the \mathbf{k} vector becomes larger.

Table 22. Non-cyclical distribution with 1 truck in an order

Prod.	Previous order	Divided by	New Orders	Orders in a cycle												Total ordered in cycle
				0	1	2	3	4	5	6	7	8	9	10	11	
1	18.8	4	4.7	4.7	4.7	4.7	4.7	4.7	4.7	4.7	4.7	4.7	4.7	4.7	4.7	56.4
2	6.4	4	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	19.2
3	5.8	4	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	17.4
4	5.0	4	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	15.0
5	4.2	4	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	12.6
6	4.0	4	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	12.0
7	3.4	3	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	10.2
8	3.1	3	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	9.3
9	2.7	2	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	8.1
10	2.5	2	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	7.5
11	3.1	3	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	6.2
12	1.5	1	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	4.5
13	1.4	1	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	1.4	4.2
14	1.3	1	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	3.9
15	1.1	1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	3.3
16	1.8	1	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	3.6
17	1.6	1	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	3.2
18	1.6	1	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	3.2
19	1.6	1	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	1.6	3.2
20	12.8	4	3.2	3.2	3.2	3.2	3.2	3.2	3.2	3.2	3.2	3.2	3.2	3.2	3.2	38.4
Truck Used				1	1	1	1	1	1	1	1	1	1	1	1	1
Average filling % of Trucks				90	83	88	87	86	79	70	74	90	83	88	87	

To compare the total cost the whole schedule was calculated to get the major, minor and holding cost. When we compare the results we see that the non-cyclical method proposed improves cost comparing to Goyal algorithm but not as much as the cyclical method proposed. The non-cyclical method however depends on the trade-off between the holding cost and minor setup cost, it could be that for smaller setup cost parameters the results could improve more. However the calculation for the rest of the scenarios gets much more complicated because lcm become greater. The point here is to show that cyclical methods can be improved by non-cyclical methods however the complexity associated with them usually makes really difficult scaling the problem for application.

Table 23. Comparison of total average cost between models

	Goyal algorithm	Goyal correcting Fill%Trucks	Goyal correcting with non-cyclic policy
Orders	12.4	10.8	49.6
Major setup cost	37,094	32,281	36,750
Minor setup cost	4,743	4,127	13,184
Holding setup cost	14,016	16,106	4,935
Total average cost	55,853	52,515	54,870

8. Conclusions, discussion and limitations

The problem observed on this work was the distribution system from a Mexican distribution company specialized in industrial and transport lubricants. The problem on the literature is defined as the joint replenishment problem, in which we have different products with a parameter setting in terms of demand, price and setup cost. The company faces major and minor setup cost for uploading the trucks and the products and a holding cost. Also there are some restrictions that the company has that are special to the operations of the company, like minimum order quantities and truck capacity limitations that we have to take in consideration.

Two models were used to compare the normal JRP model. On the literature on operation research was found first a model from Chopra & Meindl (2012) that increases efficiency by agglomerating the products with less demand around the most ordered product. On a different from Van Eijs (1993) based on an algorithm proposed in Goyal (1974), an iterative approach is used. The procedure first takes two bounds in which the optimal number of orders could be selected. From there a finite number of combinations has to be tested that includes the time that has to be waited to order trucks and the time that has to be waited to fit a product in a certain truck schedule. Several scenarios were run changing the major and minor setup and for all of them the Goyal modified algorithm delivered equal or better average cost results, however the improvements over Chopra & Meindl are very small, and the reason for that is that the formula for getting the optimal waiting time between orders is the same and the small improvements are achieved with the Goyal method because it uses an exhaustive method that tests more scenarios..

On an application related to the company, a paper from Porras & Dekker (2008) is followed that applies Van Eijs approach to include the case of minimum ordering quantities. In the case of the company the minimum ordering quantity is a pallet of product. The results as expected are an increase in cost in terms of both last approaches, however the results for just a pallet does not increase more than 3% the cost of the product. However the results for this paper just give orders that are bigger than a certain limit but it does not give results orders in closed multiples of the minimum order quantities, so that restriction has to be relaxed and does not reflect a larger increase in cost that would involve having closed orders that are only multiples of MOQ.

A model was introduced that includes the problem for the limited truck capacity. The idea is first to get the optimal times for ordering of trucks and products with the Goyal algorithm. The average filling percentage of the trucks per period is calculated according to the capacity. According to that distribution the number of orders is varied to maximize the filling percentage of the trucks. The problem however is not linear because it involves a function with rounded up numbers. Once the results were calculated, savings were generated with the method.

Finally a non-cyclical method is used to compare results with the previous method and the original cyclical truck capacity method showed a better result for the scenario compared. The non-cyclical methods offer several improvements to the cyclical counterparts, however scaling the method gets very complex very quickly and a more systematic approach is needed to include non-cyclical approaches.

On the limitations of the method also we can include the measurement of the minor cost. What was done is to try to measure the labor cost related to unpack and repack the pallets from the trucks, according to the type and size of product that will take more or less minutes and also a number of people will be required during the operation. This measurement however could be improved and maybe more characteristics of the operation should be included because the minor setup cost are very low relatively and that influences the results.

Reference list

- Andres, F. M., & Emmons, H. (1976). Note-On the Optimal Packaging Frequency of Products Jointly Replenished. *Management Science*, 22(10), 1165-1166.
- Chopra, S., & Meindl, P. (2012). *Supply chain management. Strategy, planning & operation*. Prentice Hall 5th Edition.
- Goyal, S. K. (1974). Determination of optimum packaging frequency of items jointly replenished. *Management Science*, 21(4), 436-443.
- Hoque, M. A. (2006). An optimal solution technique for the joint replenishment problem with storage and transport capacities and budget constraints. *European journal of operational research*, 175(2), 1033-1042.
- Johansen, S. G., & Melchior, P. (2003). Can-order policy for the periodic-review joint replenishment problem. *Journal of the Operational Research Society*, 54(3), 283-290.
- Khouja, M., & Goyal, S. (2008). A review of the joint replenishment problem literature: 1989–2005. *European Journal of Operational Research*, 186(1), 1-16.
- Minner, S., & Silver, E. A. (2005). Multi-product batch replenishment strategies under stochastic demand and a joint capacity constraint. *IIE Transactions*, 37(5), 469-479.
- Oudenes, L. (2015). Joint replenishment problem of lubricant ordering with truck limitations (Bachelor Thesis), Erasmus University Rotterdam, 26(1).
- Porras, E., & Dekker, R. (2008). An efficient optimal solution method for the joint replenishment problem with minimum order quantities. *European Journal of Operational Research*, 174(3), 1595-1615.
- Sindhuchao, S., Romeijn, H. E., Akçali, E., & Boondiskulchok, R. (2005). An integrated inventory-routing system for multi-item joint replenishment with limited vehicle capacity. *Journal of Global Optimization*, 32(1), 93-118.
- Van Eijs, M. J. G. (1993). A note on the joint replenishment problem under constant demand. *Journal of the Operational Research Society*, 185-191.
- Viswanathan, S. (1996). A new optimal algorithm for the joint replenishment problem. *Journal of the Operational Research Society*, 936-944.

Appendix

Appendix 1. Setting of product parameters 4 products

Product	Demand in pallets	Average Price p/pallet (€)	Minor setup price p/ pallet (€)
Drum	1,539	2,037	3.94
Pail	170	2,086	15.53
IBC	223	2,621	1.23
Rest	14	1,926	11.69

Appendix 2. Setting of product parameters 20 products

Product	Demand in pallets	Average Price p/pallet (€)	Minor setup price p/ pallet (€)
1	232	1,683	19.7
2	79	2,797	6.2
3	72	1,949	19.7
4	62	1,951	19.7
5	52	3,662	19.7
6	49	1,823	19.7
7	42	1,384	19.7
8	38	2,891	19.7
9	34	1,765	19.7
10	31	2,014	19.7
11	19	1,974	77.6
12	18	1,698	19.7
13	17	1,490	19.7
14	16	2,214	6.2
15	13	2,139	6.2
16	11	2,396	77.6
17	10	1,718	19.7
18	10	1,733	19.7
19	10	1,866	19.7
20	158	2,343	41.0