**2020 INFORMS O.R. & Analytics Student Team Competition – ENTRY FORM**

**Team Number: 3234**

# Executive Summary

Jelly Bean Manufacturing is a jelly bean production company with 5 facilities in the USA (Detroit, Omaha, Springfield, Columbus and Green Bay). These factories need to produce the jelly beans in bulk to fulfill the various orders until Halloween. Orders have different specifications for color, size, flavor and type of packaging, generating a variety of 4800 possible order combinations.

The team was formed with the objective of planning the entire supply chain of Jelly Bean Manufacturing, going through the raw material allocation steps in each of the factories, determining how much should be transported, in addition to the allocation, sequencing and sizing of the orders for each of the locations. Furthermore, the approach includes simulation of the production system of the factories, obtaining the rates of use of the machines and mapping of bottlenecks at each location, aiming to meet the demand at a lower global cost.

Initially, when faced with the problem, the team held several meetings in order to understand and debate the final statement of the problem, so that we could plan the steps that would involve solving this challenge. At these meetings, we carried out statistical, financial and logistical analyses of the data sets. At the end of this period, after analysis and brainstorming, we decided how to divide the problem and the methods that would be used.

Still aiming at a better understanding of the problem, the simulation of the factories with fictitious orders was executed as a way to verify how the available resources would behave and understand the system's operating conditions as well as bottlenecks. After the conclusion of the simulation, the focus of the work was to define the quantities of raw materials to be transported between the factories and to elaborate the heuristics for the construction and sequencing of internal work orders.

The group's first focus, the transportation model, was framed as a Mixed Integer Linear Programming problem, whose objective was, at first, to minimize transportation costs and, at the same time, ensure that we had enough RMI at the factories to meet the maximum possible demand. Later on the Model came to also strive towards minimizing excess production when possible. The transport limit of 500,000 pounds between two plants, the amount of free RMI storage at each facility and the number of empty drums at each location were restrictions imposed on the main transport model in order to meet the problem's conditions. The team concluded that a good alternative for solving this problem would be to use the GAMS language and the cloud-based NEOS solver.

The group's other focus was on building and sequencing internal work orders (hereinafter internal work orders will be considered as one customer order from the Order Bank being produced in a given facility. Macro Internal Work Orders will be used to represent internal work orders that share the same code and must be completed in order, as per stated in the Webinars and the Final Problem Statement). For this, two heuristics were developed by the team members.

Briefly, the first heuristic consists of using the result of the Master Production Schedule (MPS), in which it is known how much of each color will have in the RMI of each factory, to distribute customer orders among the factories. The second heuristic, carried out in conjunction with the simulation model, aimed to determine the sequencing of customer orders, in order to place the largest number of orders and not interrupt the factory due to the lack of stock drums. It was noticed the need to create extra orders to

empty the drums with leftover jelly beans that had already met their demand. In addition, attempts were made to order the same color, flavor and packaging in sequence, alternating between sizes.

Based on the results generated using the simulation with the heuristics and the PMP from the transportation problem, it was possible to visualize how much of the demand was met and the total cost, the main objectives of the entire modeling. In addition, it was possible to observe the behavior of resource utilization rates, the queues formed between the processes and the bottlenecks of the system. It was also possible to analyze the lost orders and the reasons that led to this.

Our recommendations to Jelly Bean Manufacturing are to increase the capacity for the Flavoring process, since that is the bottleneck at all facilities and consider substituting their small number of high-capacity RMI and PFI drums for a higher number of smaller capacity drums. Also, if possible, colors with the highest "complexity" value should have their upstream production process redesigned in order to bring their Classifier Splits in line with their historic demand percentages.

## Team Makeup & Process

The team was initially formed from an email in the graduation forum sent by the Professor, commenting on a challenge that involved topics related to operations research. At first, there were only 2 interested parties, and these were inviting other undergraduate students and friends whose analytical capacity and motivation to participate in the event was already known.

The team consists of 5 members, students of industrial engineering and also had the support of a Professor. The activities mentioned here, although allocated to certain members, had their logic built in group, with the participation of all. Members will be treated here with a number from 1 to 5 for presentation. The allocation of work to members occurred in an "organic" way, in which each member sought the tasks that seemed most attractive and challenging to them.

Team leader, Member 1 was responsible for coordinating and planning the team's work. This was done through the organization of meetings and conversations, in which it was sought to assign to each member a role that he felt more comfortable and able to perform, avoiding crunch. In addition, he had previously participated in study groups at the university in which Python programming was practiced, so he also worked in the preparation of the simulation and in the treatment of data and tables provided.

Member 2, skilled in business intelligence (BI) tools and Python programming due to his work experience in the area, took charge of simulating production for the 5 factories. Data processing and simulation were all developed in Python, using the open source libraries: Pandas, Numpy (Data), and Simpy (Simulation). This member also actively participated in the construction and integration of the aforementioned heuristics.

Member 3 was responsible for modeling and solving the transport optimization problem and the master production schedule, as well as conducting any Excel analyses that were needed. After the mathematical model was built, the GAMS software was used to obtain the solution to this problem. This member has previous experience with modeling, programming and data analysis.

Member 4 performed all the statistical analyses necessary for the distributions in the supplied Datasets, also making use of open Python libraries Pandas and SciPy. Her role was chosen due to her previous experience with statistical data analysis, programming and BI.

Member 5 worked on preparing the planning and allocation heuristics for work orders in the factories. He also assisted in the construction of the simulation model and in the codes programmed by the team. He has already worked with different programming languages, including Python.

The Professor provided all the support to the students during the process, answering questions and advising on the best decisions to be made. Special support was given in Optimization using Linear Programming models, part of his area of study. He has experience in the area of industrial engineering, with an emphasis on Operations Research in Logistics and Manufacturing, working mainly on the following topics: efficiency analysis, data envelopment analysis, productivity studies, manufacturing systems design, among others.

Additionally, four out of five members have previous experience in the areas of logistics and supply chain modeling.

Due to the experience and knowledge of the members and aiming to integrate the data processing with the simulation of the factories, these processes were all elaborated in Python, on the Jupyter platform of Anaconda Distribution, and made use of the open source libraries: Numpy, Pandas, Simpy, SciPy. Math, Matplotlib and Seaborn. All the knowledge necessary for preparing the codes was obtained from the documentation sites of these libraries and Stack Overflow. The only exception was the transport optimization stage, that was implemented using GAMS with the help of the Professor.

Regarding learning and challenges, the competition was able to provide innumerable lessons, by the interaction with people of different specialties, allowing the exchange of knowledge between the members of the group, as well as by the complexity of the simulation code and the heuristics, which required a lot of research and understanding in the Python and GAMS language. Among the main challenges encountered, it is worth mentioning the mathematical modeling of the optimization of raw material transport, given the number of variables and restrictions present in it, as well as the coding of the simulation, where many automated heuristics were used in order to improve factory efficiency.

Furthermore, the frequent meetings and presentation of analyses and results provided a rich brainstorming throughout the project that enriched the construction of the final solution. At a certain point, we allocated members to "work fronts" that would occur in parallel, but we always sought to form pairs or trios, so that no member would work alone. As demonstrated, it is possible to state that the whole result was the result of hard and dedicated teamwork

## Framing the Problem.

Considering the information provided, the group understood that the company's business problem could be framed as a production planning and control problem with a preliminary transportation problem. The main goal is to maximize the demand that is met before Halloween while also minimizing the overall cost of the operation. The company has five manufacturing facilities each with unique characteristics, such as number of machines, number and capacity of drums, processing rates and initial stock of raw material.

Based on this understanding, the group determined that the Jelly Bean Manufacturing problem should be divided into 2 macro problems: the master production plan, which determines how much should be transported between each factory, with an initial goal of seeking to meet the highest possible amount of the demand at the lowest possible cost.

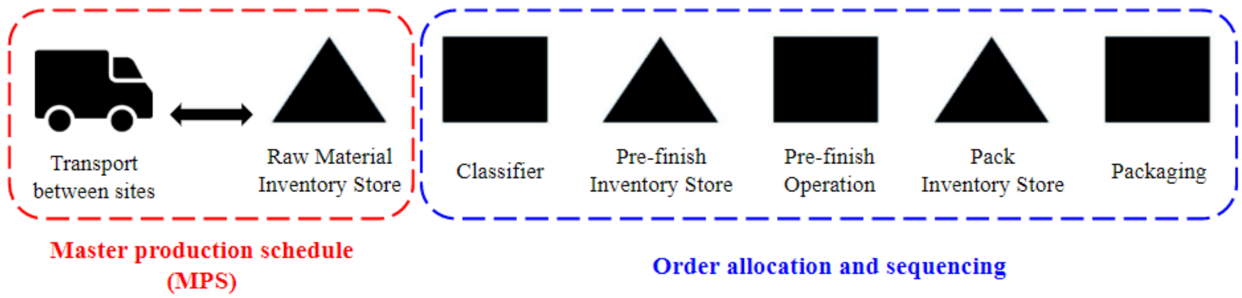


*Figure 1- Problem analysis division*

While constructing the Transportation Model it became apparent that the total aggregate demand for each color would be different from the necessary RMI processing amount due to their Classifier Split proportions not corresponding to the percentage of the demand each size represented. In order to find what the Minimum Necessary RMI Processing Amount (MNRPA) was for each color, the aggregate demand (sum of all orders in the Order Bank) for every combination of color and size was divided by their respective Classifier Split percentage. Consequently, since all colors had 5 values, from the 5 sizes of jelly beans, the MNRPA for each color became the highest value of this operation for that given color. This disparity between what the aggregate demand for any given color and its MNRPA was considered 'waste' or 'excess production'.

At another moment in time, the group realized that it was unfeasible to transport a significant portion of the initial RMI i.e. most of the initial RMI would not be transported because of the restrictions imposed on transportation amounts (a weight limit on routes between sites, the small number of empty RMI drums at each facility and the impossibility of mixing colors on drums already partially filled, severely restricting the transportation options). With that in mind, the transportation model's goal was altered to <u>meet the highest possible amount of the demand at the lowest possible cost whilst minimizing excess production.</u>

The minimization of excess production was inserted into the production cost of each facility through a corrective factor, having a unique value for each color being produced at each facility. This factor is the factory's average production cost (calculated from their given cost per 100 bags and 100 boxes) divided by the *effective production percentage* for that color, i.e. how much of the MNRPA is actually used to meet the demand. In this way, more 'wasteful' colors (lower *effective production percentage*) were de-prioritized by increasing their production costs while the Model was seeking to minimize overall cost.

Finally, the model dictates what colors must be moved where, with the stated goal of minimizing overall cost (having already taken into account the production cost correction) and, via a restriction, transporting as much of the material as possible to more efficient destinations.

Once the number of jelly beans of each color to be classified in each factory has been determined, the second macro problem starts, initiating with the distribution of customer orders from the Order Bank among the facilities. An order from a customer can be produced in any of the 5 plants, given that it respects the allocation made by the Transportation Model.

The group's first assumption was to distribute customer orders proportionally to what each site will classify for that color and size. It is evident that each color has at least one size that will not have any leftover from the classification and, therefore, the distribution of that size among the sites is critical. As the other sizes will have leftovers from the classification in all the sites, there is a margin of freedom in the proportion to distribute these orders among the sites, ensuring that demand is met.

By carrying out a brief inspection of the sites, it is possible to verify that each facility has different quantities of drums with different capacities. Thus, the second assumption of the team for the distribution of customer orders is that factories that have a smaller number of drums, but with a high storage capacity, should preferably receive orders with a higher quantity demanded. This assumption aims to make better use of the occupation of the drums. To perform this allocation, a heuristic was developed according to the assumptions made.

Once customer orders are allocated to each site, following the assumptions described above, the next step is the construction and sequencing of internal work orders. A consensus was reached such that there are no benefits in aggregating customer orders, as smaller orders are faster to manufacture and provide better machine utilization. Thus, it was initially chosen to use the customers' own orders as the initial internal work order. Customer's orders were split in different internal work orders in cases where the flavoring process would benefit. Any other splits were avoided since the gains did not outweigh the costs incurred by calculating the optimal split for each size and color and loss of traceability for the order.

Moving on to the analysis of the factories' production process, we can see that they all follow the same flow in general lines. There are three production steps: Classifier, Pre-Finish Operation and Packaging, as well as three inventories: Raw Material Inventory (RMI), Pre-Finish Inventory (PFI) and Pack Inventory (PI). The classification step separates each color into its component sizes, with input from the Raw Material Inventory and outputting to the Pre-Finish Inventory. In the pre-finish operation, the PFI drums are emptied in a tank where the one of 12 flavors are applied to the jelly beans, having a changeover time when changing the flavor present in the tank. In the packaging process, PI drums are emptied for packaging in either a bag or box, these two types of packaging having different rates.

Since jelly bean colors and sizes cannot be mixed in a given PFI drum and each classification is performed to meet a specific order of a specific size and color produces all 5 sizes, one of the great challenges is to prevent the production line from halting due to lack of free drums.

The team understood that every time an order of a certain color and size comes into the classifier, it is necessary to allocate the next orders in the sequence to produce the same color with a different size, as to make use of the other 4 sizes already classified.

Thus, when performing the sequencing of the internal work orders, a consensus was reached that initially it is more advantageous to execute all orders of a certain color, performing as necessary a rotation between the sizes to better take advantage of the occupation and use of the drums. The colors with the lowest average order quantity have the fastest orders to be produced and, thus, the team understood that they have a preference to start.

Because of the percentage split for each of the sizes, smaller split size orders will produce more jelly beans of the other sizes. Thus, an assumption that was adopted for sequencing is that for a given color, it is more advantageous to start with the smaller split size orders and, once the PFI Drums are filled with the other 4 sizes, send the orders of the other 4 sizes to consume the inventory produced as needed, starting with larger split size orders. This assumption was aimed at ensuring that production was not interrupted, as well as making the most of the drums that have already been classified, ensuring faster fulfillment of customer orders. To carry out this sequencing, a heuristic was also developed taking into account the assumptions made by the group.

In addition, when analyzing the next stage of the production process, the pre-finish operation, the group realized that during the rotation between the sizes of a certain color, one must first try to send orders of the same flavor to avoid performing setup in this operation. Moving on to the last stage of the process, the packaging, it was realized that orders should be sent with box type packaging first. This is due to the fact that this type of packaging has preference in the operation, making order queues with bag type packaging occur, if they arrive first.

When observing the whole process and the objectives of maximizing the demand served and obtaining the lowest total cost, the group decided to use as metrics and analyze the capacity utilization rate of each process, the queues between the processes, the orders lost due to lack of capacity in the stocks intermediaries, the time it takes to process an order from the beginning until it is delivered and its time at each stage of the process. These analyses were chosen in order to possibilitate an increase in capacity utilization efficiently to better meet demand and reduce excess production. Thus, optimization and heuristic methods were used to achieve the best possible result in view of the problem's constraints, having as its biggest challenge the mathematical and computational modeling of it.

## Data

The Data used to analyze and solve the problem was initially standardized. In the simulation environment, a previous treatment of the data was necessary so that the orders and resources could be read by the algorithm. In order for the data to relate to each other and the filters to work correctly, the names of the factories were all changed to follow the pattern "City, STATE", e.g. "Columbus, OH". Standardization corrections are also applied to the name of colors, drums and resources used.

After that, the data regarding the processing rates of packaging and pre-finish operation processes were statistically analyzed. For this purpose, hypothesis tests were initially applied to compare the averages of the processing rates of the pre-finish operation for the different sizes and flavors of jelly beans. Similarly, a comparison was made between the average processing rates of packaging for the different sizes of jelly beans and types of packaging.

The hypothesis test used was the bilateral Welch's t-test to compare means in samples, which does not assume equal population variance. The test was applied using Python programming SciPy library.

For the pre-finish operation, the test was initially applied comparing the processing rates for each pair of different flavors of jelly beans. This was done for each company site, in order to see if the processing of different flavors would influence the processing rate of this operation. Considering a confidence level of 99.95% and, thus, a significance level of 0.05% it is possible to affirm that the processing rates of the different flavors have equal means within each of the sites.

Thereafter, the same procedure was performed to analyze the influence of jelly beans sizes on processing rates. Considering the same confidence and significance level, the results also showed that the means of the processing rates are the same for the different sizes of jelly beans in each site.

For packaging, the analogous procedure was performed to analyze the influence of the different sizes of jelly beans and types of packaging on the processing rates of this operation. As a result, it was possible to observe that the sizes of jelly beans also do not incur significant difference in the means of the packaging processing rates, however the type of packaging does.

After these analyses and knowing that for the pre-finish operation the processing rates only change between sites and that the packaging processing rates also change only between sites and for the different types of packaging, it was necessary to analyze the behavior of the distributions of these processing rates, their means and their standard deviations.

At first, the graphs of the distribution of the pre-finish operation and box and bag packaging processing rates for each site were plotted. The resulting graphs are shown below, in figures 2, 3 and 4.

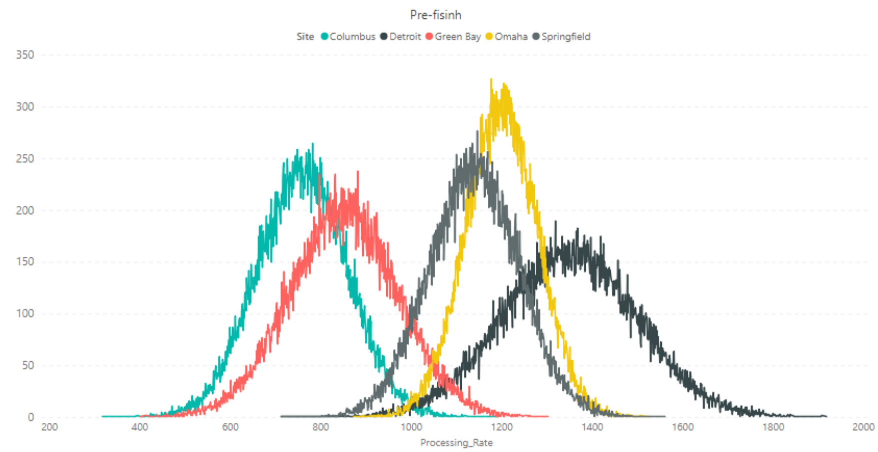


*Figure 2 - Distribution of pre-finish processing rates per site*

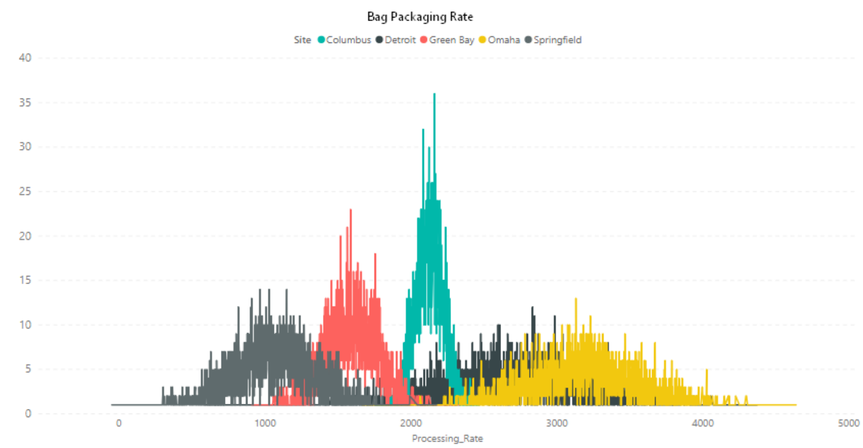


*Figure 3 - Distribution of bag packaging processing rates per site*

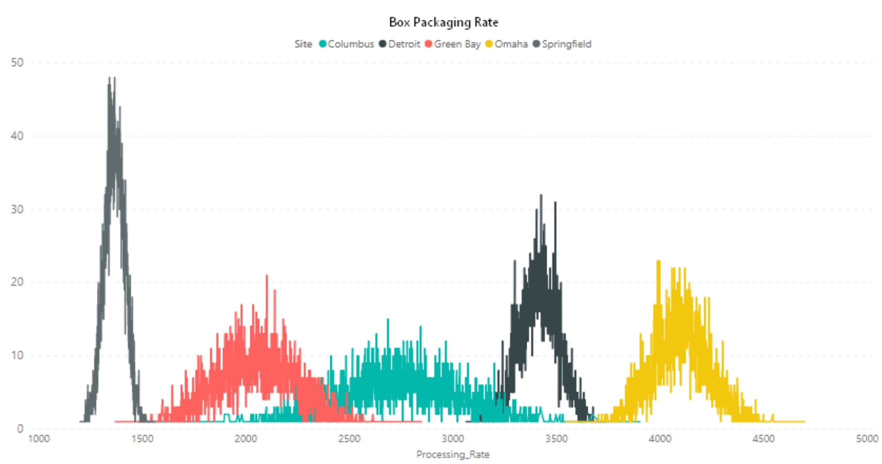


*Figure 4 – Distribution of box packaging processing rates per site*

By observing the plotted data, a similarity was noticed between the distributions presented and the normal distribution curve. To confirm this, it was necessary to perform a test of adhesion to the normal curve. To perform this analysis, normality test of the Python SciPy library was used, which tests the null hypothesis that a sample comes from a normal distribution. This test is based on the D'Agostino and Pearson tests that combines skew and kurtosis to produce an omnibus test of normality.

The test was applied to the distribution of pre-finish processing rates from all sites and packaging processing rates for bag and box at each site. Again, a confidence level of 99.95% was used .

For all distributions, the test did not reject the null hypothesis, which means that it can be concluded that they are distributions that follow the normal distribution curve. Additionally, the mean and standard deviation were calculated for each of the distributions analyzed i.e. for pre-finish operation at each site and for bag and box packaging for each site. The results can be seen in the tables below.

| Site | Mean | Standard Deviation |
|---|---|---|
| Detroit | 1349,80 | 150,08 |
| Columbus | 759,94 | 100,10 |
| Green Bay | 850,51 | 120,35 |
| Springfield | 1139,80 | 99,67 |
| Omaha | 1199,74 | 80,11 |

*Table 1 - Pre-finish operation processing rates per site (mean and standard deviation)*

| Site | Bag | | Box | |
|---|---|---|---|---|
| | Mean | Standard Deviation | Mean | Standard Deviation |
| Detroit | 2665,93 | 494,09 | 3416,83 | 99,65 |
| Columbus | 2127,12 | 100,69 | 2741,84 | 301,91 |
| Green Bay | 1589,79 | 197,11 | 2049,14 | 203,43 |
| Springfield | 1056,68 | 299,09 | 1366,61 | 50,57 |
| Omaha | 3184,70 | 398,81 | 4099,87 | 148,06 |

*Table 2 - Packaging processing rates per site and packaging type (mean and standard deviation)*

The results obtained from these analyses were essential for structuring the simulation model presented below. Through these results, it was possible to attribute normal distribution to the processing rates of the pre-finish operation and packaging, in addition to allowing us to state that these rates would remain constant when processing different sizes and flavors of jelly beans.

## Methodology Approach & Model Building

To solve the proposed problem, the team decided to use mostly Python programming, since the members already had experience in programming in that language. In addition to that, the large number of open source libraries makes it a very versatile tool to be used in the different analyses required in the problem. The use of Simio software for system simulation and IBM SPSS software for statistical analysis was considered, but the members presented only vague notions about the use of these softwares, so this would require a long time to learn them even before starting the modeling of the problem.

For the modeling of the transport optimization problem, it was chosen to use the GAMS (General Algebraic Modeling System) language, which is a modeling system for mathematical optimization designed for modeling and solving linear, nonlinear, and mixed-integer optimization problems. This choice was made due to the fact that it is a language in which some members also had knowledge and mainly because the professor used it a lot in his work.

The steps of the problem development and modeling methodology can be seen in the figure below.

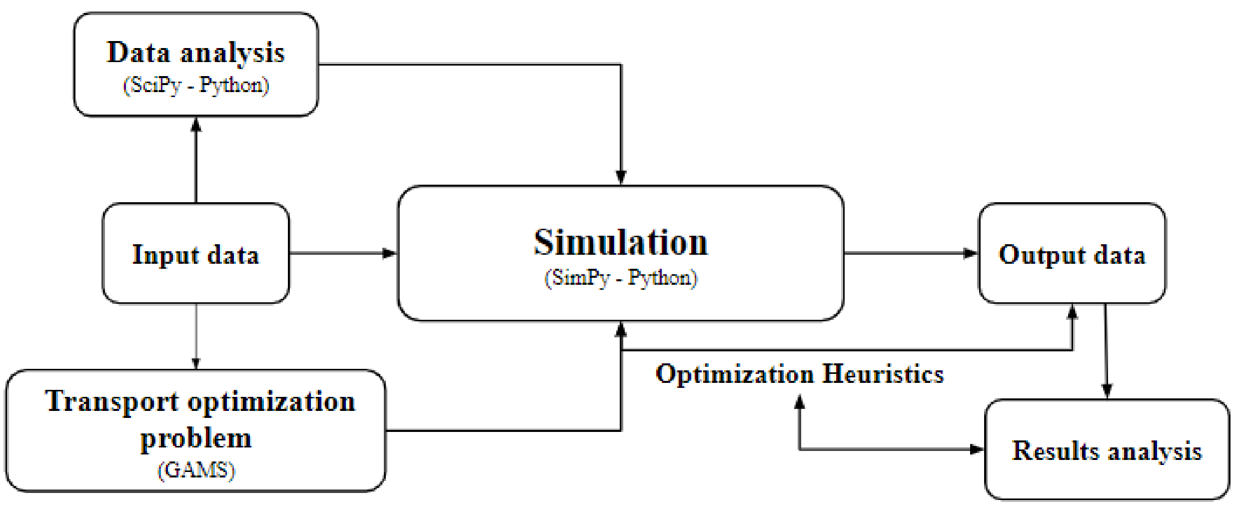


*Figure 5 - Methodology used to the problem resolution*

Initially, the input data was observed. Part of the information was used in the development of the transportation optimization problem, such as RMI inventory level, distance matrix and transportation threshold. Data on processing rates, on the other hand, went through the data analysis process using the Python SciPy library, as explained in the previous topic.

The result of the transport optimization problem, the MPS, served as input data for the simulation, as well as the data from the statistical analysis and the data provided by the problem statement.

The simulation and the development of heuristics occurred in parallel, since heuristics were designed to improve the efficiency of processes and the use of resources and reduce waste and, consequently, costs.

The details of the development of each stage of the methodology are further explained below.

**Internal Work Order Complexity Analysis**

Analyzing and comparing two Internal Work Orders, given the way in which the team had been using the term thus far (representing one specific combination of Color, Size, Flavor and Bagging Type produced in each factory), was infeasible and would carry very little significance, since there were 4800 SKUs comprised of small groups. Therefore, the team has decided to call the aggregate of all internal work orders of the same color one Macro Internal Work Order. This Macro Internal Work Order can be more easily analyzed and carries much more significant information.

Having said that, in order to be able to compare two macro internet work orders and determine which one of them is more complex, a standardized metric was needed. This raises the question: What metrics intrinsic to the macro internal work order could be taken to represent its innate complexity?

This can be answered by understanding what is *complexity*. The work orders that produced the most excess production, and, therefore, filled the system with unnecessary material, were considered more complex. These tended to be the ones whose different sizes presented the greatest disparity between Classifier Split Percentages and Relative Aggregate Demand (the size's aggregate demand divided by the whole color's aggregate demand).

Therefore, the correlation between those metrics was chosen as the metric. More specifically, the inverse of the absolute value of the correlation. The absolute value because correlations can be negative and the inverse because we wanted a metric that grew as the perceived complexity of the internal work order grew.

| color | correlation | **complexity** |
|---|---|---|
| 1 | -82.26% | **1.216** |
| 2 | 62.50% | **1.600** |
| 3 | -47.50% | **2.105** |
| 4 | -31.71% | **3.153** |
| 5 | -84.03% | **1.190** |
| 6 | -3.34% | **29.939** |
| 7 | 36.84% | **2.714** |
| 8 | -47.71% | **2.096** |
| 9 | 5.70% | **17.540** |
| 10 | 74.32% | **1.345** |
| 11 | 46.57% | **2.147** |
| 12 | 5.24% | **19.069** |
| 13 | -10.99% | **9.098** |
| 14 | 32.37% | **3.089** |
| 15 | 6.05% | **16.538** |
| 16 | 22.39% | **4.466** |
| 17 | 43.79% | **2.284** |
| 18 | -57.00% | **1.754** |

| | | |
|---|---|---|
| 19 | 72.18% | **1.386** |
| 20 | 83.72% | **1.195** |
| 21 | 44.48% | **2.248** |
| 22 | 45.82% | **2.183** |
| 23 | -66.28% | **1.509** |
| 24 | -93.59% | **1.068** |
| 25 | 53.46% | **1.871** |
| 26 | -5.70% | **17.535** |
| 27 | 45.36% | **2.205** |
| 28 | 81.31% | **1.230** |
| 29 | -25.68% | **3.895** |
| 30 | 22.34% | **4.477** |
| 31 | -37.64% | **2.657** |
| 32 | 67.68% | **1.478** |
| 33 | -71.61% | **1.397** |
| 34 | -66.07% | **1.514** |
| 35 | -22.94% | **4.359** |
| 36 | 6.97% | **14.349** |
| 37 | -23.68% | **4.223** |
| 38 | 63.19% | **1.582** |
| 39 | -59.13% | **1.691** |
| 40 | -64.58% | **1.549** |

*Table 3 – correlation between Classifier Split Percentages and Relative Aggregate Demand and complexity*

### Transport Model Formulation

The following model was formulated in order to determine quantities that have to be transported from one plant to another so that overall transport cost as well as estimated production cost at the destinations can be minimized by attending all given restrictions.

While the demand of RMI must be met by the stocks available in the cities, a set of restrictions must be observed. Following restrictions were considered:

1) Quantity of RMI available at origins;
2) Quantity of RMI demanded at destinations;
3) Transport capacity of the links between origins and destinations;
4) Number of empty drums available for storage at destinations;
5) RMI Storage capacity at destinations;
6) Minimal quantities of RMI required to meet overall demand at destinations.

### Decision variables

$X_{i,j,k}$ - Decision variable that defines the quantity of RMI of the color *k* to be transported from origin *i* to destination *j*, with *k=1, .., r*, *i=1, .., m* and *j=1, .., n*. The scalars *r*, *n*, and *m* indicate, respectively, the number of different RMI's, number of origins and number of destinations. In this case, *r=40 RMI's*, *n=m=5 cities* (Green Bay, Omaha, Springfield, Columbus, Detroit).

$D_{j,k}$ - Decision variable that defines the quantity demanded of RMI of the color $k$ at destination $j$, with $k=1, .., r$ and $j=1, .., n$ to attempt overall demand of jelly beans.

$Y_{j,k}$ - Decision integer variable that indicates the number of empty (at the beginning) drums to be used after transportation at destination $j$, with $j=1, .., n$; $k=1, .., r$.

**Objective function**

The main objective is to minimize total cost, i.e., the sum of total transportation cost with total production cost. Therefore, the objective function *f(x,d,y)* is given by:

$$f(x,d,y) = \sum_{k=1}^{r}\left(\sum_{j=1}^{n}\sum_{i=1}^{m} c_{i,j}X_{i,j,k}\right) + \sum_{j=1}^{n}\sum_{k=1}^{r} p_{j,k}D_{j,k}$$

with

$$c_{i,j} = \frac{3.5}{50{,}000}d_{i,j}$$

where

$X_{i,j,k}$ - Decision variable that defines the quantity of RMI of the color $k$ to be transported from origin $i$ to destination $j$, with $k=1, .., r$; $i=1, .., m$; $j=1, .., n$. The scalars $r$, $n$, and $m$ indicate, respectively, the number of different RMI's, number of origins and number of destinations. In this case, *r=40 RMI's*, *n=m=5 cities* (Green Bay, Omaha, Springfield, Columbus, Detroit).

$D_{j,k}$ - Decision variable that defines the quantity demanded of RMI of the color $k$ at destination $j$, with $k=1, .., r$; $j=1, .., n$ to attempt overall demand of jelly beans.

$d_{i,j}$ - Distance in miles between origin $i$ and destination $j$, with $i=1, .., m$; $j=1, .., n$ (parameter given in the Final Problem Statement and shown in Table 1).

$c_{i,j}$ - Cost of shipping one lb. of RMI from origin $i$ to destination $j$, with $i=1, .., m$; $j=1, .., n$ (Table 2).

$p_{j,k}$ - Unit (per pound) production cost at destination $j$ considering the waste necessary to meet the demand for all sizes of RMI of the color $k$, with $j=1, .., n$; $k=1, .., r$ (Table 3).

Parameter $p_{j,k}$ *is* obtained from the average production cost per pound for each facility (information given in the Final Problem Statement) divided by the effective demand percentage for each color $k$. The effective demand percentage is obtained by dividing the Minimum Necessary RMI level for each color $k$ ($mr_k$) by the Aggregate Demand for each color $k$ (the sum of all orders of the color $k$). In this manner, the model will prioritize colors that generate less waste and factories where production is cheapest.

| d(i,j) | Green Bay | Omaha | Springfield | Columbus | Detroit |
|---|---|---|---|---|---|
| Green Bay | - | 570 | 720 | 530 | 510 |
| Omaha | 570 | - | 630 | 780 | 730 |
| Springfield | 720 | 630 | - | 630 | 750 |
| Columbus | 530 | 780 | 630 | - | 210 |
| Detroit | 510 | 730 | 750 | 210 | - |

*Table 4 - Distance (d_(i,j)) in miles between origins and destinations*

| c(i,j) | Green Bay | Omaha | Springfield | Columbus | Detroit |
|---|---|---|---|---|---|
| Green Bay | - | 0.0399 | 0.0504 | 0.0371 | 0.0357 |
| Omaha | 0.0399 | - | 0.0441 | 0.0546 | 0.0511 |
| Springfield | 0.0504 | 0.0441 | - | 0.0441 | 0.0525 |
| Columbus | 0.0371 | 0.0546 | 0.0441 | - | 0.0147 |
| Detroit | 0.0357 | 0.0511 | 0.0525 | 0.0147 | - |

*T Table 5-Cost of shipping one lb. of RM, (c_(i,j)), from origin i to destination j*

| p(j,k) | Green Bay | Omaha | Springfield | Columbus | Detroit |
|---|---|---|---|---|---|
| | j=1 | j=2 | j=3 | j=4 | j=5 |
| k=1 | 1.99 | 1.95 | 2.24 | 2.14 | 2.05 |
| k=2 | 1.85 | 1.82 | 2.09 | 2.00 | 1.91 |
| k=3 | 1.65 | 1.62 | 1.86 | 1.78 | 1.70 |
| k=4 | 1.78 | 1.74 | 2.00 | 1.92 | 1.83 |
| k=5 | 2.13 | 2.09 | 2.40 | 2.30 | 2.19 |
| k=6 | 2.18 | 2.14 | 2.46 | 2.35 | 2.24 |
| k=7 | 1.83 | 1.80 | 2.06 | 1.97 | 1.88 |
| k=8 | 1.95 | 1.91 | 2.20 | 2.10 | 2.01 |
| k=9 | 1.62 | 1.59 | 1.83 | 1.75 | 1.67 |
| k=10 | 1.73 | 1.70 | 1.95 | 1.86 | 1.78 |
| k=11 | 1.97 | 1.94 | 2.22 | 2.13 | 2.03 |
| k=12 | 1.74 | 1.70 | 1.96 | 1.87 | 1.79 |
| k=13 | 1.53 | 1.50 | 1.73 | 1.65 | 1.58 |
| k=14 | 2.05 | 2.01 | 2.31 | 2.21 | 2.11 |
| k=15 | 1.86 | 1.82 | 2.09 | 2.00 | 1.91 |
| k=16 | 1.48 | 1.45 | 1.67 | 1.60 | 1.52 |
| k=17 | 2.04 | 2.00 | 2.30 | 2.20 | 2.10 |
| k=18 | 1.61 | 1.58 | 1.81 | 1.73 | 1.65 |
| k=19 | 1.26 | 1.24 | 1.42 | 1.36 | 1.30 |
| k=20 | 1.47 | 1.45 | 1.66 | 1.59 | 1.52 |
| k=21 | 1.29 | 1.27 | 1.45 | 1.39 | 1.33 |
| k=22 | 1.86 | 1.83 | 2.10 | 2.00 | 1.92 |
| k=23 | 1.77 | 1.74 | 1.99 | 1.90 | 1.82 |
| k=24 | 1.81 | 1.78 | 2.05 | 1.96 | 1.87 |
| k=25 | 1.36 | 1.33 | 1.53 | 1.46 | 1.40 |
| k=26 | 1.89 | 1.85 | 2.13 | 2.03 | 1.94 |
| k=27 | 1.88 | 1.85 | 2.12 | 2.03 | 1.94 |
| k=28 | 1.27 | 1.25 | 1.43 | 1.37 | 1.31 |
| k=29 | 2.37 | 2.33 | 2.67 | 2.55 | 2.44 |
| k=30 | 1.99 | 1.95 | 2.24 | 2.14 | 2.05 |
| k=31 | 2.20 | 2.16 | 2.48 | 2.37 | 2.27 |
| k=32 | 1.67 | 1.64 | 1.89 | 1.81 | 1.73 |
| k=33 | 1.76 | 1.73 | 1.98 | 1.90 | 1.81 |
| k=34 | 2.60 | 2.56 | 2.94 | 2.81 | 2.68 |
| k=35 | 1.94 | 1.91 | 2.19 | 2.09 | 2.00 |
| k=36 | 1.41 | 1.38 | 1.59 | 1.52 | 1.45 |
| k=37 | 1.44 | 1.42 | 1.63 | 1.56 | 1.49 |
| k=38 | 1.70 | 1.67 | 1.92 | 1.84 | 1.76 |
| k=39 | 1.66 | 1.63 | 1.87 | 1.79 | 1.71 |
| k=40 | 1.98 | 1.94 | 2.23 | 2.13 | 2.04 |

*Table 6 – Parameter p_(j,k) for each color k and destination j*

## Restrictions formulation

1) <u>Stock level of RMI at origins</u>:

$$\sum_{j=1}^{n} X_{i,j,k} \leq o_{i,k} \qquad for\ i = 1, .., m; k = 1, .., r$$

where

$X_{i,j,k}$ - Decision variable that defines the quantity of RMI of the color $k$ to be transported from origin $i$ to destination $j$, with $k=1, .., r$; $i=1, .., m$; $j=1, .., n$. The scalars $r$, $n$, and $m$ indicate, respectively, the number of different RMI's, number of origins and number of destinations. In this case, *r=40 RMI's*, *n=m=5 cities* (Green Bay, Omaha, Springfield, Columbus, Detroit)

$o_{i,k}$ - Stock level of RMI of the color $k$ at origin $i$, with $k=1, .., r$; $i=1, .., m$ (parameter given in the Final Problem Statement and shown Table 4).

| o(i,k) | Green Bay | Omaha | Springfield | Columbus | Detroit |
|---|---|---|---|---|---|
| k=1 | 0 | 880,000 | 207,683 | 0 | 0 |
| k=2 | 0 | 872,752 | 176,000 | 0 | 0 |
| k=3 | 81,562 | 0 | 88,000 | 320,000 | 300,000 |
| k=4 | 169,257 | 0 | 88,000 | 320,000 | 300,000 |
| k=5 | 0 | 880,000 | 243,209 | 0 | 0 |
| k=6 | 0 | 258,952 | 88,000 | 320,000 | 300,000 |
| k=7 | 204,049 | 0 | 88,000 | 320,000 | 300,000 |
| k=8 | 0 | 248,810 | 88,000 | 320,000 | 300,000 |
| k=9 | 141,440 | 0 | 88,000 | 320,000 | 300,000 |
| k=10 | 214,789 | 0 | 88,000 | 320,000 | 300,000 |
| k=11 | 0 | 440,000 | 88,000 | 0 | 472,695 |
| k=12 | 24,433 | 0 | 88,000 | 320,000 | 300,000 |
| k=13 | 152,076 | 0 | 88,000 | 320,000 | 300,000 |
| k=14 | 0 | 251,600 | 88,000 | 320,000 | 300,000 |
| k=15 | 0 | 262,868 | 88,000 | 320,000 | 300,000 |
| k=16 | 0 | 0 | 49,124 | 320,000 | 300,000 |
| k=17 | 0 | 440,000 | 88,000 | 0 | 511,742 |
| k=18 | 41,365 | 0 | 88,000 | 320,000 | 300,000 |
| k=19 | 0 | 0 | 0 | 308,989 | 300,000 |
| k=20 | 99,100 | 0 | 88,000 | 320,000 | 300,000 |
| k=21 | 0 | 0 | 0 | 266,874 | 300,000 |
| k=22 | 168,526 | 0 | 88,000 | 320,000 | 300,000 |
| k=23 | 173,020 | 0 | 88,000 | 320,000 | 300,000 |
| k=24 | 178,972 | 0 | 88,000 | 320,000 | 300,000 |
| k=25 | 5,657 | 0 | 88,000 | 320,000 | 300,000 |
| k=26 | 0 | 880,000 | 177,246 | 0 | 0 |
| k=27 | 0 | 440,000 | 88,000 | 0 | 500,138 |
| k=28 | 0 | 0 | 54,244 | 320,000 | 300,000 |
| k=29 | 0 | 1,028,736 | 264,000 | 0 | 0 |
| k=30 | 0 | 440,000 | 88,000 | 0 | 491,858 |
| k=31 | 0 | 919,465 | 264,000 | 0 | 0 |
| k=32 | 155,593 | 0 | 88,000 | 320,000 | 300,000 |
| k=33 | 183,058 | 0 | 88,000 | 320,000 | 300,000 |
| k=34 | 0 | 1,310,969 | 0 | 0 | 0 |
| k=35 | 0 | 245,127 | 88,000 | 320,000 | 300,000 |
| k=36 | 38,382 | 0 | 88,000 | 320,000 | 300,000 |
| k=37 | 102,187 | 0 | 88,000 | 320,000 | 300,000 |
| k=38 | 120,106 | 0 | 88,000 | 320,000 | 300,000 |
| k=39 | 106,537 | 0 | 88,000 | 320,000 | 300,000 |
| k=40 | 0 | 440,000 | 88,000 | 0 | 479,132 |

*Table 7- Stock level (o_(i,k)) of RMI of each color k and origin i*

2) <u>Quantity of RMI demanded at destinations</u>:

$$\sum_{i=1}^{m} X_{i,j,k} - D_{j,k} \geq 0 \qquad for\ j = 1,..,n; k = 1,..,r$$

where

$X_{i,j,k}$ - Decision variable already declared in restriction (1)

$D_{j,k}$ - Decision variable that defines the quantity demanded of RMI of the color k at destination $j$, with $k$=1, .., $r$; $j$=1, .., $n$ to attempt overall demand of jelly beans.

3) <u>Transport capacity of the links between origins and destinations</u>:

$$\sum_{k=1}^{r} X_{i,j,k} \leq tc_{i,j} \qquad for\ i = 1,..,m; j = 1,..,n$$

where

$X_{i,j,k}$ - Decision variable already declared in restriction (1).

$tc_{i,j}$ - Transport capacity of the link between origin $i$ and destination $j$, with $i$=1, .., $m$; $j$=1, .., $n$ (parameter given in the Final Problem Statement and shown in Table 5).

| tc(i,j) | Green Bay | Omaha | Springfield | Columbus | Detroit |
|---|---|---|---|---|---|
| Green Bay | 0 | 500,000 | 500,000 | 500,000 | 500,000 |
| Omaha | 500,000 | 0 | 500,000 | 500,000 | 500,000 |
| Springfield | 500,000 | 500,000 | 0 | 500,000 | 500,000 |
| Columbus | 500,000 | 500,000 | 500,000 | 0 | 500,000 |
| Detroit | 500,000 | 500,000 | 500,000 | 500,000 | 0 |

*Table 8Transport capacity of the link between origin i and destination j*

4) <u>Number of empty drums for storage at destinations</u>:

$$0 \leq \sum_{k=1}^{r} Y_{j,k} \leq mz_j \qquad for\ j = 1,..,n$$

where

$Y_{j,k}$ - Decision integer variable that indicates the number of empty (at the beginning) drums to be used after transportation at destination $j$, with $j$=1, .., $n$; $k$=1, .., $r$.

$mz_j$ - Maximum number of empty drums at destination $j$, with $j$=1, .., $n$ (parameter given in the Final Problem Statement and shown in Table 6).

| mz(j) | Green Bay | Omaha | Springfield | Columbus | Detroit |
|---|---|---|---|---|---|
| | 1 | 3 | 2 | 2 | 2 |

*Table 9-Maximum number of empty drums at destinations*

5) Storage capacity of RMI at destinations:

$$\sum_{i=1}^{m} X_{i,j,k} - dc_j Y_{j,k} \leq fc_{j,k} \quad for\ j = 1, .., n; k = 1, .., r$$

where

$X_{i,j,k}$ - Decision variable already declared in restriction (1).

$fc_{j,k}$ - Free capacity for storage of RMI of the color *k* available in drums already partially full at destination *j*, with *k=1, .., r*; *j=1, .., n* (parameter given in the Final Problem Statement and shown in Table 7).

$dc_j$ - Empty drum capacity available for storage of RMI at destination *j*, with *j=1, .., n* (parameter given in the Final Problem Statement and shown in Table 8).

$Y_{j,k}$ - Decision variable already declared in restriction (4).

| fc(j,k) | Green Bay | Omaha | Springfield | Columbus | Detroit |
|---|---|---|---|---|---|
| 1 | 0 | 0 | 56,315 | 0 | 0 |
| 2 | 0 | 7,246 | 0 | 0 | 0 |
| 3 | 248,436 | 0 | 0 | 0 | 0 |
| 4 | 160,742 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 20,789 | 0 | 0 |
| 6 | 0 | 181,046 | 0 | 0 | 0 |
| 7 | 125,950 | 0 | 0 | 0 | 0 |
| 8 | 0 | 191,187 | 0 | 0 | 0 |
| 9 | 188,559 | 0 | 0 | 0 | 0 |
| 10 | 115,209 | 0 | 0 | 0 | 0 |
| 11 | 0 | 0 | 0 | 0 | 127,302 |
| 12 | 305,565 | 0 | 0 | 0 | 0 |
| 13 | 177,921 | 0 | 0 | 0 | 0 |
| 14 | 0 | 188,399 | 0 | 0 | 0 |
| 15 | 0 | 177,130 | 0 | 0 | 0 |
| 16 | 0 | 0 | 38,875 | 0 | 0 |
| 17 | 0 | 0 | 0 | 0 | 88,255 |
| 18 | 288,633 | 0 | 0 | 0 | 0 |
| 19 | 0 | 0 | 0 | 119 | 0 |
| 20 | 230,899 | 0 | 0 | 0 | 0 |
| 21 | 0 | 0 | 0 | 53,125 | 0 |
| 22 | 161,471 | 0 | 0 | 0 | 0 |
| 23 | 156,977 | 0 | 0 | 0 | 0 |
| 24 | 151,026 | 0 | 0 | 0 | 0 |
| 25 | 324,340 | 0 | 0 | 0 | 0 |
| 26 | 0 | 0 | 86,752 | 0 | 0 |
| 27 | 0 | 0 | 0 | 0 | 99,860 |
| 28 | 0 | 0 | 33,753 | 0 | 0 |
| 29 | 0 | 291,261 | 0 | 0 | 0 |
| 30 | 0 | 0 | 0 | 0 | 108,139 |
| 31 | 0 | 400,532 | 0 | 0 | 0 |
| 32 | 174,404 | 0 | 0 | 0 | 0 |
| 33 | 146,939 | 0 | 0 | 0 | 0 |
| 34 | 0 | 9,030 | 0 | 0 | 0 |
| 35 | 0 | 194,871 | 0 | 0 | 0 |
| 36 | 291,616 | 0 | 0 | 0 | 0 |
| 37 | 227,810 | 0 | 0 | 0 | 0 |
| 38 | 209,891 | 0 | 0 | 0 | 0 |
| 39 | 223,460 | 0 | 0 | 0 | 0 |
| 40 | 0 | 0 | 0 | 0 | 120,865 |

*Table 10- Free storage capacity of RMI for color k available in drums already partially full at destination j*

| dc(j) | Green Bay | Omaha | Springfield | Columbus | Detroit |
|---|---|---|---|---|---|
| | 330,000 | 440,000 | 88,000 | 320,000 | 300,000 |

*Table 11 - Drum capacity available for storage of RMI at destination j*

6) <u>Minimal quantities of RMI to attend overall demand at destinations</u>:

$$\sum_{j=1}^{n} D_{j,k} \geq mr_k * mrp \qquad for\ k = 1,..,r$$

with

$$mrp = \frac{1}{10.44}$$

where

$D_{j,k}$ - Decision variable already declared in restriction (1).

$mr_k$ - Minimum amount of RMI of the color *k* needed to meet overall demand, with *k=1, .., r*.

$mrp$ - Factor that must multiply $mr_k$ in order to achieve a feasible solution. Represents the percentage of total $mr_k$ that will be moved between sites. Its value has been found through a sensitivity analysis.

Parameter $mr_k$ is the least amount of RMI (in pounds) of color *k* which must be processed in order to meet the demand for every size (S1, S2, S3, S4 and S5) of any given RMI of the color *k*. The parameter was calculated for each color *k* by dividing each size's aggregate demand (sum of all orders of the same color and size) by their Classifier Split percentage and picking the highest number for each color.

In other words:

Let *l* represent every size of jelly bean, *l=1 .. s*. The scalars *s* indicates the number of colors. In this case, *s* = 5.

Let $ad_{k,l}$ represent the aggregate demand (sum of all orders of the same color and size) of jellybeans of the same color *k* and size *l*

Let $cs_{k,l}$ represent the "Classifier Split" percentage for jellybeans of each color *k* and size *l*.

Therefore, $mr_k = \text{MAX}\left({}^{ad_{k,l}}/_{cs_{k,l}}\right)$, *l = 1 .. s.*

In such a manner, for *k=1*, $mr_1 = \text{MAX}\left(\frac{115{,}652.5}{0.12}; \frac{95{,}260}{0.29}; \frac{140{,}017}{0.13}; \frac{99{,}765.5}{0.19}; \frac{87{,}890}{0.27}\right)$, which would be $\frac{140{,}017.}{0.13} =$ 1,077,053.85. This is done in order to assure that every size of that color will have their demand met. Table 9 gives the evaluated parameters for all colors.

| k | mr(k) |
|---|---|
| 1 | 1,077,053.85 |
| 2 | 1,038,585.00 |
| 3 | 781,853.33 |
| 4 | 868,740.91 |
| 5 | 1,112,250.00 |
| 6 | 957,620.00 |
| 7 | 903,155.00 |
| 8 | 947,480.00 |
| 9 | 841,169.23 |
| 10 | 913,800.00 |
| 11 | 990,955.00 |
| 12 | 725,345.83 |
| 13 | 851,622.73 |
| 14 | 950,254.55 |
| 15 | 961,382.14 |
| 16 | 662,558.33 |
| 17 | 1,029,555.00 |
| 18 | 742,050.00 |
| 19 | 603,041.67 |
| 20 | 799,230.00 |
| 21 | 561,321.88 |
| 22 | 868,035.00 |
| 23 | 872,412.50 |
| 24 | 878,337.50 |
| 25 | 706,694.44 |
| 26 | 1,046,940.00 |
| 27 | 1,018,045.00 |
| 28 | 667,657.14 |
| 29 | 1,280,165.00 |
| 30 | 1,009,918.18 |
| 31 | 1,171,910.00 |
| 32 | 855,136.36 |
| 33 | 882,369.23 |
| 34 | 1,298,150.00 |
| 35 | 943,862.50 |
| 36 | 739,097.06 |
| 37 | 802,287.50 |
| 38 | 820,030.00 |
| 39 | 806,683.33 |
| 40 | 997,390.91 |

*Table 12 - Minimum amount of RMI of the color k needed in order to fully meet the demand for all s colors of k*

**Model scalability**

The whole Transportation model was developed with scalability in mind. Every index is defined from 1 to another scalar (be it *m, n, r* or *s*) which can be easily modified at the start of the GAMS file. The model also proves itself as very efficient in its current form given the very small Compilation, Generation and Execution times, at 0.002 seconds, 0.011 and 0.011 seconds, respectively.

This mathematical programming model comprises a total of $m\times r + 2n\times r + m\times n + n + r = m(r+n) + r(3n) + r$ restrictions, $m\times n\times r + n\times r$ continuous variables and $n\times r$ integer variables. Given that the dimensions of the problem, n=m=5 plants and r=40 colors, the proposed model has 865 restrictions, 1,200 continuous variables and 200 integer variables. The search for a solution using GAMS and Gurobi solver available at the NEOS cloud optimization server did not require expressive machine resources, being possible to obtain solutions in a few seconds. It is, therefore, a relatively simple model to be optimized with the computational resources and algorithms existing today and, therefore, still far from a massive optimization model which would need more sophisticated computational resources to obtain good solutions. This observation leads to the conclusion that the scalability of the model can be considerably increased both in terms of the number of factories and in the greater diversity of raw material (or colors) in the case of the given problem.

**Data Processing & Sequencing Heuristics**

For the development of Python programming code for data processing and implementation of heuristics, the Jupyter programming environment, from Anaconda Distribution, was used. Thus, all libraries to be used are imported, which include: Pandas, Numpy, Simpy, Math, Seaborn and Matplotlib.

Then, all data import and processing takes place, including the CSV data provided by the sponsor, as well as tables of the final problem statement and the resulting PMP file, using file reading and Data Frames (table) editing functions from the data science library, Pandas. The tables had columns added, were unified in the corresponding names and values, among other treatments, but that occurred only in the programming environment, without affecting the original files.

Before elucidating the heuristics themselves, it is worth mentioning the fact that they were divided into two stages: one responsible for the dimensioning of internal work orders in the factories, and another that deals with the sequencing of orders, done in a way that would allow any order that was sent into production did not cause the production to halt due to lack of available resources.

The first heuristic, that of dimensioning, starts from the table obtained from the MPS, by which we know how much of each color is available in each factory's RMI, and the demand for each size of each color. Then, the calculation of the required amount of jelly beans to be removed from the RMI drums begins, since the raw material in the classifier will undergo splits to then produce the quantity demanded.

In this calculation, for each color of the jelly beans, the demand for all orders for each of the 5 sizes is added. This value of the demand for size is divided by the percentage of split that the classifier will generate, obtaining 5 values that are equivalent to the minimum amount of raw material to be removed from the RMI drums of a certain color, which will allow to meet the demand for each size.

To ensure that after the split it is still possible to meet all the demand for the 5 sizes, the largest amount of raw material generated must be chosen. This is done in an automated way for all colors and all factories. This is the same as the MNRPA discussed above.

Then, for each size, order bank orders are sorted from largest to smallest. This starts the separation of orders between sites that have RMI drums of that color, as defined in the MPS.

The first factory that will produce any given macro internal work order (the aggregate of internal work orders - i.e. order bank orders - by color) is the one with the largest amount of PFI storage (that is the number of drums times their individual capacity). There, the individual order bank orders are allocated to that factory in regards to their size in a decreasing manner. The final order bank order may be split due to the lack of RMI in that factory.

For the other sizes, orders were allocated in a similar way, aiming to maintain the proportion of orders for each size being produced in each factory, since the leftovers and costs involved in their production were considered when calculating the RMI of each location in the MPS. The process is repeated for other factories that also have RMI of that color. The process takes place until all colors are allocated to a factory.

At this point, each factory has a list of customer orders without sequencing. Thus, this second heuristic is also aimed at prioritizing these orders as to guarantee better utilization of the machinery, to reduce lost sales and ensure higher utilization of the PFI capacity.

First, orders were divided by color, where the color with the lowest average order size would be the first color to start and end the classification process. Then, orders were prioritized according to their classifier split percentage. Orders with the smallest percentage were the first and those with the largest split were the last. The reasoning behind this will be further elaborated. Finally, flavors that represented the lowest amount of orders were prioritized in order more quickly vent production. As a rule, boxes were prioritized over bags for orders of the same size, color and flavor.

The strategy employed to produce the optimal sequencing consisted of three simultaneous sequencing algorithms. The first, the “Mother Sequencing”, puts to production orders that will generate the most excess, i.e. the colors with the lowest split percentage of that color. This will fill up the PFI drums and at this moment, the second sequencing, “Trash-cleaning Sequencing” enters into action. This sequencing algorithm takes over the process after classification, attempting to vent as much PFI as possible, prioritizing orders with the highest split percentage that were already classified.

The third sequencing algorithm, “Dead Production Sequencing”, enters into action whenever the plant is forced to halt production due to the lack of PFI storage. That is, all PFI drums are loaded with colors that no longer have any customer orders to fulfill. Therefore, that inventory is simply ‘discarded’ (stored in bags after being flavored with whatever the last flavor in the Pre-finish Operation was) and the system can resume production.

The final sequencing, which is also the objective of the heuristics, is then obtained from the actual order in which production orders were executed at the factory. This final sequencing is then passed over to the *actual* simulation, where all the production information is gathered.

## Simulation Methodology

In order to facilitate the team’s understanding of the production process, visualization methods were employed, such as the process mapping via flowcharts, from classification, passing through flavoring and ending at packaging. Moreover, a pseudocode for each of the processes was elaborated in the early stages of the simulation’s construction

As previously stated, the open-source SimPy Python library was employed for the simulation, given its many advantages, which include great versatility and flexibility; real-time visualization of productive processes; ease of dealing with different data types present in the simulation, such as lists, dictionaries, DataFrames, Series, CSV files and Excel files; its great scalability; and ubiquity of open-source and cloud-based solutions that could run the code, such as Jupyter and Collab notebooks.

Another great side effect of using Jupyter to conduct the simulations was the easier integration with the datasets generated by the previous parts of the process. The simulation consists of 3 basic SimPy processes. They are: Classifier, Flavoring and Packaging.

In the Classifier process, we define as ‘container’ variables the RMI and PFI drums, so that they may be filled and emptied dynamically as the simulation runs its course. The classifier machine is defined as a ‘resource’, which is used to represent any activity that consumes time. Also, the time consumed can be randomly generated, following any given distribution. The mean and standard deviation for the time each step of the process requires, calculated in the “Data” section, were employed here.

The other processes had a similar definition.

In the next moment, there is the creation of auxiliary PFI, FI and PI Dataframes, which store the current amount of stock at each moment of the simulation in each container. In this way, anytime you want to stop the simulation, you can check the current status of the inventories at the factory. Also, this is how we can observe the evolution of the level in each inventory over time.

Regarding internal work orders, the algorithm recognizes which manufacturing step each order is in, which allowed the real-time monitoring of production

Since the simulation adopts the final result of heuristics as an initial predictive solution, the production occurs without any halts or major problems, since those were already taken care of by the heuristics

## Next Steps

For future modifications of the simulation, it is suggested to evaluate through artificial intelligence techniques (Machine Learning) how to break each of the customers' orders in order to reduce the average internal work order size and consequently positively adjust the processing times.

## Analytics Solution and Results

Performance Summary:

In total, 3,446,755 lbs of RMI were transported between sites, incurring in a total transportation cost of US$ 150,979.91. We believe this was not caused by a single constraint but by the interaction of all the restrictions (the RMI drum capacity available at the receiving end, the 500,000lb limitation on the transfer of goods per route as well as the 1 color per drum rule).

This is backed by the fact that only 6 of the possible 20 links between cities came to a utilization of over 90% (i.e. over 450,000lbs transported on that route) as well as the observation that the average utilization of free RMI Capacity stayed well under 40%, at a 32% average for all cities, Detroit being the furthest above the average, with an average utilization of RMI capacity of almost 60% and Springfield the furthest below, having only 14% of its available capacity filled.

This has brought the team to the conclusion that the limiting factor of transportation was the lack of empty drums at the receiving end. The best illustration of this comes in the form of the Green Bay facility, whose inbound routes are almost all maxed out, all empty drums are filled and yet the plant's available RMI inventory utilization sits at less than 40%.

| transp amount ↓to/from→ | Green Bay, WI | Omaha, NE | Springfield, MO | Columbus, OH | Detroit, MI |
|---|---|---|---|---|---|
| Green Bay, WI | 0 | 124,340 | 499,813 | 500,000 | 500,000 |
| Omaha, NE | 0 | 0 | 500,000 | 0 | 461,585 |
| Springfield, MO | 0 | 10,361 | 0 | 46,754 | 0 |
| Columbus, OH | 0 | 0 | 0 | 0 | 138,424 |
| Detroit, MI | 0 | 499,998 | 165,476 | 0 | 0 |

*Table 13 - Total transport between sites*

| Route Utilization | Green Bay, WI | Omaha, NE | Springfield, MO | Columbus, OH | Detroit, MI |
|---|---|---|---|---|---|
| Green Bay, WI | 0% | 25% | 100% | 100% | 100% |
| Omaha, NE | 0% | 0% | 100% | 0% | 92% |
| Springfield, MO | 0% | 2% | 0% | 9% | 0% |
| Columbus, OH | 0% | 0% | 0% | 0% | 28% |
| Detroit, MI | 0% | 100% | 33% | 0% | 0% |

*Table 14 - transportation routes utilization*

| Cities: | Green Bay, WI | Omaha, NE | Springfield, MO | Columbus, OH | Detroit, MI |
|---|---|---|---|---|---|
| Empty drum capacity | 330,000 | 1,320,000 | 176,000 | 640,000 | 600,000 |
| Partially filled drums | 3,909,848 | 1,640,702 | 236,484 | 64,134 | 544,421 |
| Total Capacity | 4,239,848 | 2,960,702 | 412,484 | 704,134 | 1,144,421 |
| Amount received | 1,624,153 | 961,585 | 57,115 | 138,424 | 665,474 |
| Utilization | 38% | 32% | 14% | 20% | 58% |

*Table 15 - Available RMI inventory utilization at destinations*

| transp cost ↓to/from→ | Green Bay, WI | Omaha, NE | Springfield, MO | Columbus, OH | Detroit, MI |
|---|---|---|---|---|---|
| Green Bay, WI | $ - | $ 19,950.00 | $ 25,200.00 | $ 18,550.00 | $ 17,850.00 |
| Omaha, NE | $ - | $ - | $ 22,050.00 | $ - | $ 25,550.00 |
| Springfield, MO | $ - | $ 22,050.00 | $ - | $ 22,050.00 | $ - |
| Columbus, OH | $ - | $ - | $ - | $ - | $ 7,350.00 |
| Detroit, MI | $ - | $ 25,550.00 | $ 26,250.00 | $ - | $ - |

*Table 16 - Transportation costs:*

| Manufacturing Site | Total Production (lbs) | Total number of days to complete production | Total Production Cost ($) |
|---|---|---|---|
| Green Bay | 3,984,262.21 | 108.40 | $3,962,747.20 |
| Omaha | 10,566,167.79 | 157.99 | $10,323,145.93 |
| Springfield | 2,879,328.43 | 109.81 | $3,231,182.36 |
| Columbus | 8,487,533.66 | 212.06 | $9,103,728.61 |
| Detroit | 10,401,351.00 | 189.83 | $10,847,525.54 |

*Table 17 - Production amounts, days, and costs*

| Manufacturing Site | Classifier (Utilization) | Pre-finish Operation (Utilization) | Pack Operation (Utilization) | Bottleneck (Operation name) |
|---|---|---|---|---|
| Green Bay. | 74.68% | 98.45% | 93.94% | Pre-finish Operation |
| Omaha | 62.66% | 97.75% | 86.07% | Pre-finish Operation |
| Springfield | 86.60% | 97.04% | 54.95% | Pre-finish Operation |
| Columbus | 69.02% | 92.09% | 70.31% | Pre-finish Operation |
| Detroit | 66.74% | 99.48% | 85.8% | Pre-finish Operation |

*Table 18 - a Bottlenecks at each site*

- More detailed results:

| Location | Production until deadline (lbs) | Lost sales (lbs) | Days after deadline |
|---|---|---|---|
| Green Bay | 3,984,262.21 | 0.0 | 0 |
| Omaha | 10,566,167.79 | 0.0 | 0 |
| Springfield | 2,879,328.43 | 0.0 | 0 |
| Columbus | 7,274,787.00 | 1,212,746.66 | 29.69 |
| Detroit | 10,065,105.00 | 336,246.00 | 7.46 |

*Table 19 - analysis of lost sales per color per site*

The table for Inventory at each site by color prior to start of operations on April 1st after shipping is annexed to this document as a .csv file (Team3234_InventoryAtSite.csv)
The table for the inventory balance from the start to Packaging operation by site is annexed to this document as a .csv file. (Team3234_InventoryBalance.csv)

- Additional Results:

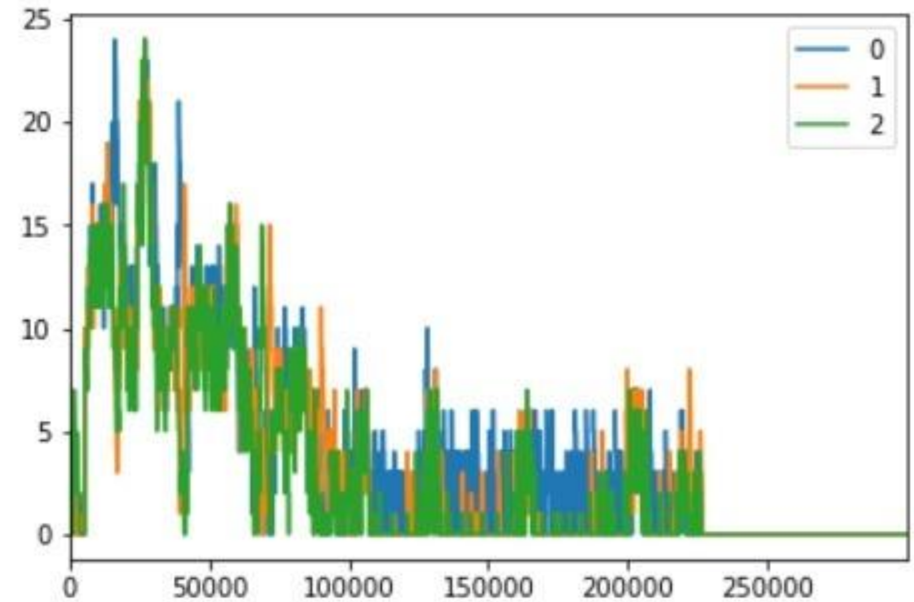


*Figure 6 - Omaha line for the classifiers (no. of workorders vs. time)*

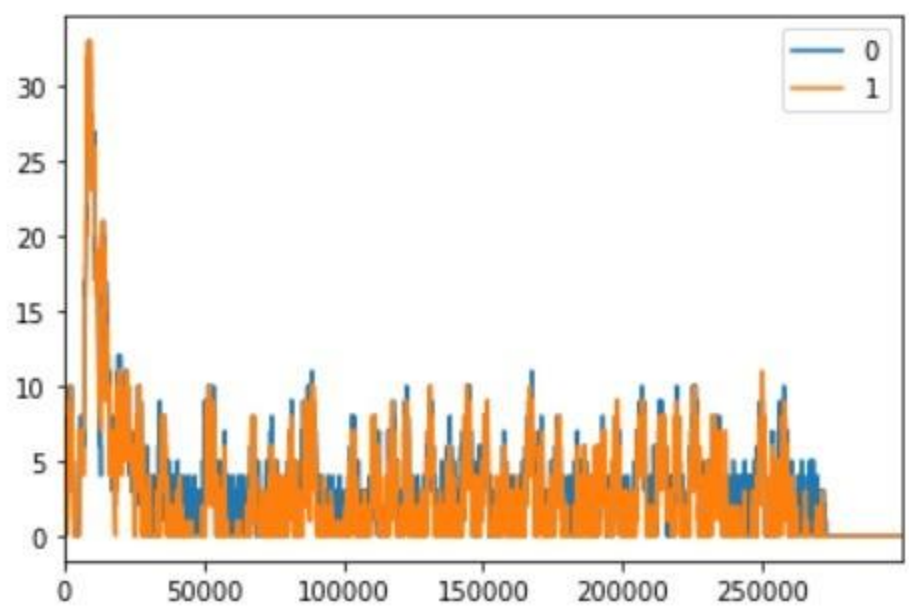


*Figure 7 - – Detroit line for the classifiers (no. of workorders vs. time)*

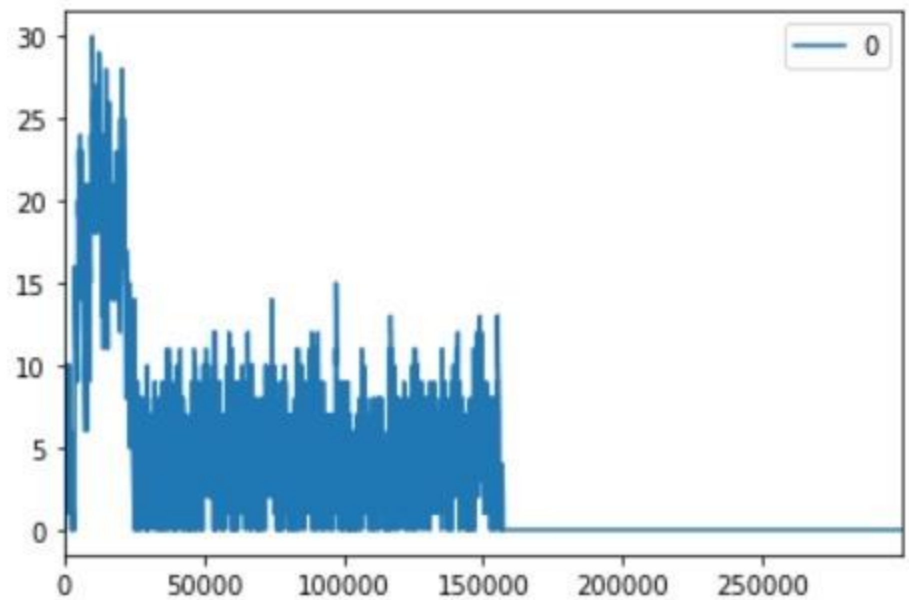

*Figure 8 - Springfield line for the classifiers (no. of workorders vs. time)*

- Insights for JB Manufacturing Executive Team:

The flavoring operation should have its efficiency increased by process optimization or the expansion of the number of equipments. The RMI inventory should be rethinked in order to better utilize the capacity i.e. more drums of a lower capacity should be prioritized over less drums of bigger capacity. Moreover, more empty drums at each facility would also greatly increase the system's efficiency as a whole, since there are plants that can cope with their full demand in less than 70% of the available time and others that cannot produce everything in time.

If possible, colors with the highest "complexity" value should have their production process redesigned in order to bring their Classifier Splits in line with their historic demand percentages.

## References


- Hillier FS, Liebermann GJ (1995) *Introduction To Operations Research*, 2nd. international revised ed. (McGraw-Hill Professional, New York)
- The SciPy community (2019) Statistics (scipy.stats). Retrieved December 20, 2019, scipy.org.
- Stack Overflow (2019) Questions. Retrieved December 18, 2019, stackoverflow.com.
- NumPy (2019) NumPy 1.17 Manual. Retrieved December 02, 2019 ,numpy.org.
- SimPy (2019) SymPy 1.5.1 documentation. Retrieved November 05, 2019 ,simpy.org.
- Pandas (2019) Pandas 0.25.3 documentation. Retrieved December 15, 2019, pandas.pydata.org