24th COBEM - 2017

24[th] ABCM International Congress of Mechanical Engineering
December 3-8, 2017, Curitiba, PR, Brazil

COBEM-2017-1748
A MULTI-CRITERIA BASED APPROACH FOR THE PRODUCTION DISTRIBUTION IN THE POULTRY INDUSTRY

Wellington Rodrigo Monteiro
Gilberto Reynoso-Meza
Industrial and Systems Engineering Graduate Program (PPGEPS),
Pontifical Catholic University of Parana (PUCPR),
Imaculada Conceicao, 1155, Zip code 80215-901, Curitiba, PR, Brazil
wellington.monteiro@pucpr.br, g.reynosomeza@pucpr.br

Abstract. *The use of multi-objective approaches to solve problems in the industry grew in the last years. Nevertheless, these strategies are not mature enough in many fields. One example is in the poultry industry with its particularly complex chain. In this paper, we will discuss a multi-objective approach tested in a real case in the production distribution of several parts of Griller chicken for multiple plants during one month.*

Keywords: *Multi-objective, Optimization, Multicriteria, poultry, Griller.*

1. INTRODUCTION

The supply chain in the industry is, by definition, a complex set of operations and resources that must be extremely optimized in order to achieve its maximum potential which does include the management of upstream and downstream relationships in order to achieve an outcome which is more profitable to all the parties in the chain (Christopher, 2016). One of its parts is the Production Plan, which defines what should one or more plants build considering a myriad of variables - i.e. market demand, production line capacity, logistics and stock limits, suppliers constraints, raw material limits, etc. Therefore, an accurate Production Plan is a key component to maximize the potential profits.

In the real world scenarios, however, the comparison between the proposed production plans and the production real output are commonly different. For example, due to both internal (worker strikes, human fatigue, machine unscheduled maintenance, lack of raw material, damaged goods) or external (issues with the suppliers, weather), the resulting production plan may suffer unexpected impacts. In order to keep track of these differences, KPIs (*Key Performance Indicators*) may be implemented where one or more factors are taken into account to rank one or more production lines or plants.

In the poultry industry, the challenges are considerably greater. Since it is a livestock type of material, the supply chain must be at the same time very large and very tight (Flanders and Gillespie, 2015). Large since the livestock production involves genetics, feeding, breeding and growth control up to the chicken grandparents and tight since during the whole process there are very strict sanitary controls with the ration, water, diseases, effluents, vaccination and temperature, for example. Furthermore, due to the demand from various markets and to minimize the waste, almost all the parts of the chicken must be industrialized.

Considering these challenges, the production planners need to create production plans for a given product line and distribute it accordingly along with its plants. Since the variables are prone to changes over the time (depending, for example, on the market seasonal demands and the plant availability), the production plans are usually changed periodically.

In order to automate the creation of such plans, many industries use software algorithms where proposed solutions are generated based on these ever-changing variables. In the poultry industry, these algorithms are exposed to a larger degree of complexity since the number of materials to be produced are higher than the number of materials in the other industries summed with the higher volatility of the market.

However, many algorithms used by these industries will attempt to optimize a single objective - the profits. Of course this poses as an additional problem because, as exposed earlier, the estimated profits are not true since they are *theoretical* profits - the real world situations would then attenuate these profits. Since this consideration is not taken in account by simple, single-objective algorithms, it is likely that there are better alternatives with slightly lower theoretical profits but, at the same time, with higher reliability. From these alternatives the solution designer should then be able to do the choices accordingly.

Considering this situation, the proposal is to convert this case, currently used as a single-objective problem by a large poultry processing company, into a multi-objective problem (MOP) - that is, a problem that instead of having one single objective (in this case, the profitability), would have multiple objectives (in this scenario, both the profitability and the reliability). Then, the results will be evaluated and compared against each other.

This document is divided in five sections, as follows: the first and current section is the introduction. The second section explains the background of the case being tested as well as an explanation of how does the Multi-Objective Optimization (MOO) works. The third section, on the other hand, shows the proposal to modify and test the presented case into a MOP. Then, the following section specifically shows the technical details of the created MOP as well as its results after its optimization. The fifth section presents the conclusions and the future work to be done from this document.

2. BACKGROUND

Currently, some single objective optimization solutions used in the industry are usually specialized algorithms built from the scratch with the profitability in mind such as Otimix™. While it is easier to use and faster to run from a computing standpoint, it cannot be easily customized for another objective different from the profitability. Moreover, due to its nature, industry solutions have a lesser degree of control over the algorithm customization and parameter tuning. Since such algorithms are targeted towards only one objective, the problem designer usually has only one solution as the result of the minimization/maximization, eliminating the possibility to analyze the tradeoffs between different production plans considering two or more objectives.

Generally, as shown in (Rao, 2009), a single-objective problem is defined as:

$$Find\, \mathbf{X} = \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ ... \\ x_n \end{Bmatrix} which\, minimizes\; f(\mathbf{X}) = \mathbf{Y} \tag{1}$$

In this example, $x_i, i = 1, 2, ..., n$ are the *design variables* of a problem, where each variable is a characteristic of the problem. This single-objective problem is subject to a series of constraints which are a list of equations and/or inequations the objective function must comply. These constraints define limits for the algorithm in a way that only reasonable solutions can be achieved. As also noted by (Rao, 2009), these constraints are defined as:

$$g_j(\mathbf{X}) \leq 0, j = 1, 2, ..., m \tag{2}$$

$$l_j(\mathbf{X}) = 0, j = 1, 2, ..., p \tag{3}$$

where m and p can be different than n and/or themselves.

Considering that not all the plants have the same degree of reliability, an additional attention point of the production planners is to consider this when preparing the plans and attempt to, empirically, forecast any shortcomings from these plans without any aid from the algorithm. The issue becomes worse because such algorithms are normally used to generate the weekly production plans for a span of four weeks and for a myriad of plants and materials.

$$\begin{aligned} &\textit{Minimize} \Big\{ f(\mathbf{X}) = f_1(\mathbf{X}), f_2(\mathbf{X}), ..., f_n(\mathbf{X}), \\ &\textit{Subject to:} \begin{cases} h_i(\mathbf{X}) = 0, i = 1, 2, ..., I, \\ g_j(\mathbf{X}) = 0, j = 1, 2, ..., J, \\ \mathbf{X}_k^u \geq \mathbf{X}_k \geq \mathbf{X}_k^l, k = 1, 2, ..., K \end{cases} \end{aligned} \tag{4}$$

For this kind of problem, an alternative would be using a Multi-objective Problem (MOP) instead of a single-objective strategy one. The MOP, as seen in (Huang *et al.*, 2006) and shown in the Eq. (4), essentially supports more than one objective where *all* of them must be improved to the maximum. For this reason, the complexity may exponentially increase since more restrictions might be required as well as the quantity of variables involve increase. Furthermore, the determination of what makes a given solution better than the other may also be complex on itself - on this specific part there are the concepts of Pareto optimality, Pareto set and Pareto front. As explained by (Van Veldhuizen and Lamont, 1998), a given solution (i.e. vector) v is said to be *Pareto dominated* by another solution u if u is partially less than v such as:

$$\forall i \in \{1, ..., p\}, u_i \leq v_i \wedge \exists i \in \{1, ..., p\} : u_i < v_i \tag{5}$$

If a solution is non-dominated when compared against all the other solutions it is said that it is a *Pareto optimal* solution. A set of these solutions is part of the *Pareto front*. Yet, an important detail should be noted: it is possible that a given set of non-dominated solutions generated by an algorithm are not exactly Pareto optimal because the real Pareto front was simply not found by the algorithm. Then, as mentioned in (Reynoso-Meza *et al.*, 2014), these solutions are part of what is called the *approximated Pareto front* instead.

In the scenario presented by this document, the problem would have two objectives - the profitability and the reliability. Using the concepts explained above it is reasonable to suppose that the solution with the best possible reliability might sacrifice the profitability, and vice-versa. From these two extremes a series of more balanced solutions might be found along the Pareto front generated.

The strategy used to successfully process MOPs usually encompass additional steps as shown by (Coello, 2006) - after the definition of the MOP, an Evolutionary Multi-objective Optimization (EMO) algorithm would attempt to find the best results possible by considering and processing all the variables and restrictions (examples: Fonseca *et al.* (1993), Miettinen (1999), Marler and Arora (2010), Coello (2006)). Finally, the Multi-criteria Decision Making (MCDM) uses visual and/or mathematical methods (examples: (Behzadian *et al.*, 2010), Blasco *et al.* (2008)) to organize all the solutions found in the previous step. The design of this process is also known as the Multi-objective Optimization Design (MOOD) Reynoso-Meza *et al.* (2014). For all of the aforementioned steps there are differing variations and implementations. These variants have tradeoffs which makes one solution or another more useful depending on the size and type of the problem. The choice of the best implementation depends not only on the comparison between these tradeoffs, but also on the performance.

3. PROPOSAL

As mentioned in the section 2, this article proposes to implement a MOP to generate the production plans in a line of products in the poultry industry - more specifically, the Griller chicken production line. The development of the MOOD would use a set of scrambled data for information security purposes. This set, however, is the same as used for the single-objective algorithms.

The proposed MOP should have two objectives - reliability and profitability. Therefore, it is expected to have a Pareto front generated with the best solutions found considering the tradeoffs between both objectives, and shown graphically for comparison purposes.

Since the MOOD development usually includes as well the determination of the anchor points - the best possible values for each of the objectives, the comparison with the single-objective approach is done by comparing with the profitability anchor point.

The intention with the MOOD is to understand if there is a possible inverse relationship between profitability and reliability. If the theory is correct, the current single-objective strategies attempt to maximize the profitability while not necessarily having the maximum possible reliability. Since the reliability is a percentage of how much a given plant historically covers the production plans, the weighted profitability Wr is determined in Eq. (6), where it is determined by the sum of the products of the reliability R of the plant p by the profitability P found for the same plant p for all the four weeks and all the plants. This weighted profitability should show that it is less than the value returned for the profitability by any solution. Therefore, proposed solutions with higher reliability rates are expected to have less differences between the theoretical and the weighted profitabilities even though both of them should be lesser than the ones found by the solutions geared towards the profitability in mind.

$$\mathbf{Wr} = \sum_{i=1}^{4} R_i p * P_i p \tag{6}$$

4. TEST

4.1 Multi-objective Problem

The definition of the MOP attempted to follow all the rules and values described by the team in charge of the Production Plan creation. These rules basically were the following:

1. There are nine plants;
2. All of them must be used. Also, all of the plants have a *reliability* rate assigned to them according to historical data. 6 plants were rated as 55.6% while the remaining 3 were rated as 77.8%;
3. A minimum quantity must be assigned for production for 124 materials according to the market demand. This quantity should be distributed along four weeks;

4. Each material belongs to one of the 11 available material groups and each material has a different value for sale;
5. There are varying production availabilities depending on the material group, plant and week. In other words, a given plant may offer varying maximum production capabilities for a given material group depending on the week. Also, one plant can hold the production of more than one material group which, in turn, may distribute the production along one or more materials within a given material group;
6. If a plant is able to produce stock from a material group it does not necessarily mean that all of the materials under that group are allowed for that plant. Therefore, a list of allowed materials per plant was provided as well.

Since in the provided data the sum of the monthly supply and demand were equal, naturally there is less freedom to explore the possible solutions - i.e. if the algorithm cannot take advantage of the most profitable materials in the expense of the lesser ones and manufacture more of them, the maximum it can do is to try to push the production of the most profitable materials to the most reliable plants.

Based on all the aforementioned information, the MOP had the following characteristics:

1. 2 objectives: maximize the profitability and maximize the reliability;
2. 2032 variables - 508 per week. The variables are the production assigned for each material, for each plant.
3. 488 constraints - 124 of them are the market demands and all the others are the plant capacities for each week.

The constraints are defined in two different ways: the first one, as shown in the Eq. (7), refers to the weekly market demands. The sum of the production of a given material m in a week w across all the plants p should be greater than or equal the market demand D_{mw} for that material in that week. In other words, the production must cover the market needs - there are 124 constraints covering all the scenarios.

$$g_j(\mathbf{x_m w}) = \sum_{p=1}^{9} x_{mpw} \geq D_{mw} \tag{7}$$

In the same sense, all the plants have a given maximum production capacity. This capacity refers to the *weekly* production capacity and must be respected - i.e. the plant must not have a weekly output greater than its own capacity. For this reason, another set of 364 constraints are defined as shown in the Eq. (8): all the materials manufactured in the plant p for a given week w must be less than or equal the capacity C_{wp}.

$$g_j(\mathbf{x_w p}) = \sum_{m=1}^{124} x_{mpw} \leq C_{wp} \tag{8}$$

Since not all the plants are able to manufacture for all the materials and since not all the materials are required in all weeks, there are 2032 x_{mpw} variables.

In the first objective - reliability - the *plant's* reliability is assigned to each variable - therefore all of the 2032 x_{mpw} variables sharing the same plant will have the same value assigned to it. On the second objective - profitability - the sales price of each material is assigned instead for each variable.

4.2 Evolutionary Multi-objective Optimization

The first strategy used in the Evolutionary Multi-objective Optimization step (EMO) was the implementation of the MATLAB®'s own Multi-objective Genetic Algorithm (*gamultiobj*). Since this algorithm works only with the $Ax \leq b$ format for the constraints and since the algorithm works solely with minimization, the constraints and the objectives which are *greater than or equal to* inequations were inverted accordingly.

This strategy proved itself inadequate, though. Due to the large amount of variables and the algorithm complexity, *gamultiobj* was not able to find starting points.

Considering this situation, a second approach had been chosen. The Normalized Normal Constraint (NNC) is a strategy first explained by (Messac *et al.*, 2003). First, a known search space is normalized as well as its anchor points, determined based on the objectives available - i.e. the anchor points are the best points from the standpoint of the individual objectives. Afterwards, an Utopia line is drawn between the anchor points. Then, using the user-defined number of solutions to be determined, the algorithm divides the line between equally-divided regions. These regions are then used by the algorithm to find the best possible solutions evenly spread along the search space. In the end, the algorithm denormalizes the values found.

Based on an adapted implementation of MATLAB®'s nonlinear programming solver *fmincon* it was possible to successfully execute the tests. However, due to the complexity of the problem, the following strategy had been chosen considering the computing and time constraints:

1. 2 runs would be executed, one with a limit of 430 thousand iterations per *fmincon* call and another with a limit of 1 million iterations;

2. Each run would attempt to find 10 solutions across the Utopia line;

3. Each run would, in parallel, attempt to find both anchor points. After both are found, the algorithm attempts to optimize in parallel each one of the possible solutions.

With the considerations above the algorithm took approximately 70 hours to be completely executed in an 8-core desktop and completely used all the processors during the whole step.

4.3 Multi-criteria Decision Making

Considering the quantity of objectives and solutions generated, instead of mathematical tools or other visual strategies such as radar plots or parallel coordinates, simple two-dimensional scatter plots were chosen instead. These plots uses the median reliability rates and the sum of the profitability for the suggested production distribution in its axes. The points in the plot shows the position of the solutions relative to both axes. If the number of proposed solutions were bigger to a point that the decision-making was difficult to the problem designer other strategies could be used instead.

As seen in the first plot of the Fig. 1, below, running with 430 thousand iterations proved that the available window to optimize the reliability was very small, with less than 2% of difference between both anchor points. However, by maximizing the reliability the profitability would be too affected, with a financial difference of approximately $3 * 10^8$, or more than 50%, from the anchor point belonging to the profitability maximization. Even after considering the difference caused by the changes in the reliability the variance in the weighted profitability would still be too high, as seen in the bottom plot of the Fig. 1.

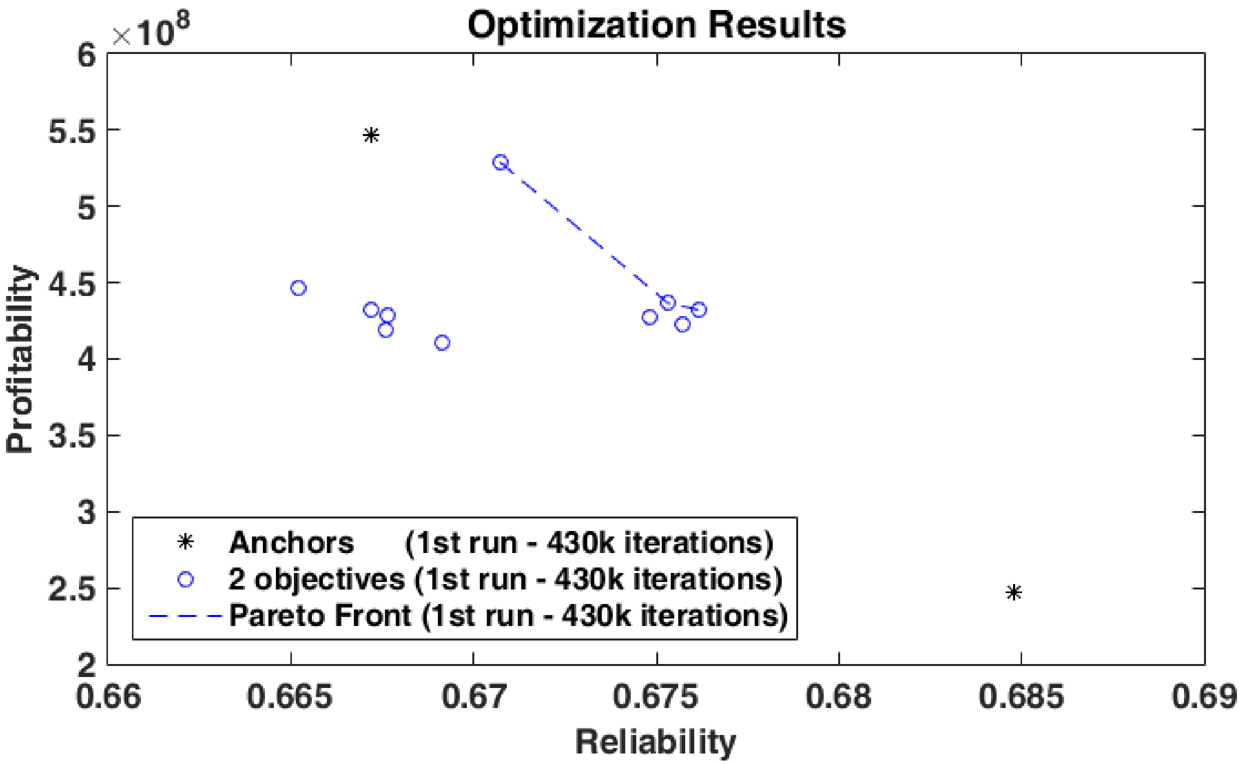

Figure 1. Results of the first run done with 350000 iterations. The first plot shows the tradeoffs between both objectives while the second plot shows the comparison between the theoretical maximum profits and the more realistic profits taking in account the reliability rates found for each solution.

By choosing the best solution found (i.e. the only solution with more than $5 * 10^8$ in profitability found) it was possible to determine that, although there was in fact an inverse relationship between profitability and reliability, the differences were negligible. By choosing this solution instead of the solution found in a single-objective approach - i.e the profitability anchor point - there was an increase of approximately 0.3% in the reliability with a decrease in the profits of around 3.2% or, when the own reliability is taken in account in the determination of the weighted profitability, the decrease is of approximately 2.7%. Also, since all but one solution had expected profits closer to the maximum profits and the difference in the reliability is small in percentage, it is possible to verify that more accurate solutions could be generated if more iterations were provided.

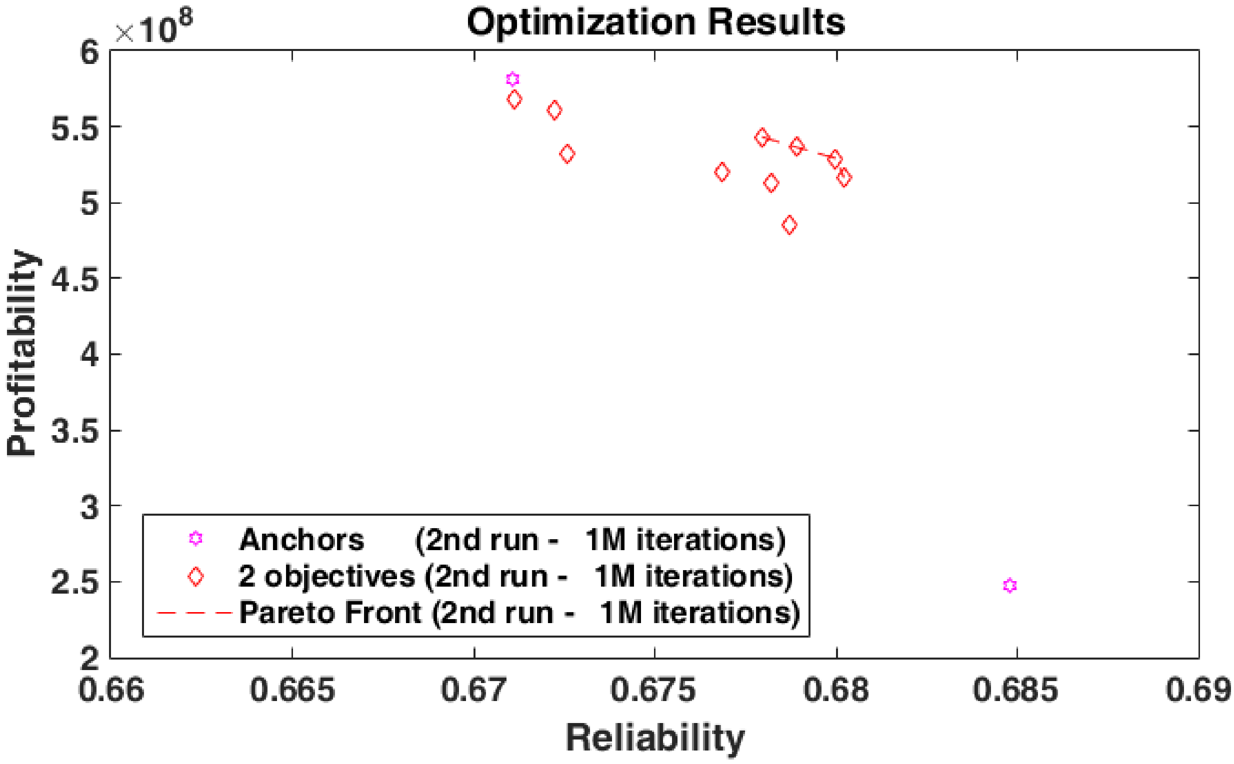

Figure 2. Results of the second run done with 1 million iterations. The first plot shows the tradeoffs between both objectives while the second plot shows the comparison between the theoretical maximum profits and the more realistic profits taking in account the reliability rates found for each solution.

In order to test this hypothesis, the second run, with 1 million iterations per *fmincon* call took place. As seen in first plot of the Fig. 2, the results were more evenly spread as expected. The profitability anchor point had higher values for both objectives, with all the solutions following the same pattern. However, the reliability anchor could not be improved since it had already reached its best value. Also, as shown in the second plot, the same situation of the first run repeated itself - by choosing almost any solution the tradeoffs would be negligible - although the reliability improvement could be a higher, but still under 1%, the reduction in the weighted profitability would stay between 2% and 3% for the best solutions within the Pareto front.

In this MOP, it is also important to note a pattern associated with the number of iterations used. As seen in the Fig. 3, the results had better solutions with their general positions located both to better reliability and profitability, mainly due to the increased number of iterations. As seen in the second plot of the Fig. 3, for example, the anchor point for the profitability in the first run is worse than at least three solutions found for the second run from a profitability standpoint.

On the other hand, the maximum reliability possible could not be improved due to the limits placed by the problem. Since the losses in the profitability would be too high for small improvements in the reliability, for both runs the solutions would be more oriented towards the profitability anchor.

5. CONCLUSIONS

The implementation of the NNC algorithm which is a type of Multi-objective Optimization Design proved itself to be useful in the problem of the production plan generation in the poultry industry considering its large quantity of variables and restrictions. More specifically, MATLAB®'s *fmincon* approach was able to successfully process the large data sets instead of MATLAB®'s *gamultiobj* algorithm.

On the problem itself, it was possible to prove the feasibility of the strategy with the data provided. There is indeed an inverse relationship in the reliability and profitability - in other words, the previous single-objective algorithms attempted to generate the production plan with the maximum profitability possible. However, this aggressive strategy reduced the production plan reliability which meant that the risk of the plants to not fully commit to the plan and therefore reduce the real profits was greater than more moderate strategies. Analogously, conservative strategies which attempt to maximize the plan reliability caused impacts in the estimated profits large enough to be discarded. Therefore, moderate solutions which attempted to balance both objectives could represent the best strategies from the solution designer standpoint.

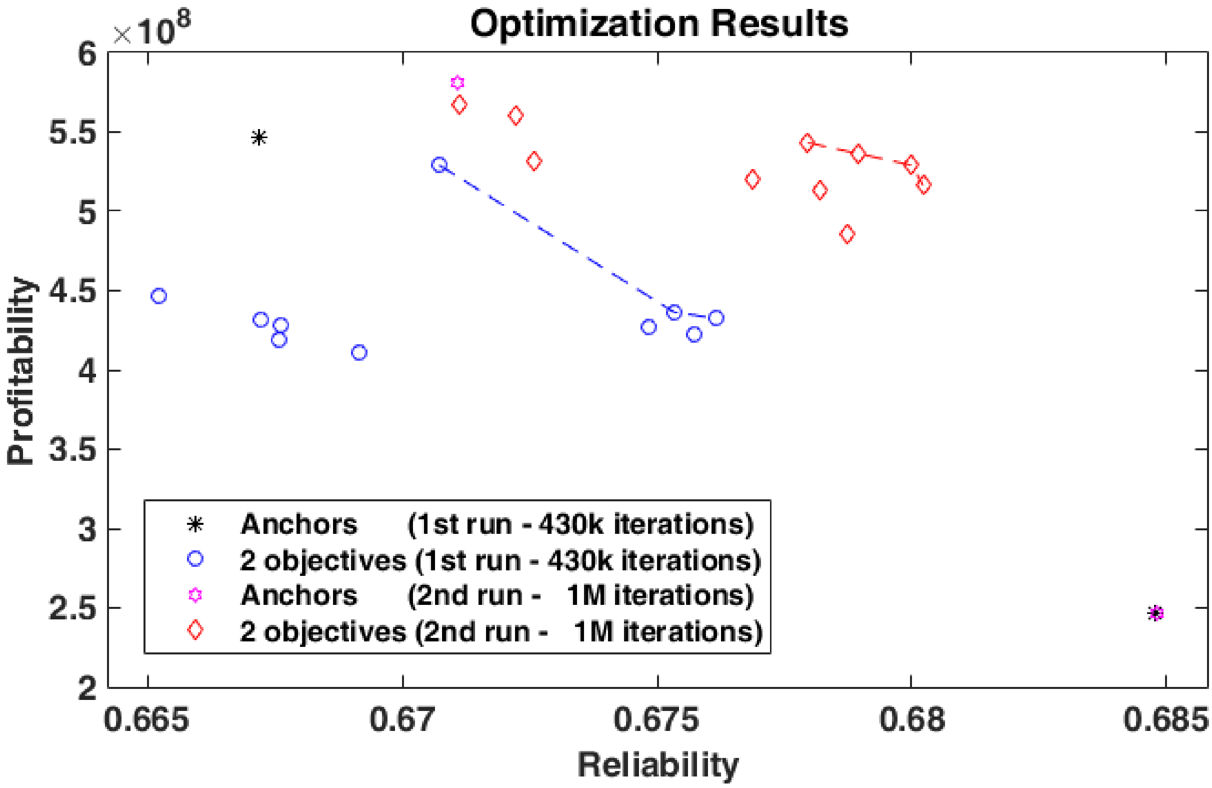

Figure 3. Comparison between both runs. The second run achieved noticeably better results.

Having that said, on the original dataset provided there were some issues identified in the data itself - two main issues were found and further validated by the data provider (i.e. the analysts in charge of the data management): the constraints were too tight and the lack of diversity in the reliability rates. Both issues also encouraged the analyst in charge of the data to find ways to better improve the source so that it can be more useful in the future.

On the constraints, the data provided had an exact sum between the plant capacity and the market demands - i.e. the sum of all the plant capacities for all the weeks was exactly the sum of the market demand for all the materials. Therefore, the algorithm efforts to find both valid combinations (i.e. solutions that not only complied with all the constraints) but also that improved upon the solutions from the previous generation were greatly increased. If there was more room to improve (e.g. if the plant capacities were greater than the market demand and/or if it was possible to not fully cover all the market demands under lesser penalties) the algorithm would have a less restricted search space resulting in increased performance and possibly better solutions - in this case, one possibility is to convert some restrictions into additional objectives. If the aforementioned features were introduced in the MOP design, additional constraints delimiting additional real-world scenarios (e.g. batch size limits, warehouse preferences, etc.) could also be introduced while keeping the same performance.

On the reliability rates, as previously informed there were nine plants available with only two grades available - 55.6% and 77.8%. Even if the grades were normalized there would still be a small diversity across the reliability axis in the Pareto front due to the number of grades available. For this reason, it was suggested to the data provider to make use of different techniques to evaluate the plants. If they could be evaluated with more diverse grades - at least 5 different grades instead of 2 - a more diverse Pareto front would be generated.

Considering the dataset provided, two major issues were identified in the data itself: the constraints were too tight, keeping the sum of both market demand and production limits equal, which reduced the available space for solutions, and the lack of diversity in the reliability rates for all the plants which greatly reduced the search space from a reliability standpoint. By loosing the constraints the algorithm would be able to find more combinations to distribute the most profitable materials to the most reliable plants. On the other hand, redefining the problem by having more diverse reliability rates could lead the algorithm to attempt to redistribute the production between the plants and also enable more diverse median reliability rates across the proposed solutions. This, by extension, would further enable more tangible tradeoffs (e.g. the designer could be able to find solutions with tradeoffs such as reducing the potential profits in 7%, but improving the reliability in 10%).

From an algorithm standpoint, this study also enabled the possibility to create a new strategy specialized in real world, complex scenarios like the one presented here - i.e. with a large number of variables and restrictions under a multi-objective problem. Under that sense, this new algorithm is currently being designed by the author in order to offer better performance (i.e. reduce the computing time) as well as keeping a good Pareto front and, therefore, a good set of solutions generated. The final objective on this is to develop a technique that is useful for the industry by providing a good set of solutions in a timely fashion and with an acceptable degree of fine-tuning by the problem designer.

6. ACKNOWLEDGMENTS

The authors would like to thank the Pontifical Catholic University of Parana for supporting the research as well as the Coordination for the Improvement of Higher Education Personnel (CAPES) and to the Brazilian Council for Scientific and Technological Development (CNPq) for their financial support through the PROSUC fellowship and the PQ2|304066/2016-8 grant.

7. REFERENCES

Behzadian, M., Kazemzadeh, R.B., Albadvi, A. and Aghdasi, M., 2010. “Promethee: A comprehensive literature review on methodologies and applications”. *European journal of Operational research*, Vol. 200, No. 1, pp. 198–215.

Blasco, X., Herrero, J., Sanchis, J. and Martínez, M., 2008. “A new graphical visualization of n-dimensional pareto front for decision-making in multiobjective optimization”. *Information Sciences*, Vol. 178, No. 20, pp. 3908–3924.

Christopher, M., 2016. *Logistics & supply chain management*. Pearson UK.

Coello, C.C., 2006. “Evolutionary multi-objective optimization: a historical view of the field”. *IEEE computational intelligence magazine*, Vol. 1, No. 1, pp. 28–36.

Flanders, F. and Gillespie, J.R., 2015. *Modern livestock & poultry production*. Cengage Learning.

Fonseca, C.M., Fleming, P.J. *et al.*, 1993. “Genetic algorithms for multiobjective optimization: Formulationdiscussion and generalization.” In *Icga*. Vol. 93, pp. 416–423.

Huang, H.Z., Gu, Y.K. and Du, X., 2006. “An interactive fuzzy multi-objective optimization method for engineering design”. *Engineering Applications of Artificial Intelligence*, Vol. 19, No. 5, pp. 451–460.

Marler, R.T. and Arora, J.S., 2010. “The weighted sum method for multi-objective optimization: new insights”. *Structural and multidisciplinary optimization*, Vol. 41, No. 6, pp. 853–862.

Messac, A., Ismail-Yahaya, A. and Mattson, C.A., 2003. “The normalized normal constraint method for generating the pareto frontier”. *Structural and multidisciplinary optimization*, Vol. 25, No. 2, pp. 86–98.

Miettinen, K., 1999. “Nonlinear multiobjective optimization, volume 12 of international series in operations research and management science”.

Rao, S.S., 2009. *Engineering optimization: theory and practice*. John Wiley & Sons.

Reynoso-Meza, G., Blasco, X., Sanchis, J. and Martínez, M., 2014. “Controller tuning using evolutionary multi-objective optimisation: current trends and applications”. *Control Engineering Practice*, Vol. 28, pp. 58–73.

Van Veldhuizen, D.A. and Lamont, G.B., 1998. “Evolutionary computation and convergence to a pareto front”. In *Late breaking papers at the genetic programming 1998 conference*. pp. 221–228.

8. RESPONSIBILITY NOTICE