

Variable Neighborhoods Search for Multi-Site Production Planning

Agung Mustika Rizki¹, Gusti Eka Yuliasuti², Wayan Firdaus Mahmudy³,
Ishardita Pambudi Tama⁴

^{1,2,3}Computer Science, Faculty of Computer Science, Brawijaya University

⁴Industrial Engineering, Faculty of Engineering, Brawijaya University

{agungmustikarizki@gmail.com, gustiekay@gmail.com, wayanfm@uib.ac.id, kangdith@uib.ac.id}

Received: 26 August 2018; accepted: 24 October 2018

Abstract. In the home textile industry, production planning needs to be done so that the production costs incurred by the company can be well controlled. Production planning is a problem that cannot be solved in a short time. Problems are more complex if the company has several production branches in other cities, with rules and standards that are certainly very different from one city to another. Based on this background, an algorithm is needed that can solve production planning problems for companies with many production branches in order to obtain optimal solutions. VNS is applied by the author and produces an optimal and efficient solution because the time needed is relatively short compared to the planning carried out previously by the company.

Keywords: production planning, multi-site production planning, variable neighborhoods search.

1. Introduction

The government in Indonesia has an active role in supporting the movement of domestic industry. One industry that continues to grow today is the textile industry. This is because the textile industry is a basic part of the needs of human life. But in the industrial sector there are several complicated obstacles. One of these problems is that industry players must plan the production process appropriately to deal with changing market demands. This fluctuation in demand occurs due to several factors including people's purchasing power, economic development and the emergence of new industry players in the textile sector. Thus the implementation of production must be carried out efficiently so as not to overdo it or lack it. For that, production planning is needed so that consumer demand is met properly. In this case the planning of aggregate production can be done to produce this efficient production process.

Aggregate production planning is a plan that aims to determine the amount of production, inventory, employees, overtime level, and demand satisfaction level which leads to minimum costs or maximum profits [1]. In the early stages of aggregate production planning is estimating consumer demand as input from the planning process. Furthermore, it also takes into account the availability of materials, workers, production time and costs required. Like combinatorial problems in general,

the resolution of this problem is quite complicated. Besides that, the settlement in the normal way requires energy, thought and a short time to get the most optimal results.

Aggregate production planning is a difficult job for company decision makers because if there is a slight error, it will have a fatal impact on the costs incurred by the company. In addition, it will be more difficult if the company has a production branch in another city (multi-site) because the basic rules that apply in each production branch in each city are certainly very different. It requires an algorithm that can be used to complete the planning of aggregate production optimally.

There are several algorithms that can be applied to solve these problems, one of which is Variable Neighborhood Search (VNS). There have been previous studies conducted by Xu et al.[2] who applied VNS to solve flexible job shop scheduling problems. The VNS algorithm aims to minimize makespan, obtain the smallest maximum engine load and the smallest total engine load. The results obtained by them indicate that VNS is quite optimal compared to other classical algorithms to verify that if the search algorithm for multi-population variable-population environment is a feasible and effective optimization or not. Another case with research conducted by Tasgetiren et al.[3] related search environment variables to solve the problem of multidimensional backpacks. The results obtained after the application of VNS are optimal with relatively short computational time. VNS can also solve the problem of fuzzy numbers in linear programming [4]. The objective function values obtained with VNS change to be more accurate than other methods. Based on the background related to the results of the application of VNS, the author also applies VNS to resolve the problems of production planning with many regional production branches.

2. Problem Description

Aggregate planning is a medium-term capacity plan that typically covers 2 to 12 months. This planning is especially useful for companies with uncertain, volatile or fluctuating demand. The purpose of aggregate planning is to make a production plan by using company resources to effectively fulfill demand. Planning carried out takes a decision by considering the level of results, the level of work and changes in employment, level of inventory and changes in inventory, pending orders, and subcontracting [5].

The main problem in this study is a home textile company that has several regional production branches in several cities in East Java (multi-site). The problem faced by this company in addition to fluctuating consumer demand is that there are differences related to production capacity and also standard rules for each regional production branch. The calculation of aggregate production planning in one branch is certainly not the same as the other branches, this is something that needs to be considered so that there is no swelling of production costs at all branches. This problem is not a simple thing that can be resolved in a short time because it must consider many parameters in it to obtain optimal solutions.

There are several different standard rules related to production parameters in each production branch. Some of the production parameters include: work time, regular time production costs, overtime production costs, subcontract production costs, storage costs, new employee recruitment costs, costs of laid off workers, and so forth. For example In the first regional branch there were 4 workers, a maximum of 2 subcontract workers, an average production of regular time workers as many as 13 products, an average production of overtime workers of 4 products, an average production of subcontract workers of 13 products, the maximum storage limit is 4000

products, the minimum production limit is 500 products, the cost for regular time workers is Rp. 23,000, - for each product, the cost for workers overtime amounts to Rp. 24,000, - for each product, the cost for subcontract workers is Rp. 22,000, - for each product. The value of these parameters differs from other regional branches.

3. Variable Neighborhoods Search

The Neighborhoods Search Variable (VNS) is an algorithm that applies the process of finding solutions based on proximity, both far and near, so that it is most likely to produce the best solution. The search for solutions to neighboring which is far from being interpreted as a random system search so that it can reach a wide range of problems [6].

Hansen and Mladenovic carried VNS as a metaheuristic based algorithm. VNS explores neighbors from the latest incumbent solutions. Most solutions that are at optimal value will be stored and used to obtain promising neighboring solutions [7]. Simple and effective metaheuristics can be obtained through processing systematic changes in local search.

The main flow of VNS begins with initializing the solution then shaking, local search and selecting move or not [8]. VNS iteration continues to be carried out using local search technique where in each iteration the technique explores the solution search space from a new point determined by the neighboring structure. This neighboring structure serves to generate new solutions by changing the initial solution. Solutions that have undergone change will be used as potential solutions if there has been a better quality improvement solution. The flow used in VNS mostly uses iteration until it encounters a satisfactory stop condition. A large $kmax$ value will provide a better solution even though it requires high computation time.

There are several factors that make up the VNS algorithm, including [6]:

- a) **Solution Value**, is the initial value that represents the solution to a problem. The value of this solution is generated randomly.
- b) **The Value of k** , is the neighboring value calculated in the process of producing a solution.
- c) **Shaking Technique**, is technique used to produce new solutions. The value of a solution will be exchanged / scrambled randomly.
- d) **Local Search Technique**, this technique is also used to generate new solutions after going through the shaking process. In local search, some values in the solution will be changed to a number n according to the values specified at the beginning.

4. Experimental Analysis

Initial solutions value were generated as many as 72 random numbers consisting of production values for 6 production branches within 12 months. Then the authors determine the initial parameters used for testing. The testing was conducted to find out how the results of the VNS algorithm were applied and also how the quality of the solution was produced.

In VNS, the iteration value is an important factor because it will affect the computation time. Besides that there is a neighbor value that will also be tested. Neighbor's value is also an important part of the VNS process because in VNS there are conditions which state that if the results are not obtained better the system will be trapped and will not stop. For this reason, it is necessary to obtain optimal neighboring values.

In testing the neighboring value (k) from 1 until 9 neighbors is done 10 times with an iteration value of 250. Besides the k value, the computation time is also taken

into account in this testing because the iteration of the value of k can be repeated until infinite or commonly referred to as the never ending looping.

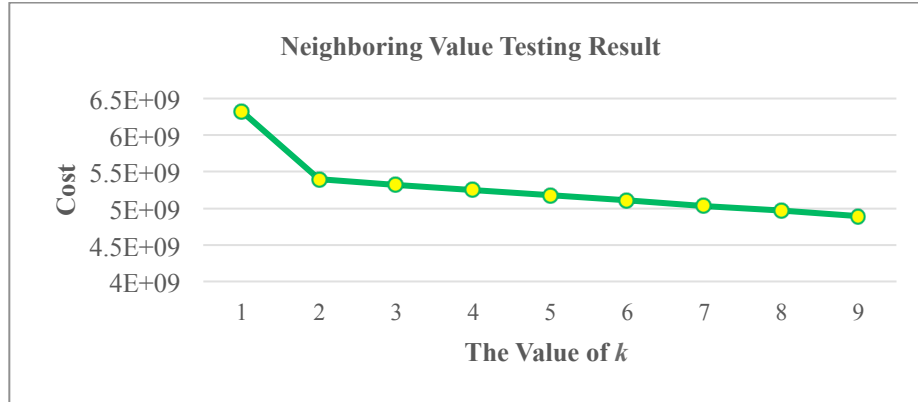


Figure 1 Neighboring Value Testing Result

The Figure 1 indicates that the higher the neighbor's value the better the results obtained. This happens because the neighboring values in the VNS have conditions that will only accept new solutions when the new solutions generated are better than the previous solutions.

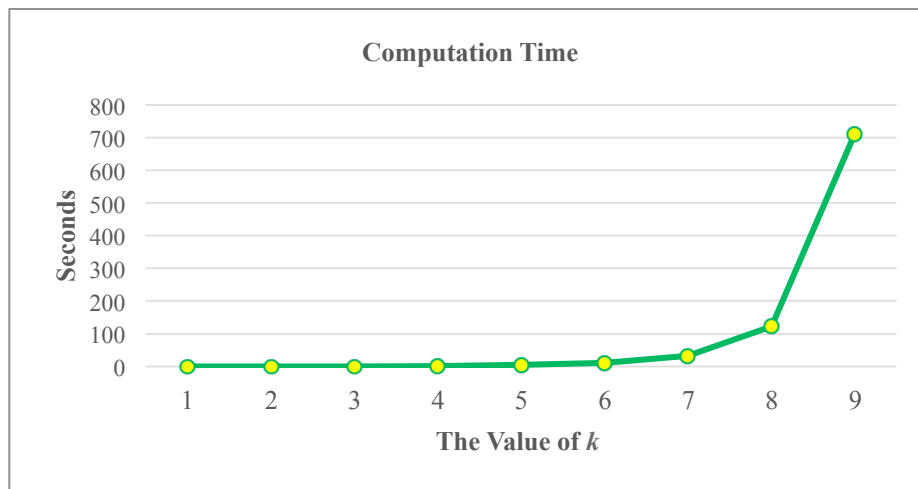


Figure 2 Computation Time of k Value Testing

Based on the Figure 2, it can be concluded that the optimal k value is 3. The determination of this value is based on the results obtained are good enough with a computation time that is still not too long, 0 seconds.

In the other hand, authors also tested the iteration to get to know the best iteration value for VNS process as shown in Figure 3.

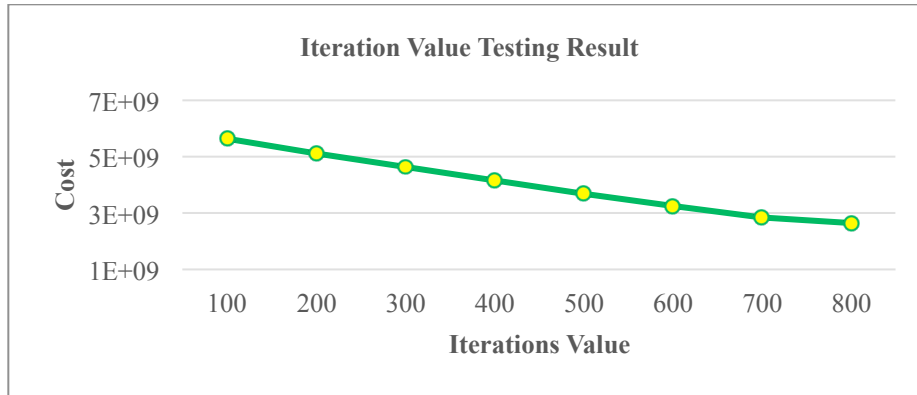


Figure 3 Iteration Value Testing Result

The best results are stored for use in subsequent tests. Testing the iteration value serves to get a better solution value. The test of VNS iteration value is carried out in the range of 100-1000 with a multiple of 100. Each test is carried out 10 times and the average value is taken.

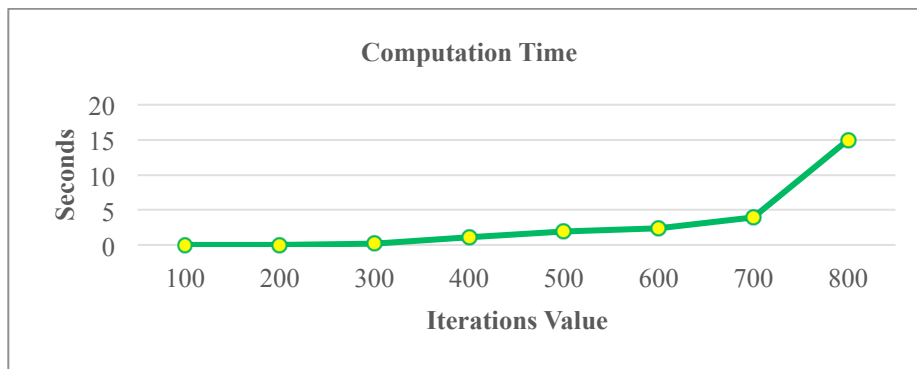


Figure 4 Computation Time of Iterations Value Testing

Based on Figure 3 and Figure 4, the graph results of iteration testing on VNS tends to decrease proportionally to the number of iterations but the computation time increase. From the results of these tests can be taken the optimal iteration value that is during the iteration of 300 iterations. When this iteration is selected taking into account the computational time required. The best solution for VNS algorithm produces a production value of Rp. 2,778,862,400,- for 6 regional production branches survived 12 months.

5. Conclusion and Future Work

The application of VNS algorithm to solve the problem of optimal multi-site production capacity is quite efficient. Efficient because the time needed to get a production capacity solution is not as long as the company did before.

VNS algorithm which has advantages in local search. But this advantage is also a shortage of VNS. Searches that focus solely on local search will result in narrow exploration space. This also results in less than optimal results. With these deficiencies, the author will combine the two algorithms to improve the quality of the solutions produced for future research.

References

- [1] O. Saracoglu, M. C. Arslan, and M. Turkay, "Aggregate Planning Problem from Sustainability Perspective," *Int. Conf. Adv. Logist. Transp. Aggreg.*, pp. 181–186, 2015.
- [2] L. Xu, S. Weiping, and H. Ming, "Flexible Job Shop Scheduling Based on Multi-Population Genetic-Variable Neighborhood Search Algorithm," in *International Conference on Computer Science and Network Technology (ICCSNT)*, 2015, vol. 1, no. Iccsnt, pp. 244–248.
- [3] M. F. Tasgetiren, Q. K. Pan, D. Kizilay, and G. Suer, "A Differential Evolution Algorithm with Variable Neighborhood Search for Multidimensional Knapsack Problem," *IEEE Congr. Evol. Comput.*, pp. 2797–2804, 2015.
- [4] K. Ghorbani-moghadam, R. Ghanbari, and N. Mahdavi-amiri, "Solving Fuzzy Number Linear Programming Problems using A Variable Neighborhood Search Algorithm," in *Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS) Solving*, 2018, no. 1, pp. 31–33.
- [5] W. J. Stevenson and S. C. Chuong, *Manajemen Operasi: Perspektif ASia*, Edisi 9. Jakarta: Salemba Empat, 2014.
- [6] W. F. Mahmudy, "Optimization of Part Type Selection and Machine Loading Problems in Flexible Manufacturing System Using Variable Neighborhood Search," *IAENG Int. J. Comput. Sci.*, no. July, 2015.
- [7] N. Mladenović, R. Todosijević, and D. Urošević, "An efficient GVNS for solving Traveling Salesman Problem with Time Windows," *Electron. Notes Discret. Math.*, vol. 39, pp. 83–90, 2012.
- [8] P. Hansen and N. Mladenović, "Variable Neighborhood Search: Principles and Applications," *Eur. J. Oper. Res.*, vol. 130, no. 3, pp. 449–467, 2001.