Journal of Engineering Issues and Solutions

Enhancing production efficiency through integer linear programming-based production planning

Saurabh Chaudhary[1,2,*], Sushant Raj Giri[2]
[1]*Herald College Kathmandu, University of Wolverhampton, Sano Gaucharan, Kathmandu*
[2] *Department of Industrial Engineering, Thapathali Campus, IOE, Thapathali, Kathmandu*
**Corresponding email: skcsaurabhkumar@gmail.com*
Received: March 3, 2024; Revised: June 9, 2024; Accepted: June 16, 2024

doi: https://doi.org/10.3126/joeis.v3i1.63469

Abstract

This paper investigates the effectiveness of operations research in addressing traditional production planning issues and determining potential of decreasing production costs by implementing visible constraints. This study used an integer linear programming (ILP) model to forecast the monthly batch output from each product to meet demand, with a focus on the manufacturing facility at Shree Pashupati Biscuit Industries Pvt. Ltd. The aim is to minimize the plant's production costs. The labor and machine hour constraints are regarded as soft constraints, whereas demand constraints are regarded as hard constraints. The model was built as a python program and solved using the simplex approach with PuLP library. By efficiently employing both human and physical resources, the model calculates the lowest monthly cost and the number of batches that must be produced. The required overtime expenses, labor and machine idle periods, and additional hours are also computed by the solution and added to the monthly production costs. By efficiently utilizing both human and physical resources, the production approach avoids producing excessive amounts of biscuits. This study explores methods to improve traditional production planning problems and effectively minimize production costs by considering multiple constraints through operations research. This study exactly finds out the difference in monthly production cost by using linear programming to prevalent production planning system. In addition, the method can be used to compute overtime expenses, labor and machine idle periods, and additional hours worked. These figures are then added to the monthly production costs.

Keywords: *Hard constraints, integer linear programming, Python, soft constraints.*

1. Introduction

Production planning is crucial for modern manufacturing operations, as it helps identify inefficiencies and implement necessary changes. It involves understanding the name, code, quantity, volume, and raw material requirements of an item as well as technical specifications for quality control at each level (Hamal et al., 2023). It was realized that production planning (PP) problem was crucial in Shree Pashupati Biscuit Industries Pvt.

Ltd. The manufacturing plant had integrated Just in Time (JIT) production and inventory management system which was not adequate considering the large capacity of plant's inventory and high demand of products in the market. It was also identified that JIT aims to reduce inventories, but requires a pull system, setup time reduction for smaller batches, and stable, reliable production operations for successful operation (Afriansyah & Mohruni, 2021), which was not required by the manufacturing facility. Similarly, research conducted by (Li et al., 2017) proposes a mathematical model for production planning in a foundry flow shop, aiming to maximize profit by considering material, process, delay, and facility occupy costs. An Improved Genetic Algorithm (IGA) is introduced to address the interaction between minimizing delay and facility occupy costs. The model and IGA approach demonstrate better profit rates in the foundry industry. However, they state that future research should consider additional factors like labor constraints.

Planning, scheduling, and synchronizing all production processes need effective implementation for a manufacturing plant to operate correctly. Since PP promotes efficiency, removes limitations, and organizes production flow, it is essential to organizational management. Before the manufacturing process starts, it directs tasks and time constraints to guarantee an organized and effective workflow (Afolalu, et al., 2021). Efficient production planning strategies can improve businesses' production performance, operational effectiveness, and resource utilization by considering demand changes and fostering cooperation among units (Khezrimotlagh & Chen, 2018). The manufacturing or production department is responsible for creating a production plan, a crucial component of a company plan. To meet customer demand with the available plant resources, plant management must establish the total output that the manufacturing plant should produce from each period in the planning horizon. The PP problem can be solved quantitatively in a variety of ways. Using a mathematical model to resolve a PP problem is one approach. Operations research (OR) applications are frequently used in the industrial sector because of their capacity to maximize gains in a given scenario. Linear programming is a significant branch of mathematics known as 'optimization techniques' used in real-world applications (Oladejo et al., 2019). The production planning model utilizes data-driven approaches to improve decision-making in large-scale production environments, focusing on resource considerations like labor, materials, and machinery for efficiency and cost-effectiveness (Christefa et al., 2022). Linear programming (LP) is a powerful tool for managers to optimize resource allocation, operations, and profit maximization. However, unique data input is needed for effective production planning (Solaja et al., 2019).

This paper solves an aggregate production planning (APP) problem for Shree Pashupati Biscuit Industries Pvt. Ltd. using operations research and Python programming as part of a quantitative methodology. By implementing this production plan, extra biscuit production may be prevented, maximizing the use of both human and physical resources. This can be achieved by utilizing LP to build up a mathematical model. LP is considering various linear inequalities in multiple circumstances and locating the best possible solution within those constraints. Most research identified in the literature uses LP strategies to handle PP problems (Alden et al., 2023) (Hojati & Patil, 2011) (Jamalnia & Soukhakian, 2008) (Nash, 2000) (Oladejo et al., 2019) (Solaja et al., 2019) (Wang & Fang, 2001). Many researchers employed LP techniques to address the APP problem. APP aims to determine each product's total (aggregate) production amounts (Alden et al., 2023). The APP problem has been solved using many methods, such as goal programming (Jamalnia & Soukhakian, 2008). Goal setting is a managerial choice; hence, the issue is simplified to a LP issue. Due to insufficient or unavailable information, the input data or parameters for real-world APP problems, such as demand, resources, cost, and objective function, could be more precise (fuzzy).

Additionally, (Wang & Fang, 2001) introduced a fuzzy linear programming method for solving the APP and acknowledged the fuzziness of parameter values. However, the manufacturing plant was where the study's cost and resource data came from. As the monthly demand is determined by the top management and communicated to the production managers at the beginning of each month, the complexity of the problem decreases. An LP model can be created to tackle this PP problem using a single month as the only uniquely determined data input. To meet the monthly demand with the resources at hand while reducing the production cost, the study aims to identify the number of batches that must be produced from each product each month. All variables must be integers because the factory does not create incomplete batches. So, in this study, the PP problem is resolved using an integer linear programming (ILP) model. All of the choice variables in our mathematical model must be integers (Wolsey, 1998). However, the mixed integer linear programming (MILP) problem can be used to address the issue when all the variables are integers.

The study aims to determine the number of batches produced from each product each month to meet monthly demand with available resources and reduce production costs. An ILP model is used to address the PP problem, which requires all variables to be integers (Wolsey, 1998). The MILP problem can also be employed when all variables are integers. A better production plan is crucial for reducing time spent in stores, preventing additional inventory expenditures, and ensuring product quality (Jamalnia et al., 2019). The factory manufactures biscuits made of soft and firm dough, with some unique items requiring additional manufacturing steps. The mathematical model considers these variations and calculates coefficients for the objective function and restrictions. Multiple constraints are identified and converted into linear functions using python programming. The paper seeks answers to traditional production planning problems using operations research and if implementing visible constraints can reduce production costs.

2. Materials and Methods

The number of batches to be produced from each item was calculated mathematically using ILP to minimize cost while adhering to the limitations of resources and meeting the demand constraint (see Table 1 for a list of the products selected for the study). The following sections show how all the required information was gathered and how the estimated coefficients for the objective and constraints matched up.

Table 1: List of products for study

Type	Product	Creaming	Flavoring
Hard Dough	Milk Puff (P_1)	Yes	Yes
	Cheese Crackers (P_2)		Yes
	Marie (P_3)		
Soft Dough	Twist (P_4)		Yes
	Soaltee (P_5)		Yes

2.1 Formulating the mathematical model

Let x_i represent the number of batches that will be created from the i^{th} product (P_i) each month, where, I = 1, 2,…., 5.

2.1.1 Objective Function

The goal is to reduce the monthly manufacturing cost of packed biscuits and baked cookies that are heaped into bins rather than being boxed. The production cost per kilogram of unpackaged biscuits was gathered as raw information. It includes the cost of raw materials, labor, oven fuel, electricity, etc. These expenses were added, and the following objective function was created:

Let C_i represent the cost of manufacturing each completed batch of the i^{th} product.

$$\min Z = \sum_{i=1}^{5} C_i x_i + OC \;; \qquad (1)$$

Where,

Z is the linear function representing minimized cost

C_i is the cost per batch of the i^{th} product

X_i is the number of batches of the i^{th} product

OC is the overtime cost.

Table 2: Wastage and weight loss in each section

Product	Dough weight (kg)	Baking loss %	Waste at mixing (kg)	Waste at cutter (kg)	Waste in cooling and stacking %
Milk Puff (P_1)	350	14	1.35	2	1
Cheese Crackers (P_2)	350	13.21	1.2	2.5	1
Marie (P_3)	350	20	1	3.5	1
Twist(P_4)	350	10.28	0.85	1.5	1
Soaltee(P_5)	350	4.28	1.15	3	1

Table 3: Weight loss of baked biscuits per batch

Product	Dough weight (kg)	Waste in mixing (kg)	Oven input (kg)	Baking loss (kg)	Oven output (kg)	Waste in cooling & stacking (kg)	Final weight of baked biscuit batch (KG)
Milk Puff (P1)	350	1.35	348.65	48.81	299.84	3.00	296.84
Cheese Crackers (P2)	350	1.2	348.8	46.08	302.72	3.03	299.70
Marie (P3)	350	1	349	69.80	279.20	2.79	276.41
Twist (P4)	350	0.85	349.15	35.89	313.26	3.13	310.12
Soaltee (P5)	350	1.15	348.85	14.93	333.92	3.34	330.58

Table 4: Cost per batch

Product (Pi)	Production cost/ kg (Rs) Ki	Final weight	Cost per batch
Milk Puff (P_1)	140	296.84	41557.69
Cheese Crackers (P_2)	135	299.70	40459.00
Marie (P_3)	128	276.41	35380.22
Twist (P_4)	132	310.12	40936.47
Soaltee (P_5)	112	330.58	37024.96

Table 2 illustrates the weight losses and waste generated during different processes of biscuit manufacturing. The weight of the wet dough mixture was then adjusted to account for all the wastes and weight losses listed in Table 2. Assuming that these are the only wastes that will happen during the production process, the model was created and solved. Table 3 displays the relevant information. It computes the weight of a batch of cooked cookies (B_i) which is according to the formula of equation 2. The following formulas can be used to determine the cost per batch of the i[th] product (C_i), given that the price per kilogram (k_i) was gathered as raw data. Table 4 contains the computations for C_i according to the formula stated by equation 3.

Bi = Dough Weight – Waste in Mixing -Baking Loss -Waste in Cooling & Stacking.................................. (2)

Ci = Production cost per kilogram of biscuits (Ki) ×Weight of a baked biscuit batch (Bi)............................(3)

Therefore, the objective function according to equation 1 for the concerned problem can be written as:

Min Z = 41557.69 x1 + 40458.99 x2 + 35380.22 x3 + 40936.47 x4 + 37024.96 x5.......................................(4)

Where,

the integers are cost per batch of the i[th] product obtained from Table 4;

x_1, x_2, x_3, x_4, x_5 are the number of batches to be produced for five products accordingly.

2.1.2 Constraints

For the viability, all limitations must be met. However, many models include two types of constraints: soft constraints with varying relative importance that may or may not be satisfied, and hard constraints that any feasible solution must satisfy. In their discussion of applying both soft and hard restrictions in production planning. The possible solutions are defined by the hard constraints, while the function to be optimized in choosing between the feasible solutions is defined by the soft constraints. If a model contains both types of constraints, soft constraints can be changed until a workable solution is found.

1) Hard Constraint

In order to ensure that no consumer is dissatisfied, the management places the highest priority to meeting **demand**. The need must be satisfied by any workable solution that the model produces. Therefore, demand constraint was regarded as a hard limitation for this investigation.

B_i x_i provides the total number of kilograms that must be produced from the i[th] product each month since B_i represents the final weight of baked biscuits and x_i represents the number of batches to be produced from the i[th] product which can be stated according to the formula in equation 5.

Total Production = Final Weight of Biscuits (Bi) per batch ×Total number of batch (xi) .... (5)

Since the plant doesn't want to create any extra biscuits, this quantity should be the same as the kilogram demand for each product. As a result, the demand restriction can be expressed as follows:

Bixi $\geq Di$ for i = 1,2,…,5; .... (6)

Where,

$B_i x_i$ is total production per month in kilograms;

D_i is the monthly demand in kilograms for the i[th] product.

The senior management makes the monthly demand decision, which is then communicated to the biscuit plant at the beginning of each month. The quantity of boxes of cookies required for various stock keeping units (SKU) of each product is how the demand is expressed. These data were gathered from the manufacturing facility, and using them, the monthly demand in kilos for each product was computed. The calculations needed to determine the total demand in kilos per month for each commodity are listed in Table 5 which is according to the formula in equation 7.

Total demand in kg = no.of cartons demanded × no.of packets in a carton ×wt of biscuits per packet (in kg) ....(7)

Table 5: Total demand calculation in kilogram

Product (Pi)	SKU (gm)	No. of packets in a carton	Demand (in carton)	Monthly demand
Milk Puff (P_1)	40	120	285	1368
Cheese Crackers (P_2)	60	96	3000	17280
Marie (P_3)	160	18	1875	5400
Twist (P_4)	30	144	1500	6480
Soaltee (P_5)	35	144	6,000	30240

The demand restrictions for each product can be expressed as follows; once all the coefficient values and right-hand side values, which represent the requirements, have been calculated:

$296.84\ x1 \geq 1368$

$299.69\ x2 \geq 17280$

$276.41\ x3 \geq 5400$

$310.12\ x4 \geq 6480$

$330.58\ x5 \geq 30240$ .... (8)

Where,

The right-hand side integers are total demand in kilograms from Table 4;

The left-hand side values are total production in kilograms from Table 5.

2) Soft Constraint

The manufacturing company's resources are scarce. So, in order to reduce manufacturing costs, it's crucial to take into account the plant's resources. When making biscuits, the production process cannot use more resources than that are available, such as the number of workers and the machine's capacity. The management frequently modifies the amount of labor and equipment needed in accordance with the monthly demand. Therefore, labor and machine hour limits were viewed as soft constraints in this study.

- Machine Hour

Each product has a set processing time for a batch. The overall processing time needed for the monthly output should be less than or equal to the total monthly machine hours. The number of working hours per day and working days per month might be extended up to a certain point in the model solving step if the demand cannot be met with the total number of machine hours that are available.

Machine hour constraints were therefore viewed as soft constraints. The machine hour restriction is then, generally, as follows:

$$\sum_{i=1}^{5} t_i x_i \leq T \; for \; i = 1,2,..,5 \quad (9)$$

Where,

T is the total monthly machine time available (in minutes);

t_i is the total processing time (in minutes) per batch of the i^{th} product.

Table 6: Available machine hours per month

Available machine hours per month			
No. of working hrs per day	**No. of working days per month**	**Total available machine hours (hr)**	**T (min)**
16	26	416	24960

Table 7: Available labor time per month

Section	**Available no. of labours**	**No. of days**	**No. of hours**	**L (hr)**	**L (min)**
Mixing	5	26	16	2080	124800
Cutter	6	26	16	2496	149760
Baking, Cooling, & Stacking	7	26	16	2912	174720
Packaging	50	26	16	20800	1248000

Table 8: Required labor time

Product	**Total processing time**	**No. of labors needed**				**Required labor time (min)**			
		Mixing	**Cutter**	**Baking, Cooling & Stacking**	**Packaging**	**Mixing**	**Cutter**	**Baking, Cooling & Stacking**	**Packaging**
Milk Puff (P_1)	40	5	6	7	45	200	240	280	1800
Cheese Crackers (P_2)	50	5	6	7	38	250	300	350	1900
Marie (P_3)	45	5	6	7	40	225	270	315	1800
Twist (P_4)	39.29	5	6	7	45	196.45	235.74	275.03	1768.05
Soaltee (P_5)	37.14	5	6	7	40	185.7	222.84	259.98	1485.6

Table 6 calculates the number of machine hours that are available each month (T), and Table 8 includes total processing time per batch (in min). It is possible to express the machine hour constraint based on equation (9) as follows:

40 x1 +50 x2 +45 x3 +39.29 x4 +37.14 x5 ≤ 24960 ....(10)

Where,

The integers on left hand side are required processing time from table 2.8;

The value of right hand side is total available machine time from table 2.6.

- Labor Hour

The monthly labor hours available should be greater than the monthly labor hours required for the output. As a result, the overall formulation of the labor hour constraint for each section: mixing, cutter, baking and cooling, and stacking is as follows:

$\sum_{i=1}^{5} t_i l_i x_i \leq L$, ...(11)

Where,

ti is the amount of time (in minutes) needed to process one batch of the i^{th} product through each segment;

li is the number of workers required for each production section;

L is the total number of hours of labor that are available.

Each area of the industrial facility has a set number of laborers assigned to it (see Table 7). During the manufacturing of various products, different numbers of these laborers are assigned to each department (see Table 8). Additionally, Table 8 illustrates the calculations of the required labor time for each segment during production, while Table 7 displays the available labor time in minutes.

This limitation is also referred to as a soft constraint because the number of working days or hours can be changed until a workable solution is discovered. The following is how the labor constraint for each portion of this PP problem might be presented:

Mixing:

200 x1 + 250 x2 + 225 x3 + 196.45 x4 + 185.7 x5 ≤ 124800 ....(12)

Cutter:

240 x1 + 300 x2 + 270 x3 + 235.74 x4 + 222.84 x5 ≤ 149760 ....(13)

Baking, Cooling, and Stacking:

280 x1 + 350 x2 + 315 x3 + 275.03 x4 + 259.98 x5 ≤ 17472 ....(14)

Packaging:

1800 x1 + 1900 x2 + 1800 x3 + 1768.05 x4 + 1485.6 x5 ≤ 1248000 ....(15)

Where,

The integer values on left hand side is required labor time for different section derived from table 8;

The left-hand side values are available labor time for different section derived from table 7;

Additionally, because the plant doesn't manufacture incomplete batches, all of the x_i values should be positive integers. That is: $x_i \geq 0$ and integer for i = 1, 2,..,5.

2.2 Solving the formulated mathematical model

The Simplex Algorithm, created by Dantzig in 1947, is the most well-known method for resolving LP problems. It is known as Integer Linear Programming (ILP) because the defined model uses integer decision variables (Nash, 2000). As a result, the simplex method by itself is insufficient to resolve the issue. Cutting plane methods are a subclass of precise algorithms that can be used to solve ILP (Neto, 2012). In order to move the solution toward being integer without rejecting any integer viable places, this method first applies LP relaxation and then adds linear restrictions. The branch and bound method's variations represent another class of precise algorithms. Since ILP is NP-hard (Schrijver, 1999), many issues are intractable, hence heuristic methods are utilized in their place. ILPs can be solved using a variety of heuristic techniques, including ant colony optimization. But in this work, the ILP problem was solved using a spreadsheet model.

Model in Python

Python programming was used to develop the aforementioned integer linear programming paradigm. The model coefficients must be chosen with particular values when creating a linear programming model. Rough intervals are regarded as helpful new instruments to address the imprecise, ambiguous, and uncertain data in decision-making situations (E.E. & A, 2020). For formulating the linear model, PuLP library was used. Using expressions that are native to the Python language, users can develop programs with PuLP, a high-level modeling library that makes the most of the language's capabilities while avoiding specific syntax and keywords (Mitchell et al., 2011). The problem function was established once the library was imported, and then decision variables — the number of batches of various items to be produced — were formed (xi). Equation (4) was used to define the objective function. Next, the constraints were applied using the formulas for hard constraints (equation (8)) and soft constraints (equations 10, 12, 13, 14, and 15). After then, a Python solution to the issue was printed. The optimal choices of x_1, x_2, x_3, x_4, and x_5 that minimize the objective function while meeting all restrictions were determined by code programming. Furthermore, the code was designed to output the current status as well. A workable solution was established when the number of working hours was above the required hours, while the number of laborers assigned to each area was kept constant. The number of working hours and working days were modified within a specific range that was acceptable to the management.

3. Results and Discussion

3.1 Optimized solution for production efficiency

To identify the best answer, specific constraints have been created. While labor and machine hours were classified as soft constraints because they were not as necessary to be satisfied, they were still desirable to be taken into consideration. Demand constraints, on the other hand, were classified as hard constraints because they had to be met at all times. Table 5, similarly, demonstrated the available demand that the manufacturing plant needed to satisfy in accordance with equation (8). The appendix A1 is code in python which revealed that the demands were at least higher than the demand constraints. Similarly, it appeared that the total machine hours of 8241.97 were less than the machine hours limitation as indicated by equation (10). Additionally, labor hours satisfied the requirements according to equations (12, 13, 14, and 15). The number of batches to be generated from each product each month in accordance with the determined optimum solution is shown in appendix A2. According to the solution in Python, the monthly demand may be supplied utilizing the available resources at a minimal cost of roughly Rs.7.53 million. The facility should create the following number of batches from each product in order to accomplish that:

X_1 (Milk Puff) = 5

X_2 (Cheese Crackers) = 58

X_3 (Marie) = 20

X_4 (Twist) = 21

X_5 (Soaltee) = 92

Furthermore, the production facility can have enough hours for production of other biscuits if it runs constantly for 16 hours each day for 26 days in a particular month in order to meet client demand. The solution printed the status as optimal which means that an optimal solution was found.

3.2 Discussion

Heuristics and intuition are frequently used in manual planning processes, which results in poor decisions and ineffective resource management. These inefficiencies can cause increased operational expenses, extra inventory, and unused resources. The lack of a consistent mathematical model makes decision-making more difficult and also limits our ability to thoroughly assess alternative scenarios. If the integer constraint is shifted, the total monthly cost can be decreased by Rs. 474,705. The extra monthly cost is brought on by the extra number of biscuits that the integer constraint will cause to be created. As a result, if the company can make partial batches, the monthly additional cost of making more biscuits could be decreased.

Goal programming and batch determination approach can be used to determine the number of batches that should be generated for any month, even though this study was based on data from a specific month. Applying fresh Di's to the created Python code would provide the number of products needed each month because of Di's change from month to month. Additionally, the model can be appropriately modified to vary the working hours and working days until it becomes feasible. After determining a probability distribution for the monthly demand and simulating a scenario to account for the anticipated monthly demand, management can examine the future demand for the business. Furthermore, management still needs to establish goals, but if they do so, a goal programming model can address this issue.

The production department at Shree Pashupati Biscuit Industries Pvt. Ltd. discovered an improvement in production efficiency after implementing this integer linear programming formulation for production planning. While there was a heuristic manual and JIT planning based on the orders placed, there was no specific estimate of the required cost. Therefore, cost-saving measures were taken using other ineffective techniques. The production need was determined using the ILP approach in a matter of seconds, which aided in the analysis of the plant to determine the number of laborers and raw materials needed. The plant might save money by controlling labor costs, equipment usage, and the quantity of raw materials purchased.

4. Conclusions

The integer linear programming model provides a solution to reduce monthly manufacturing costs by addressing hard constraints such as monthly demand and soft constraints like equipment and personnel availability. By reducing wastage of time and enhancing process efficiency, labor expenses are decreased, and inventory expenses are minimized. The model also increases capacity and optimizes equipment consumption, thereby increasing punctuality of product and service deliveries.

Management can schedule production for each day, as the total number of batches and labor hours per month are known in advance. Overproduction of biscuits can lead to excessive inventory costs and damage the production facility's reputation. This production plan ensures that neither physical nor human resources are wasted, allowing manufacturing facilities to lower production costs.

Acknowledgements

We would like to express our sincere gratitude to Shree Pashupati Biscuit Industries Pvt. Ltd. for facilitating with data. Their assistance and support greatly enhanced the depth and quality of the outcomes.

References

Afolalu, S. A., Matthew, B. B., Ongbali, S. O., Abdulkareem, A., Emetere, M. E., & Iheanetu, O. U. (2021). Overview impact of planning in production of a manufacturing sector. *IOP Conference Series: Materials*, *1036*. doi:10.1088/1757-899X/1036/1/012060

Afriansyah, A., & Mohruni, A. S. (2021). Production planning and control system with just in time and lean production: a review *Journal of Mechanical Science and Engineering*, *6*(2), 019-027. doi:https://doi.org/10.36706/jmse.v6i2.31

Alden, A., Mohammed, A., Mohamed, S. A., & Ahmad, M. (2023). Solving aggregate production planning problem with uncertainty using fuzzy goal programming. *International Journal of Mathematics in Operational Research*, *25*(4). doi:10.1504/IJMOR.2022.10049130

Cheraghalikhani, A., Khoshalhan, F., & Mokhtari, H. (2018). Aggregate production planning: A literature review and future research directions. *International Journal of Industrial Engineering Computations*, *10*(2), 309-330. doi:10.5267/j.ijiec.2018.6.002

Christefa, D., Mawengkang, H., & Zarlis, M. (2022). Data-Driven Decision Making In Large Scale Production Planning. *Sinkron : Jurnal dan Penelitian Teknik Informatika*, *6*(3). doi:10.33395/sinkron.v7i3.11600

E.E., A., & A, E. A. (2020). A mathematical model for solving integer linear programming problems. *African Journal of Mathematics and Computer Science Research*, *13*(1), 39-50. doi:10.5897/AJMCSR2019.0804

Hamal, V., Parmar, R., & Patel, T. (2023). A research paper on study of production, planning and ctrolling in BIPICO Pvt. Ltd. *International Research Journal of Modernization in Engineering Technology and Science*, *05*(02). doi:https://www.doi.org/10.56726/IRJMETS33639

Hojati, M., & Patil, A. S. (2011). An integer linear programming-based heuristic for scheduling heterogeneous, part-time service employees. *European Journal of Operational Research*, *209*(1), 37-50. doi:https://doi.org/10.1016/j.ejor.2010.09.004

Jamalnia, A., & Soukhakian, M. A. (2008). A hybrid fuzzy goal programming approach with different goal priorities to aggregate production planning. *Computers & Industrial Engineering*, *56*(4). doi:10.1016/j.cie.2008.09.010

Jamalnia, A., Yang, J.-B., Feili, A., Xu, D.-L., & Jamali, G. (2019). Aggregate production planning under uncertainty: a comprehensive literature survey and future research directions. *The International Journal of Advanced Manufacturing Technology*, *102*, 159-181.

Khezrimotlagh, D., & Chen, Y. (2018). Production Planning Problem. In *Decision Making and Performance Evaluation Using Data Envelopment Analysis. International Series in Operations Research & Management Science* (Vol. 269). Springer, Cham. doi:https://doi.org/10.1007/978-3-319-76345-3_9

Li, X., Guo, S., Liu, Y., Du, B., & Wang, L. (2017). A Production Planning Model for Make-to-Order Foundry Flow. *Mathematical Problems in Engineering*, 1-15. doi:https://www.doi.org/10.1155/2017/6315613

MacDonald, Z. (1995). Teaching Linear Programming using Microsoft Excel Solver. Computers in Higher Education Econmics Review, *9*(3).

Mitchell, S., O'Sullivan, M., & Dunning, I. (2011). PuLP: A Linear Programming Toolkit for Python. *Optimization Online*. Retrieved from https://optimization-online.org/?p=11731

Nash, J. (2000). The (Dantzig) simplex method for linear programming. *Computing in Science & Engineering*, *2*(1), 29-31. doi:10.1109/5992.814654

Neto, J. (2012). A simple finite cutting plane algorithm for integer programs. *Operations Research Letters*, *40*(6), 578-580. doi:10.1016/j.orl.2012.10.003

Oladejo, N. K., Abolarinwa, A., Salawu, S. O., Lukman, A. F., & Bukari, I. H. (2019). Optimization principle and its application in optimizing landmark university bakery production using linear programming. *International Journal of Civil Engineering and Technology*, *10*(02), 183-190.

Schrijver, A. (1999). *Theory of linear and integer programming*. A Wiley-Interscience Publication.

Solaja, O. A., Abiodun, J. A., Abioro, M., Ekpudu, J. E., & Olasubulumi, O. (2019). Application of linear programming in production planning. *International Journal of Applied Operational Research*, *9*(3), 11-19.

Wang, R. C., & Fang, H. H. (2001). Aggregate production planning with multiple objectives in a fuzzy environment. *European Journal of Operational Research*, *133*(3). doi:10.1016/S0377-2217(00)00196-X

Wolsey, L. A. (1998). *Integer Programming*. A Wiley Interscience Publication.

Appendix 1: Python code to solve ILP and results

```
from pulp import *

# Create the 'prob' variable to contain the problem data
prob = LpProblem("Integer Linear Programming Problem", LpMinimize)

# Create decision variables
x1 = LpVariable("x1", 0, None, LpInteger)
x2 = LpVariable("x2", 0, None, LpInteger)
x3 = LpVariable("x3", 0, None, LpInteger)
x4 = LpVariable("x4", 0, None, LpInteger)
x5 = LpVariable("x5", 0, None, LpInteger)

# Objective function
prob += 41557.69*x1 + 40458.99*x2 + 35380.22*x3 + 40936.47*x4 + 37024.96*x5, "Total Cost"

# Hard constraints
prob += 296.84*x1 >= 1368
prob += 299.69*x2 >= 17280
prob += 276.41*x3 >= 5400
prob += 310.12*x4 >= 6480
prob += 330.58*x5 >= 30240

# Soft constraints
prob += 40*x1 + 50*x2 + 45*x3 + 39.29*x4 + 37.14*x5 <= 24960
prob += 200*x1 + 250*x2 + 225*x3 + 196.45*x4 + 185.7*x5 <= 124800
prob += 240*x1 + 300*x2 + 270*x3 + 235.74*x4 + 222.84*x5 <= 149760
prob += 280*x1 + 350*x2 + 315*x3 + 275.03*x4 + 259.98*x5 <= 174720
prob += 1800*x1 + 1900*x2 + 1800*x3 + 1768.05*x4 + 1485.6*x5 <= 1248000

# The problem is solved using PuLP's choice of Solver
prob.solve()

# The status of the solution is printed to the screen
print("Status:", LpStatus[prob.status])

# Each of the variables is printed with its resolved optimum value
for v in prob.variables():
    print(v.name, "=", v.varValue)

# The optimised objective function value is printed to the screen
print("Total Cost = ", value(prob.objective))
```

Results:

Status: Optimal

x1 = 5.0 x2 = 58.0

x3 = 20.0 x4 = 21.0

x5 = 92.0

Total Cost = 7527976.46