DOI: 10.24996/ijs.2025.66.6.21

ISSN: 0067-2904

Mathematical Optimization Strategies for address Production Mixing Challenges

Hiba A. Ahmed[1]*, Ahmed Hasan Alridha[2]

[1] *Department of Mathematics, College of Science, University of Baghdad, Iraq*
[2] *Department of Mathematics, Ministry of Education, Directorate of Education, Babylon, Iraq*

Received: 29/3/2024 Accepted: 18/7/2024 Published: 30/**6**/2025

Abstract

The mixing problem is one of the most important problem in optimization strategies that falls within a wide range of applications, especially productivity ones, which makes it face many challenges. The problem is to determine how much of each item should be obtained and mixed with others so that the properties of the mixture fall within specified parameters and the total cost is minimized and the total profit is maximized. In this paper, we presented the problem modelling according to the linear optimization technique to obtain optimal results, which was implemented using the Python Pulp library. The proposed approach shows its effectiveness through advanced computational analysis in achieving compliance with the specified standards and obtaining maximum profit margins.

Keywords: Mathematical modelling, Optimization technique, mixing (blending) problem, linear programming, python pulp's functions.

استراتيجيات التحسين الرياضي لمواجهة تحديات خلط الإنتاج

هبة عبد الله احمد[1]*, احمد حسن ال رضا[2]

[1]قسم الرياضيات، كلية العلوم، جامعة بغداد، بغداد، العراق

[2]قسم الرياضيات، وزارة التربية، مديرية التربية، بابل، العراق

الخلاصة

تعد مشكلة الخلط من أهم المشكلات في استراتيجيات التحسين والتي تقع ضمن نطاق واسع من التطبيقات وخاصة الإنتاجية منها مما يجعلها تواجه العديد من التحديات. تكمن المشكلة في تحديد الكميات التي يجب الحصول عليها من كل عنصر وخلطها مع العناصر الأخرى بحيث تقع خصائص الخليط ضمن معايير محددة ويتم تقليل التكلفة الإجمالية وتعظيم إجمالي الربح. في هذا البحث، قمنا بعرض نمذجة المشكلة وفق تقنية التحسين الخطي للحصول على النتائج المثلى، والتي تم تنفيذها باستخدام مكتبة Python Pulp. يظهر النهج المقترح فعاليته من خلال التحليل الحسابي المتقدم في تحقيق الالتزام بالمعايير المحددة والحصول على الحد الأقصى من هوامش الربح.

* Email : amqa92@yahoo.com

1. Introduction

The mixing problem has been and continues to be a fundamental challenge in various fields of the real world, especially manufacturing fields, where it requires the wise distribution of raw materials to obtain formulations to produce optimal products. Mixed Integer Linear Programming was commonly used to solve mixing difficulties (MILP). It entails combining numerous resources or commodities to make one or more goods that meet demand [1-4]. It is critical in nearly every industry, particularly natural resources, to make the best judgments feasible in order to maximize the value of each project by optimizing the revenue created by the required resources being expended. Operations research is a discipline in which real-world problems are quantitatively characterized and optimized in order to improve decision-making [5-7]. Linear programming (LP) is a must-have tool for dealing with optimization difficulties, [8]. This technique is a mathematical process for evaluating linear models under particular limitations, and in this case, to discover optimal mining production results, [9-14]. Many scholars employed various algorithms to define and solve the problem in order to get the best solution. They also presented algorithms for identifying the essential constraints for LP models. LP problems with practical applications in a wide variety of fields are identified. Several researchers have updated their findings in recent years. LP optimization is used by many algorithms in various fields, particularly mining and production engineering, to cut costs and boost revenues in mining operations. In view of the importance of what was previously mentioned, it is clear to think about using the optimization technique to obtain the desired results with the least amount of error. To achieve this end, programming languages that have amazing capabilities in calculating parameters must be used and avoid values that do not give optimal results, [15-18]. So, from this concept, the Python programming language was used to solve the mixing problem to extract the quantities accurately and give the amount of profit according to the planned production quantity. Therefore, this paper seeks to provide a precedent for future endeavour's that could be similar to it in revealing unexplored avenues that fall within important areas, including optimizing manufacturing, paving the way for entry into enhancing efficiency and achieving profitability.

2. Problem statement

Blending (Mixing) several components (or commodities or materials) to create one or more products corresponding to a demand, [19-23], for example:

1. A metal blending (mixing some metals to form an Alloy).
2. A set of oil blending (combining different types of crude oil to form a gasoline).
3. A food blending (mixing different kinds of oil to make final product).

The Problem is to determine how much of each commodity should be purchased and blended with the rest so that the characteristics of the mixture lie within specified bounds and the total cost (total profit) is minimized (maximized), see Fig.1. Mathematically, the optimization problem can be expressed as follows:

Let n, m represented as types of raw materials and products respectively. Also, the types of raw materials are indicated as i where $i = 1,2,..,n$ and the products as j where $j = 1,2,..,m$. While the variables will be of two types, x_{ij} and y_i, where x_{ij} denotes the quantity of raw material i used in product j and y_i denotes to the quantity of product j generated. As for the parameters, there will be two types in the confrontation: c_i which is the cost per unit of raw material i and h_i which is hardness of raw material i. Therefore, the optimization problem will be formulated according to the following model:

$$\text{Maximize} \sum_{j=1}^{m} f_j\, y_{j} - \sum_{i=1}^{n} c_j \sum_{j=1}^{m} x_{ij}.$$

Subject to:
$\sum_{j=1}^{m} x_{ij} \leq V_i$ (Raw material).
$y_i \leq E_i$ (Product demand).
$\sum_{i=1}^{n} h_i x_{ij} \leq Q_i$ (Quality).
$x_{ij}, y_j \geq 0$, where $i = 1,2,\dots,n$ and $j = 1,2,\dots,m$.
And f_j denote as the profit per unit of product j.

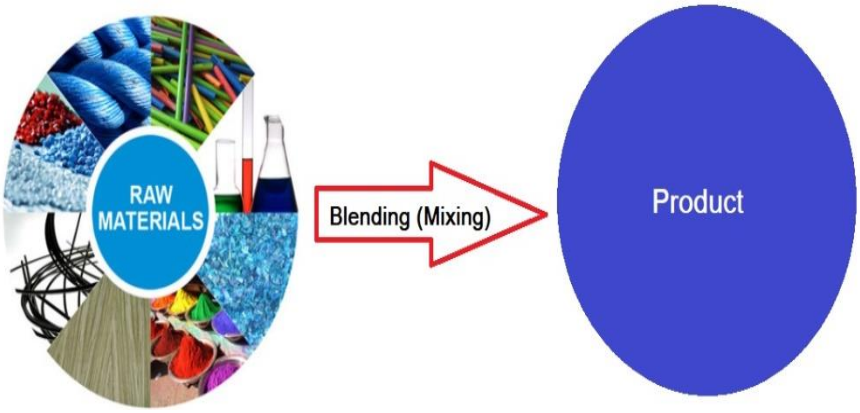

Figure 1: From raw materials, blending (mixing) to final product.

3. Methodology

According to the concept of linear programming, it is possible to formulate a model of the problem so that it can meet all the requirements within the scope of the required conditions as follows:

Index:

- i = Raw material (i.e. Raw oil types)
- j = Production lines

Model Parameters:

- $Capj$ = The refining capacity of the jth production line
- ci = The raw material's unit ith price
- p = Unit price of the final product
- ai = The amount of hardness ith of a raw material
- a = Minimum final product hardness required
- $\bar{a}$ = Maximum final product hardness required

Decision variables:

xi = The quantity of the ith raw ingredient.
y = The quantity of the final product.

The optimization problem:

$\boldsymbol{Maximize}$: $py - \sum i\ ci\ xi$.
Subject to:
$\sum i\ xi \leq Capj\ \forall\ j$ (Capacity constraint).
$\sum i\ aixi - a\ y \geq 0$ (Hardness constraint–1).
$\sum i\ aixi - \bar{a}\ y \leq 0$ (Hardness constraint–2).
$\sum i\ xi = y$ (Equality constraint)
$xi \geq 0\ \forall\ i,\ y \geq 0$

4. From manufacturing to the linear programming

Using mathematical optimization methods to maximize profit and optimize resource use in manufacturing industries is a very important and mathematically valid goal. The model is formulated to maximize efficiency by using mixing five types of raw materials to produce a single product according to specific costs and hardness values. In this paper, manufacturing a type of fuel requires the refining and merging of crude oil bearing in mind that crude oil here is divided into two categories (Brent crude and West Texas crude) as follows:

Table 1: Components of the fuel to be manufactured.

Brent crude	West Texas crude
BRE 1	WES 1
BRE 2	WES 2
	WES 3

Brent crude and West Texas crude require distinct refining lines. Assuming It is not possible to refine more than 400 barrels of Brent crude and 500 barrels of West Texas crude in any one day. The final product has a technological hardness limitation. This must be between 3 and 6 in the units used to assess hardness. Hardness is supposed to blend linearly. The required fuel cost (per barrel) is calculated according to the data shown in the following table:

Table 2: The cost and Hardness of each components.

Types	Cost	Hardness
BRE 1	110$	8.8
BRE 2	120$	6.1
WES 1	130$	2.0
WES 2	110$	4.2
WES 3	115$	5.0

The finished product costs 120$ a barrel. The goal is to maximize the net profit of the fuel production company. Therefore, the role of the improvement approach to achieve the desired goal based on the proposed formulation.

4.1 The linear model:

Maximize: $Z = 120y - 85x_1 - 91x_2 - 95x_3 - 85x_4 - 90x_5$
Subject to:
$x_1 + x_2 \leq 400$ "Capacity for Refining "Constraint 1"
$x_3 + x_4 + x_5 \leq 500$ "Capacity for Refining "Constraint 2"
$7.8x_1 + 5.1x_2 + 1.5x_3 + 3.5x_4 + 4.0x_5 - 2y \geq 0$ "Determine the hardness of the final product Constraint-1"
$7.8x_1 + 5.1x_2 + 1.5x_3 + 3.5x_4 + 4.0x_5 - 5y \leq 0$ "Determine the hardness of the final product Constraint-2"
$x_1 + x_2 + x_3 + x_4 + x_5 - y = 0$ "The amount of the final product should be equal to the amount of ingredients."
$x_i \geq 0\ \forall i, y \geq 0.$

4.2 Solving the problem by python with pulp functions:

In order to start the blending problem model with python, we will import all of the pulp module's classes and functions to construct a new model object with the Model class and then identify the raw of material types:

```
from pulp import *
# List (Type of Rawmaterials)
Rawmaterialtypes = ["BRE1", "BRE2", "WES1", "WES2", "WES3"]
Products = "Y"
# Paramters and Data
cost = {"BRE1":85,"BRE2":91,"WES1":95, "WES2":85, "WES3":90}
Hardness = {"BRE1":7.8,"BRE2":5.1,"WES1":1.5, "WES2":3.5, "WES3":4.0}
# Setting the Problem
prob = LpProblem("Blending Problem", LpMaximize)
```

Decision variables: According to the optimization strategy, the technology must be of two parts, the first is to identify the decision variables and then formulate the model as follows:

1. x_j = Quantities (barrels) of BRE1, BRE2, WES1, WES2, WES3 (i.e. raw material) that must be purchased, refined, and blended in a single day
2. y = The amount of product to be made

In this step, we will define the decision variables. In our problem, we have two variables ingredient-1 and ingredient-2. Let's create them using LpVariable.dicts.

```
# Desicion Variables
x_var = LpVariable.dicts("RawMaterial", Rawmaterialtypes, 0, None)
y_var = LpVariable.dicts("Product", Products, 0, None)
# Objective Function
prob += lpSum(120*y_var[i] for i in Products) - lpSum(cost[i]*x_var[i] for i in
Rawmaterialtypes)
```

Define the constraints and solving the model: The term constraint refers to the restriction on the values of the decision variables. A linear constraint, for example, stipulates that a linear expression on a set of variables must have a value that is less-than-or-equal, greater-than-or-equal, or equal to another linear expression. Finally, solving the model by call the optimize method to address the LP problem. Using the for loop below, we can output the final value.

```
# Constraints
prob += lpSum(x_var[i] for i in Rawmaterialtypes[:2])<=400
prob += lpSum(x_var[i] for i in Rawmaterialtypes[2:])<=500
prob += lpSum(Hardness[i]*x_var[i] for i in Rawmaterialtypes) - lpSum(2*y_var[i] for i in
Products) >= 0
prob += lpSum(Hardness[i]*x_var[i] for i in Rawmaterialtypes) - lpSum(5*y_var[i] for i in
Products) <= 0
prob += lpSum(x_var[i] for i in Rawmaterialtypes) - lpSum(y_var[i] for i in Products) == 0
prob.solve()
print("Solution Status = ", LpStatus[prob.status])
# Print the solution of the Decision Variables
for v in prob.variables():
   if v.varValue>0:
      print(v.name, "=", v.varValue)
# Print Optimal
print("Total Profit = ", value(prob.objective))
```

5. Results and discussion

The current optimization model has proven effective in maximizing profits according to the objective function and meeting potential constraints by mixing different raw materials. After running the optimization strategy using the Python language, the results shown in Table 3 were obtained. the optimal objective value with time is 30677.77778 - 4 iterations time 0.002 and the state of the solution according to what has been prepared is optimal. In addition, the quantity of production Y is 900 barrels and the amount of raw materials to be used (RawMaterial_BRE1, RawMaterial_BRE2, RawMaterial_WES2) are (262.96296, 137.03704, 500.0) barrels respectively so we can get the total profit 30677.77776000001.

Table 3: Optimal Production Quantities.

Target	Result
Optimal - objective value	30677.778
Optimal objective with time	30677.77778 - 4 iterations time 0.002
Solution Status	Optimal
Product_Y	900
RawMaterial_BRE1	262.96296
RawMaterial_BRE2	137.03704
RawMaterial_WES2	500.0
Total Profit	30677.77776000001

Figure 2 shows the distribution of raw materials, including the required optimal limits, in addition to the target product. The colored bars depict the allocations of raw materials required to produce the targeted product, showing the dominance of some raw materials over others in the production process. We also imagine the main substance that had a significant impact on the production of product Y, which was represented by substance WES2, while the role of the other substances BRE1 and BRE2 comes as complementary roles in achieving the optimal mix for production.

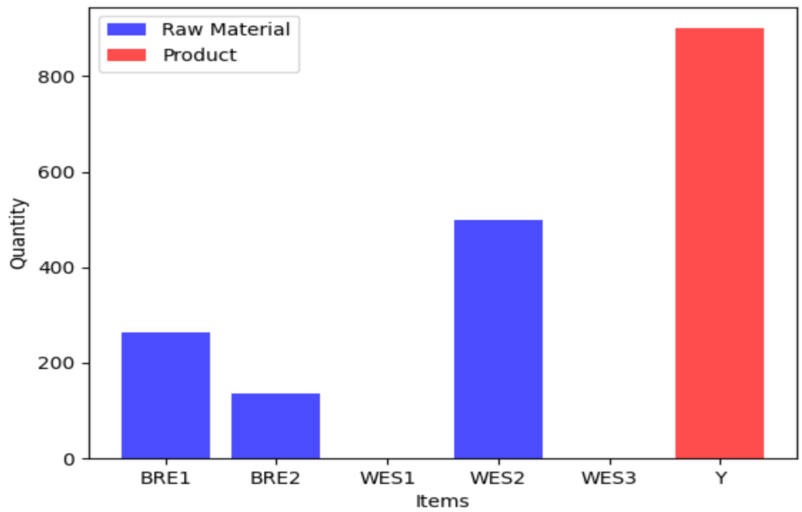

Figure 2: Optimal Production Quantities.

Figure. 3 provides compelling evidence in verifying the feasibility of the solution, as it shows compliance with the required hardness restrictions according to the target product. It shows the hardness values calculated for the assumptions of the final quantities of optimal raw materials and how these hardness restrictions are met within the specified range.

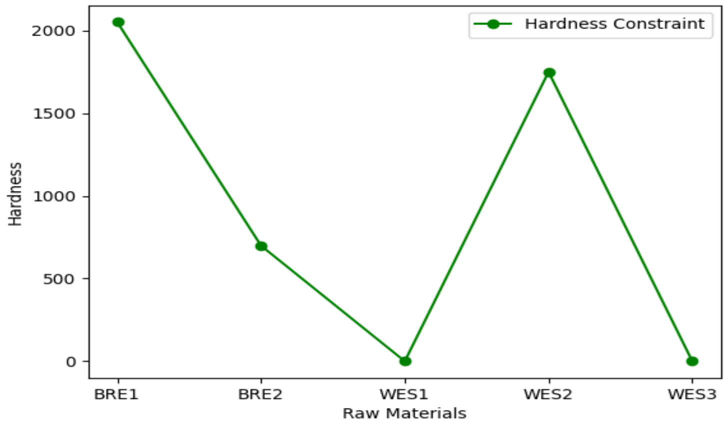

Figure 3: Hardness Constraint.

Finally, the results obtained mathematically confirm that the optimal solution not only increases profitability, but that the methodology also demonstrates strict adherence and compliance with the basic constraints, and thus we have achieved the desired goals of the optimization process.

6. Conclusions

The mixing problem has been investigated in detail, starting with the formulation of the problem and ending with the implementation using the Python pulp's library. Actually, by harnessing the computational power of the Python language, the effective capabilities of linear programming methodologies in facing complex challenges have been proven, making many leaps forward in the field of manufacturing improvement. In addition, the mixing problem was employed according to our approach, which is related to obtaining a total net profit without dispersion according to the requirements, which in our research was related to mixing raw materials from oil to obtain our product. The amount that must be obtained from each element and mixed with other elements has been determined so that the properties of the mixture fall within the required standards according to the objective function. Finally, we solved the mixing problem in Python using linear programming as we reached the optimal results in the least time and the highest possible value. Future work could include several directions in exploring the optimization process. For example, we mention the constraints of sustainability and dynamic cost, in addition to the supply chain. Implementing an improvement approach in these areas leads to the possibility of obtaining a more robust, adaptable and efficient improvement model, which results in achieving the desired goals of maximizing profit while ensuring quality and sustainability of production processes.

References

[1] W. T. Lunardi, E. G. Birgin, P. Laborie, D. P. Ronconi, and H. Voos, "Mixed Integer linear programming and constraint programming models for the online printing shop scheduling problem," Comput. Oper. Res., vol. 123, no. 105020, pp. 105020, 2020.

[2] I. X. Tassopoulos, C. A. Iliopoulou, and G. N. Beligiannis, "Solving the Greek school timetabling problem by a mixed integer programming model," J. Oper. Res. Soc., pp. 1–16, 2019.

[3] G. Greivel, A. Newman, M. Brown, and K. Eurek, "Improving Mathematical Exposition of an Industrial-Scale Linear Program," INFORMS Transactions on Education, vol. 24, no. 2, pp. 119–135, 2024.

[4] M. M. Hassan, S. A. Bader, M. A. Ali, W. R. Abdellah, and G. S. Abdelhaffez, “Linear programming as a tool to design the mix of cement plant raw materials,” Mining Geol. Pet. Eng. Bull., vol. 37, no. 4, pp. 109–117, 2022.

[5] A. M. Ramadan, A. M. Saleh, T. A. Taha, and M. R. Moharam, “An attempt to improve mechanical properties of brick produced from El-Maghara coal washing plant waste,” Physicochemical Problems of Mineral Processing, vol. 35, pp. 153–160, 2001.

[6] S. Maher, M. Miltenberger, J. P. Pedroso, D. Rehfeldt, R. Schwarz, and F. Serrano, “PySCIPOpt: Mathematical programming in python with the SCIP optimization suite,” in Mathematical Software – ICMS 2016, Cham: Springer International Publishing, 2016, pp. 301–307.

[7] G. Paredes-Belmar, A. Lüer-Villagra, V. Marianov, C. E. Cortés, and A. Bronfman, “The milk collection problem with blending and collection points. Computers and electronics in agriculture,” vol. 134, pp. 109–123, 2017.

[8] Y. Xie, A. Neumann, and F. Neumann, “Heuristic strategies for solving complex interacting stockpile blending problem with chance constraints,” arXiv [cs.NE], 2021.

[9] G. Ntourmas, F. Glock, F. Daoud, G. Schuhmacher, D. Chronopoulos, and E. Özcan, “Mixed Integer Linear Programming formulations of the stacking sequence and blending optimisation of composite structures,” Compos. Struct., vol. 264, no. 113660, pp. 113660, 2021.

[10] D. A. Abduljabbar, “Community Detection in Modular Complex Networks Using an Improved Particle Swarm Optimization Algorithm,” Iraqi Journal of Science, vol. 64, no. 8, pp. 4228–4243, 2023.

[11] P. Scardaoni, M. Montemurro, M. Panettieri, and E. Catapano, “New blending constraints and a stack-recovery strategy for the multi-scale design of composite laminates,” Structural and Multidisciplinary Optimization, vol. 63, no. 2, pp. 741–766, 2021.

[12] S. Peng, F. Maggioni, and A. Lisser, “Bounds for probabilistic programming with application to a blend planning problem,” Eur. J. Oper. Res., vol. 297, no. 3, pp. 964–976, 2022.

[13] S. Jarernsuk and B. Phruksaphanrat, “Solving tea blending problems using interactive fuzzy multi-objective linear programming,” Processes (Basel), vol. 11, no. 1, pp. 49, 2022.

[14] F. H. Ketafa and S. Al-Darraji, “Path Planning for Autonomous Mobile Robots Using the RFO-GWO Optimization Algorithm,” Iraqi Journal of Science, pp. 1070–1088, 2024.

[15] N. S. Hassan and Z. Al-Khafaji, “Determine the levels of importance of units in the reliability network,” in 2023 6th International Conference on Engineering Technology and its Applications (IICETA), 2023.

[16] A. Alridha, A. M. Salman, and A. S. Al-Jilawi, “The Applications of NP-hardness optimizations problem,” J. Phys. Conf. Ser., vol. 1818, no. 1, pp. 012179, 2021.

[17] M. Taifouris, M. Martín, A. Martínez, and N. Esquejo, “Challenges in the design of formulated products: multiscale process and product design,” Curr. Opin. Chem. Eng., vol. 27, pp. 1–9, 2020.

[18] D. J. Presser, D. C. Cafaro, I. E. Grossmann, P. Misra, and S. Mehta, “Mathematical programming model for the optimal management of carbon intensity indicators in global supply chains,” Comput. Chem. Eng., vol. 182, no. 108546, pp. 108546, 2024.

[19] A. Ghosh and R. K. Verma, “APPLICATION OF MATHEMATICAL PROGRAMMING TECHNIQUES IN PRODUCTION PLANNING INTHE PETROLEUM INDUSTRY,” International Journal of Futuristic Innovation in Engineering, vol. 2, no. 2, pp. 302–314, 2023.

[20] M. A. Hannan et al., “Solid waste collection optimization objectives, constraints, modeling approaches, and their challenges toward achieving sustainable development goals,” J. Clean. Prod., vol. 277, no. 123557, pp. 123557, 2020.

[21] D. J. Fiorotto, R. Jans, and S. A. de Araujo, “Integrated lot sizing and blending problems,” Comput. Oper. Res., vol. 131, no. 105255, pp. 105255, 2021.

[22] H. A. Mueen and M. A. K. Shiker, “A new projection technique with gradient property to solve optimization problems,” J. Phys. Conf. Ser., vol. 1963, no. 1, pp. 012111, 2021.

[23] B. Ivanov, P. S. Stanimirović, G. V. Milovanović, S. Djordjević, and I. Brajević, "Accelerated multiple step-size methods for solving unconstrained optimization problems," Optim. Methods Softw., vol. 36, no. 5, pp. 998–1029, 2021.