TABLE OF CONTENTS

CHAPTER

1	**Classification of Optimization Problem**	**3**
	1.1 Linear Programming	3
	1.2 Quadratic Programming	4
	1.3 Non-Linear Programming	4
	1.4 Integer Programming	5
	1.5 Mixed Integer Linear Programming	6
	1.6 Mixed Integer Non-Linear Programming	7
	1.7 Direct Search	8
	1.8 Stochastic Search	8
2	**Application of Optimization Method in Chemical Engineering Industry**	**10**
	2.1 Optimization Problem Applying Mixed Integer Linear Programming	10
	2.2 Linear Programming- Application of Linear Programming in Enrichment of Protein in Bioreactors	11
	2.3 Mixed Integer Non-Linear Programming- Problem of Trimming Encountered in the Paper Industry	12
	2.4 Quadratic Programming- Optimization of Chemical Reactor to Produce Chemical Product	13
	2.5 Nonlinear Programming- Solvent Extraction of Jatropha Curcas Seed Oil for Biodesiel Production	15
3	**MATLAB Application in Optimization**	**17**
	3.1 Linear Programming Model for Optimizing Product Mix & Machine Processing Time in a Fast Food Restaurant: A Case Study of Ostrich Bakery –‘linprog’	17
	3.2 Wastewater treatment for water quality management for rivers affected by phosphorus- *‘quadprog’*	18
	3.3 Optimization in the Production of Stainless Steel- ‘intlinprog’	20
	3.4 Production of Thermoplastic in a Chemical Reactor- ‘fmincon’	23
	3.5 Optimization of Biological Wastewater Treatment Plant – ‘fminunc’	26
	3.6 Script File and Short Video Summary	28
REFERENCES		**28**

Chapter 1
Classification of Optimization Problem

1.1 Linear Programming *[Choong Yi Jie]*

Objective Function:

A linear equation used to represent the goal of the optimization problem. It is a combination of decision variables, in order to be maximized or minimized while adhering the given constraints. The objective function can be used to represent different quantities relevant to the problem such as time, cost and profit. It is crucial in determining the optimal values of decision variables that achieve the desired goal while satisfying the constraints. The function can be written as below:

$$\text{Min } Z = 9X + 5Y,$$

where the Z is the function to be optimized, with the X and Y decision variables.

Solving Method:

Various methods can be used such as graphical method, simplex method, OpenSolver and R software (Vedantu, 2020).

i. Graphical Method
- Involves plotting the inequalities on an X-Y coordinate plane to obtain the feasible region and the best solution.
- Limited to the problems with two or three decision variables, as it is impossible to graphically illustrate more than three dimensions.

ii. Simplex Method
- Manually solve technique.
- Powerful and widely used algorithms and it involves modifying the values of fundamental variables iteratively to achieve the highest value for the objective function.
- Use tableaus and pivot variables to find the optimal solution.
- Each constraint inequality and decision variable should be non-negative.

iii. R software and OpenSolver
- Open-source data science software.
- Simplify linear programming and allow us to find the optimal solution in a few steps.

Table 1.1: Advantages and Disadvantages of Linear Programming

Advantages	**Disadvantages**
• Achieve optimal utilization of resources & can lead to wise resource allocation decisions. • Improve the quality of decision-making by providing an objective approach. • Allows for the re-evaluation of plans to adapt to changing conditions. • Simplex method is an efficient algorithm for solving complex LP problems with many variables.	• Limited to addressing only 1 objective, whereas real life problems often have multiple objectives. • Time and uncertainty are not considered in LP, which can limit its applicability in certain situations. • Parameters in the model are assumed to be constant, which may not be the case in real life circumstances. • Does not guarantee integer solutions, which may be important in certain applications. • Graphical method is limited to 2 variables.

1.2 Quadratic Programming *[Choong Yi Jie]*

Objective Function:

A multivariable quadratic function that used to be maximized or minimized. It can be subjected to multiple linear equality or inequality constraints. It can include a set of decision variables that are either bound or unbound. It may contain bilinear or up to second polynomial terms.

Minimize $f(x,y) = x^2 + 5y^2 - 14y + 17$

Solving Method:

i. Active Set Method

- An iterative algorithm that identifies the active sets that are binding to the current solution and do not need the iterates or solution to be vertices of the feasible set, making them useful for nonconvex QPs.
- At each iteration, the active set is updated, and the problem is solved based on the reduced set of constraints.

ii. Interior Point Method

- A numerical optimization algorithm that solves a sequence of barrier problems
- The solution that violates the constraints can be handled by adding a penalty term by barrier element to the objective function.
- Compute iterates within the feasible region defined by the constraints, resulting in a continuous path to the optimal solution.

Table 1.2: Advantages and Disadvantages of Quadratic Programming

Solving Method	Advantages	Disadvantages
Active set method	• More efficient for large scale QP problems as it converges faster, and the identification of active sets is not required.	• Time consuming as it requires the solution for a sequence of barrier problems. • Require more memory.
Interior Point Method	• More recommended for problems with smaller number of constraints as the active set can be identified faster. • Easier to implement & more memory efficient.	• May not converge if the problem has degenerate constraint. • Required the identification of active sets for each iteration. • Require the use of numerical technique such as pivoting.

1.3 Non-linear Programming (NLP) *[Choong Yi Jie]*

Objective Function:

A non-linear function that involves 1 or more decision variables to represent the cost, utility or performance of a system that is being optimized. The NLP seeks to optimize each decision variable that may maximize or minimize the objective function while adhering to the constraints that might be bound, linear, non-linear and so on.

Solving Method:

i. Interior Point Method

- A numerical optimization algorithm that solves a sequence of barrier problems.
- The solution that violates the constraints can be handled by adding a penalty term by barrier element to the objective function.

- Compute iterates within the feasible region defined by the constraints, resulting in a continuous path to the optimal solution.

Table 1.3: Advantages and Disadvantages of Non-Linear Programming

Solving Method	Advantages	Disadvantages
Interior Point Method	• Can handle large scale problems with a high degree of accuracy as it converges faster. • NLP is powerful in modeling & solving complex real-world problems in different fields with high accuracy and flexibility as it can handle non-linear, non-convex & high complexity of constraints.	• Time consuming as it requires the solution for a sequence of barrier problems. • Require more memory and expertise. • May not always converge to the global optimum. It may converge to a local optimum which is not the best solution.

1.4 Integer Programming (IP) *[Choong Yi Jie]*

Objective Function:

A linear or non-linear function that represents the goal of the problem that is being either maximized or minimized, subject to constraints that restrict the feasible solutions. Integers are usually used as the variables in the objective function of IP as the solutions obtained must be integers. IP seeks the optimal solution for the objective function while meeting the constraints.

Solving Method:

i. Branch-and-bound

- The total set of feasible solutions is divided into smaller subsets, and each of the subsets is then evaluated systematically until the best solution is obtained (Dutta, 2016).

ii. Cutting plane

- The constraints are added to cut off parts of the feasible region that do not include an optimal integer solution. These constraints tighten the LP relaxation of the problem. Hence, the optimal integer solution becomes an extreme point that can be found using the simplex method (Lavrov, 2019).

iii. Branch-and-cut

- Combination of branch-and-bound and cutting planes. The algorithm starts with a branch-and-bound to distribute the solution into smaller subsets and is followed by the cutting planes to tighten the LP relaxation of the subproblem. Finally, the new linear programs are solved using the simplex method and the steps are repeated to find the optimal solution (Mitchell, 2009)

Table 1.4: Advantages and Disadvantages of Integer Programming

Solving Method	Advantages	Disadvantages
Branch-and-bound	• Can efficiently handle large problem sizes as it does not evaluate the entire region, only cover smaller subsets of feasible region.	• May converge slowly due to many branching levels. • Required more memory.

Cutting plane	• Effective in improving the quality of solution. • Help to reduce the number of branching level in branch-and-bound method.	• Increase the complexity of the LP relaxation due to additional constraints.
Branch-and-cut	• More effective and efficient as it can reduce the number of subproblems that required evaluation. • Improve the quality of solutions obtained.	• Required more memory.

1.5 Mixed-Integer Linear Programming (MILP) *[Khadija Nadim]*

Objective Function:

A mixed-integer linear program (MILP) is a problem where the objective function can be maximized or minimized when the function and its constraints are linear. The variables in MILP's can be rational, integers or binary, making the function mixed.

Solving Method:

i. **Branch and bound method**
Branch and bound method divide the set of feasible solutions into smaller subsets. The best possible solution is selected if its bounds can contain the optimal solution else it is disregarded.

ii. **Cutting plane method**
This algorithm uses LP relation process to cut off part of the feasible region making the bound of the function tighter. Thus, the optimal solution is found at the extreme points by using the simplex method.

iii. **Branch and cut method**
This method uses both the branch and bound and cutting method to find the optimal solution. After the original problem is divided into subsets, the algorithms use cutting planes to tighten the lower bounds. Higher quality cutting lead to better results.

Table 1.5: Advantages and Disadvantages of Mixed-Integer Linear Programming

Solving Method	**Advantages**	**Disadvantages**
Branch and bound	• As the algorithm does not check all different subsets the time taken to find the optimal solution is decreased. • The algorithm does not repeatedly check the same node for the solution.	• If the problem is too large the number of subsets and nodes created might be very large.
Cutting plane	• It is faster to obtain a solution as the bounds are tightened faster with this method.	• It can be time consuming as the number of constraints increases.

Branch and cut method	• The quality of the solution obtained increases with branch and cut method.	• As the number of constraints that need to be evaluated increases, computers require more memory to find results.

1.6 Mixed-Integer Non-Linear Programming (MINLP) *[Khadija Nadim]*

<u>Objective Function:</u>

Mixed integer nonlinear programming refers to when the objective function and the constraints are nonlinearly containing both discrete and continuous integer variables. The objective function is also a twice continuously differentiable function. The goal of the function is to be maximized or minimized.

<u>Solving Method:</u>

i. **Nonlinear Branch and Bound (NBB)**

This algorithm breaks down the feasible set to many subproblems thereby lowering the bounds. The goal is to create strong relaxation compared to the parent so that the result is achieved faster. The algorithm selects the best solution as the optimum solution while comparing it to other partial solutions.

ii. **Extended Cutting Plane method**

This method requires projecting cutting planes made by MILP solution points at each iteration. These constraints tighten the convex sets so that the optimal solution can be found at the intersection.

iii. **Outer-Approximation /Equality Relaxation algorithm, OA/ER**

The OA method solves MINLP problems by creating subproblems and solving them by alternating them into MILP and NLP problems. The NLP problems create an upper bound for the objective function and the MILP generates a master global linear approximation that collects linear approximations at each iteration which are relaxed as inequalities. At each iteration, the master problem creates a lower bound until it is found that the lower bound created exceeds the new objective function's upper bound.

Table 1.6: Advantages and Disadvantages of Mixed-Integer Non-Linear Programming

Solving Method	**Advantages**	**Disadvantages**
Non-linear branch and bound	• It is faster as the subsets which are not optimal are discarded.	• This method is only used for smaller networks as the number of parallel searches increases exponentially with its size needing more computer memory.
Extended cutting plane	• ECP methods do not require the use of an NLP solver only MILP solvers. As the number of cuts increases the constraints are tightened resulting in fewer subproblems that are easier to solve.	• As the number of cuts increases the computer efficiency needed to solve the MINLP problem also increases.

Outer-Approximation	• This method uses NLP for their upper bounds which gives more efficient iteration loops. • It is better at solving MINLP problems containing more continuous variables.	• Despite NLP's creating less iteration loops they also require greater computational work than using just MILP's.

1.7 Direct Search Method *[Khadija Nadim]*

Objective Function:

Direct search method is applicable when the objective function that needs to be maximized or minimized is nonlinear and has no derivatives. A direct search algorithm searches the points surrounding the objective function to find one that is lower in value than the current point (Mathworks, 2013).

i. Nelder–Mead Method

This algorithm breaks down the feasible set to many subproblems thereby lowering the bounds. The goal is to create strong relaxation compared to the parent so that the result is achieved faster. The algorithm selects the best solution as the optimum solution while comparing it to other partial solutions.

ii. Hooke Jeeves Method

Hooke Jeeves method works by creating N number of linearly independent search directions for an N-dimensional problem. The algorithm starts at some point and at every iteration moves across these directions only to find the optimum solution.

Table 1.7: Advantages and Disadvantages of Direct Search Method

Solving Method	**Advantages**	**Disadvantages**
Nelder–Mead	• This method produces better results in only a small number of iterations.	• The method typically requires only one or two function evaluations per iteration.
Hooke Jeeves	• As there are only two points that are evaluated at every iteration, this algorithm requires less storage.	• As the algorithm searches the vicinity surrounding every point, this requires high number of functions to be evaluated. • If the variables are nonlinear the algorithm may converge to a wrong solution. So the algorithm needs to be run a few times to get an accurate answer.

1.8 Stochastic Method [*Khadija Nadim]*

Objective Function:

Stochastic optimization refers to maximizing or minimizing a nonlinear function where there are random variables present in the objective or the constraints. Randomness in the optimization creates noise that the algorithm needs to sort through to find the global maximum or minimum (Brownlee, 2021).

i. Iterated Local Search

This method looks for multiple local optima and then searches the neighborhood of that point to find a better local optimum. It is based on the principle that the best local optimum is located close to the other local optima. To avoid getting stuck in a neighborhood, the algorithm restarts with a random point after every iteration so that new points can be obtained, giving better results (Brownlee, 2021).

ii. Stochastic Gradient Descent

In this method many samples, called batches are selected from the whole data set. From here the gradient for each sample is found at each iteration. This batch is then randomly shuffled in each iteration until the local minimum is found.

Table 1.6: Advantages and Disadvantages of Stochastic Method

Solving Method	**Advantages**	**Disadvantages**
Iterated Local Search	• As the algorithm does not search the entire space of points, it is very efficient in providing very good answers.	• This method is very prone to getting stuck at local points, especially if the program is too large.
Stochastic Gradient Descent	• Even though this method is noisier than others and requires more iterations due to the presence of random batches, It is less expensive and the computational load is reduced.	• This method is only used to minimize a problem until it converges.

Chapter 2
Application of Optimization Method in Chemical Engineering Industry

2.1 Optimization Problem Applying Mixed Integer Linear Programming *[Choong Yi Jie]*

Background:

Fagersta AB, a Swedish steel company that produces high-grade steel, aims to select the appropriate ingredients, including the weight and percentage of each chemical element to create a blend that produce 25 tons of steel. The desired steel blend should consist of 5% of carbon and 5% of molybdenum by weight, which means that 1.25 tons of carbon and 1.25 tons of molybdenum are required for the production of 25 tons of steel. The primary goal of the company is to minimize the total cost of steel blend.

To create the steel blend, there are 4 ingots of steel available for purchase, with each ingot providing only one unit per purchase.

Ingot	**Weight in Tons**	**% Carbon**	**% Molybdenum**	**Cost/Ton**
1	5	5	3	$350
2	3	4	3	$330
3	4	5	4	$310
4	6	3	4	$280

Additionally, there are 3 grades of alloy steel and 1 grade of scrap steel available for purchase, with fractional amounts available for each purchase.

Alloy	**% Carbon**	**% Molybdenum**	**Cost/Ton**
1	8	6	$500
2	7	7	$450
3	6	8	$400
Scrap	3	9	$100

<u>Methodology:</u>

The Optimization Toolbox in MATLAB is used to solve this case. This problem involves decision variables that are defined using the “optimvar” function. There are 4 binary integer variables called ‘ingots’ used to indicate whether each of 4 ingots of steel are used in the production process. Each variable is defined with some constraints. The variable ‘ingots’ have a binary constraint with a lower bound of 0 and an upper bound of 1.

- Variable ‘ingots (1) = 1’, means ingots1 is purchased.
- Variable ‘ingots (1) = 0’, means ingots1 is not purchased (do not purchase).

There are 3 continuous variables called ‘alloys’ that represent the quantity in tons of 3 different grades of alloys used in the production process. There is a continuous variable called ‘scrap’ indicating the quantity of scrap steel that is used in tons. Both of them have a lower bound of 0, which means that their values cannot be negative.

```
steelprob = optimproblem;
ingots = optimvar('ingots',4,'Type','integer','LowerBound',0,'UpperBound',1);
alloys = optimvar('alloys',3,'LowerBound',0);
scrap = optimvar('scrap','LowerBound',0);
```

The optimization problem is subject to some constraints on the weight and composition of the final product. First, the cost of producing steel is based on the costs of the ingots, alloys and scrap used in the process. The weight of each ingot is stored in the “weightIngots” array whereas the cost per unit weight of each ingot is stored in the “costIngots” array. The cost per unit of each grade of alloy is stored in the “costAlloys” array while the cost per unit of scrap is stored in the scalar “costScrap”. The sum of the costs of the ingots, alloys and scrap are computed as the total cost used in the process.

```
weightIngots = [5,3,4,6];
costIngots = weightIngots.*[350,330,310,280];
costAlloys = [500,450,400];
costScrap = 100;
cost = costIngots*ingots + costAlloys*alloys + costScrap*scrap;
```

The problem optimization is set up to minimize the cost of producing steel.

```
steelprob.Objective = cost;
```

Then, the constraints on the weight and composition of the final product are set up. The total weight of the steel product must be 25 units which is stored in the scalar "totalWeight".

```
totalWeight = weightIngots*ingots + sum(alloys) + scrap;
```

The total carbon and molybdenum content of the product must be 1.25% each, which are stored in the scalar "totalCarbon" and "totalMolyb" respectively. The carbon and molybdenum content of each ingot, alloy and scrap components are also stored in arrays. The total carbon and molybdenum content of the steel product are computed as a weighted sum of these components.

```
carbonIngots = [5,4,5,3]/100;
carbonAlloys = [8,7,6]/100;
carbonScrap = 3/100;
totalCarbon = (weightIngots.*carbonIngots)*ingots + carbonAlloys*alloys + carbonScrap*scrap;
```

```
molybIngots = [3,3,4,4]/100;
molybAlloys = [6,7,8]/100;
molybScrap = 9/100;
totalMolyb = (weightIngots.*molybIngots)*ingots + molybAlloys*alloys + molybScrap*scrap;
```

The constraints in the problem are included.

```
steelprob.Constraints.conswt = totalWeight == 25;
steelprob.Constraints.conscarb = totalCarbon == 1.25;
steelprob.Constraints.consmolyb = totalMolyb == 1.25;
```

Finally, the objective function, cost and the constraints on the total weight and composition of the final product are defined. These are stored in the "steelprob" optimization problem object, which then can be solved using an optimization solver.

```
[sol,fval] = solve(steelprob);
```

```
Solving problem using intlinprog.
LP:                Optimal objective value is 8125.600000.

Cut Generation:    Applied 3 mir cuts.
                   Lower bound is 8495.000000.
                   Relative gap is 0.00%.


Optimal solution found.

Intlinprog stopped at the root node because the objective value is within a gap
tolerance of the optimal value, options.AbsoluteGapTolerance = 0 (the default
value). The intcon variables are integer within tolerance,
options.IntegerTolerance = 1e-05 (the default value).
```

```
sol.ingots
```

```
ans = 4×1

    1.0000
    1.0000
         0
    1.0000
```

```
sol.alloys
```

```
ans = 3×1

    7.2500
         0
    0.2500
```

```
sol.scrap
```

```
ans = 3.5000
```

```
fval
```

```
fval = 8.4950e+03
```

The result shows that the optimal purchasing cost to produce the steel is $8495. Ingots 1, 2 and 4 are recommended to be purchased. 7.25 tons of alloy grade 1 and 0.25 tons of alloy grade 3 as well as 3.5 tons of scrap are recommended to be purchased.

2.2 Linear Programming- Application of Linear Programming in enrichment of protein in bioreactors (ELSEVIER,2002) *[Jee Pei Qi]*

Problem Statement:
A chemical company possesses bioreactors to produce an enriched protein by solid-state fermentation using various agricultural by-products. Due to the low moisture level, only a few microorganisms, primarily yeasts, and fungi, although some bacteria have been utilized, are capable of carrying out fermentation. Thus, due to the reduced Aw in SSF bioreactors, smaller fermenters are needed to create a more concentrated product, which also lowers the energy needed for downstream processing. Due to the restriction on free water, sterilization costs are reduced, which also lowers the operating expenses required for effluent treatment. For SSF, there are three general reactor groups that each employ a different design in an effort to promote fermentation. A straightforward to more intricate protein enrichment design is used to understand how and why modifications have been made in an effort to regulate the main limiting elements in SSF. However, the company is facing a problem as the availability of labor and raw materials to produce the desired products is limited.

Methodology:

The bioprocess company has 2000 hours of labor and 5000 units of raw materials available per week. Product X requires 5 hours of labor and 4 units of raw materials (starchy material) to produce, product Y requires 3 hours of labor and 5 units of raw materials (carob pobs) to produce, and product Z requires 4 hours of labor and 3 units of raw materials (citrus wastes) to produce. The expected profit margins for products X, Y, and Z are RM20, RM18, and RM22 per unit, respectively. The company wants to determine the optimal production plan that maximizes its profit.

Decision Variable:
x=number of units of product x
y=number of units of product y
z=number of units of product z

Objective function:
Let P= Profit
Maximize P= 20X + 18Y + 22Z

Subjected to:

5x+3y+4z <=2000 --------Total labor available per week is limited to 2000 hours
4x+5y+3z <=5000 --------Total raw material available per week is limited to 5000 units
x>=0, y>=0, z>=0

The first step is to select the pivot column, as this case study is to maximize the daily profit thus choose the first column with the largest negative entry in row 0 in the initial tableau. After that, selecting the pivot column by using the first value of each equation divides the right-hand-side value of the objective function and the constraints. The smallest ratio (>=0) indicates the row is the pivot row. The third step will be developing the equation based on the pivot row and the equation created the new simple tableau. Furthermore, as this is a maximizing case, the first three steps will be repeated until there is no more negative values in row 0. Once it happens, the optimal tableau is tabulated. The last step will be substituting the optimum values of x, y, and z into the objective function to provide the best production strategy that optimizes the company's profit.

2.3 Mixed Integer Non-Linear Programming-Problem of Trimming Encountered in the Paper Industry *[Khadija Nadim]*

Problem Statement:

Consider a simple problem of trimming encountered in the paper industry. The problem is to cut the wider raw paper rolls into smaller paper rolls of different types while minimizing the paper lost and time taken for cutting. The number of paper rolls produced for each type *i*, is given by N_i. The corresponding widths of each type *i*, is B_i. Width of the raw paper roll is given by, B_{max}. The cutting pattern is given by *j* and for each roll type *i*, with a cutting pattern *j*, the number of paper rolls produced is n_{ij}. To define whether or not a certain paper roll has a cutting pattern, y_j is defined as a binary variable. The number of product rolls should not exceed N_{max}.

Objective function:

$$\min_{m_j, y_j, n_{ij}} \left\{ \sum_{j=1}^{J} \left(c_j m_j + c_j y j \right) \right\}$$

Subject to,

$$\sum_{i=1}^{i} B_i n_{ij} - B_{\max} \leq 0, \qquad j = 1,\ldots..J \qquad \text{-------------(1)}$$

$$\sum_{i=1}^{i} n_{ij} - N_{\max} \leq 0, \qquad j = 1,\ldots..J \qquad \text{-------------(2)}$$

$$y_j - m_j \leq 0, m_j - My_j \leq 0, \qquad j = 1,\ldots..J \qquad \text{-------------(3)}$$

$$N_j - \sum_{j=1}^{J} m_j n_{ij} \leq 0, \qquad j = 1,\ldots..J \qquad \text{-------------(4)}$$

$$n_{ij,m_j} \in Z^+, y_j \in \{0,1\}$$

This problem is non-convex die to the bilinear constraint (4). Therefore, we convert it to convex form. An advantage of this is that even though the number of variables increase the number of constraints will not increase as much as if they were converted to linear form. In the problem considered, the order consists of four types of product paper rolls with widths from 330 to 415 mm, totaling an amount of 38 product paper rolls. The number of different cutting patterns J is restricted to 4. The raw paper width is 1900 ram (Skrifvars et al., 2003). By using the extended plane cutting method algorithm, the equations converge to a linear equation in only 1 iteration and thus the solution can be found.

2.4 Quadratic Programming- Optimization of Chemical Reactor to Produce Chemical Product *[Tham Ting Woon]*

Quadratic programming (QP) is an optimization technique that can used to solve a wide variety of optimization problems. One chemical engineering related scenario that can solve by using quadratic programming (QP) is optimization of chemical reactor to produce chemical product. For example, reaction between acetic acid (x) and ethanol (y) to produce ethyl acetate (z). A reaction of fixed amount of acetic acid with 5 moles and ethanol with 4 moles are carried out to maximize the yield of production, ethyl acetate (Libretexts, 2022). The aim is to maximize the production rate of chemical product, z or minimize the cost of operating the reactor subject to certain constraints based on the performance of chemical reactor.

$$CH_3COOH + C_2H_5OH \rightarrow CH_3COOC_2H_5 + H_2O$$

Rate of reaction:

$$r = kC_xC_y$$

where C_x = Concentration of Reactant x
C_y = Concentration of Reactant y

To solve this optimization problems, a mathematical model equation for the chemical reactor should be developed. The quadratic objective function in this scenario signifies to maximize the production rate of chemical product, z. This nonlinear objective function represents the production rate of chemical product, z as a function of number of moles of acetic acid, x and ethanol, y. In the reaction of acetic acid and ethanol, the decision variables are the number of moles of reactants, acetic acid and ethanol, which are subject to constraints.

Objective Function:

$$max, z = 2xy - 0.5x^2 - 0.4y^2$$

Where z = production yield of ethyl acetate

x = number of moles of acetic acid

y = number of moles of ethanol

In the above equation, 2xy represent the number of moles of ethyl acetate produced. The second and third term $-0.5x^2 and -0.4y^2$represent the cost of acetic acid and ethanol. The costs are quadratic because the production yield of ethyl acetate is limited by the two reactants.

In this scenario, there are a few linear constraints that need to be considered. Constraints will impose limit on the performance of reactor. For example, total number of moles of reactants, the feed flow rate, mass balance constraint, temperature constraint and pressure constraint.

In feed flow rate constraint, the flow rate will be specified at certain range. It is a bound constraint. The minimum and maximum allowable flow rate will be determined respectively. The stoichiometric ratio of acetic acid and ethanol is 1:1. The reactant flow rate can be determined depends on the reaction condition and desired rate of reaction (Libretexts, 2023). The reactor flow rate of 1 to 2 ml/min is suitable for this reaction to ensure good mixing and heat transfer. If the flow rate is set to be too low, it will cause poor conversion of reactant to product thus leading to low yield of production in ethyl acetate.

$$F_{min} \leq F \leq F_{max}$$

In mass balance constraint, the total flow rate of the reactants must be equal to the flow rate of product. If the flow rate of reactant greater than the product, accumulation of reactant in the reactor will occur which will cause overflow. If the flow rate of product greater than the reactants, it will lead to low yield of production which cause underflow in the reactor.

$$F_A + F_B = F_C$$

In total number of moles of reactant constraint, the total number of moles of acetic acid and ethanol should be equal to 9.

$$x + y = 9$$

In temperature constraint, the temperature of the reactor will be specified at certain range. The minimum and maximum allowable temperature will be determined respectively. The suitable temperature range in this reaction is in between $30\ °C\ to\ 80\ °C$ (Calvar et al., 2007). At high temperature, the rate of reaction will increase but the selectivity of the desired product will decrease (Calvar et al., 2007). At low temperature, the rate of reaction will decrease which may cause low yield of production and low conversion. To determine the suitable temperature for a reaction, the production yield and conversion rate should be determined to ensure the reaction can fulfill the condition of high rate of reaction, high conversion and high yield of production. Besides, the fluctuation of temperature change within the range should be as low as possible to ensure the quality of product and heat transfer (Ashraf & Chowdhury,2018).

$$T_{min} \leq T \leq T_{max}$$

In pressure constraint, the pressure of the reactor will be specified at certain range. The minimum and maximum allowable pressure will be determined respectively. This reaction is usually carried out at atmospheric pressure, which is around 1 bar (Calvar et al., 2007). The factors that can affect the pressure in the reactor are the reaction kinetics and equilibrium of the reaction (Key, 2014). At high pressure, the rate of reaction will increase but the selectivity of desired product will decrease. At low pressure, the rate of reaction will decrease and may lead to low conversion and production yield. Besides, the fluctuation of pressure change within the range should be as low as possible for safety purpose and ensure the quality of product.

$$P_{min} \leq P \leq P_{max}$$

The non- negativity constraint such as the number of moles of acetic acid (x) and ethanol (y) cannot be negative value.

$$x, y \geq 0$$

To solve this optimization problem using quadratic programming, the first step is to create the objective function. The objective function in this problem is to maximize the yield of ethyl acetate. The objective function should be expressed in quadratic function. The second step is to create constraints. The constraints in this problem are total number of moles of reactants, the feed flow rate, mass balance constraint, temperature constraint, pressure constraint and non-negativity constraint. Next, a quadratic programming problem can be set up as below.

$$max, z = 2xy - 0.5x^2 - 0.4y^2$$

Subject to:

$$F_{min} \leq F \leq F_{max}$$
$$T_{min} \leq T \leq T_{max}$$
$$P_{min} \leq P \leq P_{max}$$
$$F_A + F_B = F_C$$
$$x + y = 9$$
$$x, y \geq 0$$

The optimization of chemical reactor to produce chemical product is a quadratic programming problem. Quadratic programming allows us to optimize the operation of chemical reactor to produce chemical product. Quadratic programming techniques and MATLAB can be used to solve this optimization problem by creating a suitable objective function and constraints that fulfill the case. Other methods such as barrier method, active set method, simplex method and conjugate gradient method can also be used to solve optimization problem because these methods had been combined with MATLAB and it can be found in libraries for programming languages (D. Poupon, 2023). The rate of reaction, operating conditions can be obtained from MATLAB that able to maximize the production of ethyl acetate, C subject to the constraints. After solving the above quadratic programming problem, optimal value of x and y represent the ratio needed for the reactant to maximize the yield of product. This enables chemical engineers to optimize and design the chemical reactor which can work more effectively and efficiently by achieving maximum production yield and minimize the operation cost.

2.5 Nonlinear Programming for Solvent Extraction of Jatropha Curcas Seed Oil for Biodesiel Production (Ogunleye & Omodele Eletta, 2012) *[Wong Pei Yu]*

Jatropha curcas contains numerous seeds that have a thick oil, which can be used to make candles and soap in the cosmetic industry. To extract the oil, two solvents (n-hexane and isopropanol) were used at a ratio of 1:5 (powder weight to solvent volume) and a particle size of 0.5mm to 0.75mm. This case study conducted a randomized experiment using a central composite design with three factors: solvent composition (0-100% n-hexane), time of extraction (1-5 hours), and extraction temperature (40-60℃), each with five levels. This study also had measured various properties of the extracted oil, including oil yield, specific gravity, viscosity, free fatty acid content, and iodine value. After that, the measurements used to create response equations for each property in terms of solvent composition, time of extraction and temperature. A nonlinear programming is modelled to optimize the oil yield and minimize the other four properties, according to the ASTM D6751-07b and EN 14214-2008 (E) standards for biodiesel production.

Oil yield (%), specific gravity, viscocity (centipoises), free fatty acid (FFA), (%) and iodine value (g iodine/100g) is determined to identify the extraction factors at the design centre point. The extraction factor includes solvent composition, time of extraction and extraction temperature. Table 2.5.1 shows the extraction factors at the design centre point.

Table 2.5.1: Factors symbol and centre point

Factors	Symbol	Centre point
Solvent composition (%)	x_1	50
Time of extraction (hours)	x_2	3
Extraction temperature (°C)	x_3	50

The design was created by selecting symmetrically spaced increments of variation around a central point, and the range of variation was limited to the boundaries of the factors. Table 2.5.2 shows the ratios of the variation increments for each variable around the central point.

Table 2.5.2: Experimental Increments, values, and coded levels

X_i coded level						
Factors	$\pm$Increment	-2	-1	0	+1	+2
x_1	$\pm$25.00	0.00	25.00	50.00	75.00	100.00
x_2	$\pm$1.00	1.00	2.00	3.00	4.00	5.00
x_3	$\pm$5.00	40.00	45.00	50.00	55.00	60.00

The coded X_i ratios and x_i related in the following equations:

$$X_1 = \frac{(x_1-50)}{25} \text{ (Eq. 1)}, X_2 = \frac{(x_2-3)}{1} \text{ (Eq. 2)}, X_3 = \frac{(x_3-50)}{5} \text{ (Eq. 3)}$$

Regression analysis is a way to study the relationship between different variables. In this study, regression models were made for five properties of the oil, based on three factors. The response equations is approximated by a second degree polynomial equation:

$$R_k = b_{k0} + \sum_{i=1}^{3} b_{kii} X_i + \sum_{i=1}^{3} b_{kii} X_i^2 + \sum_{i\neq j=1}^{3} b_{kij} X_i X_j \quad \text{(Eq. 4)}$$

Where,

R_1: Oil yield (%), R_2: Oil specific gravity (%), R_3: Oil viscosity at room temperature (cP),
R_4: Free Fatty Acid (%), R_5: Iodine value (g/g)

Equations 5 to 10 were used to create a nonlinear programming problem from the equation vector 4, in the following manner:

$$\text{Maximize : } R_1 = (X_i) \qquad \text{(Eq. 5)}$$

Subject to:

$R_2(X_i) \leq b_2$ (Eq. 6), $R_3(X_i) \leq b_3$ (Eq. 7), $R_4(X_i) \leq b_4$ (Eq. 8), $R_5(X_i) \leq b_5$ (Eq. 9), $-2 \leq X_i \leq +2$ (Eq. 10)

Where,

b_i: biodiesel feedstock requirements

There are 31 set of central composite design were used to fit into the response equation and formulate a nonlinear programming problem that maximizes oil yield and minimizes the other four properties, while meeting the standard requirements for biodiesel production. The problem was solved using an optimization routine from the Matlab 7.50 version of Mathworks Inc.'s optimization toolbox. The equations and constraints is shown by following:

Objective Function:

Maximize oil yield, $R_1 = 28.069 + 1.650X_1 + 0.737X_2 + 2.256X_3 - 1.078 \times 10^{-3}X_2^2$

Subject to:

Specific gravity, R_2:

$0.914 - 5.507 \times 10^{-3}X_1 - 2.227 \times 10^{-3}X_2 - 6.891 \times 10^{-3}X_3 + 8.122 \times 10^{-5}X_1X_3 \leq 0.90$

Viscosity, R_3:

$40.182 - 0.551X_1 - 0.211X_3 + 0.198X_1^2 + 8.028 \times 10^{-2}X_1X_3 \leq 42$

Free Fatty Acid (FFA), R_4

$3.024 - 0.293X_1 - 9.090 \times 10^{-2}X_3 - 3.443 \times 10^{-2}X_1X_3 \leq 2.1185$

Iodine value, R_5:

$105.809 - 0.764X_1 - 0.416X_2 - 0.710X_3 - 0.129X_1^2 \leq 120$

Values of X_1, X_2 and X_3 were obtained using the optimization toolbox in MATLAB 7.5 and the values obtained for X_1, X_2 and X_3 are 2. Substituting these values into the equations above, R_1, R_2, R_3, R_4 and R_5 can be obtained.

Result:

Table 2.5.3: Result after Optimization

Symbol	Values
X_1	2
X_2	2
X_3	2
R_1	37.3507
R_2	0.8860
R_3	39.771
R_4	2.1185
R_5	101.5130

This study has effectively shown how nonlinear mathematical programming can be used to select the best extraction conditions for jatropha oil from its seed. This approach not only maximizes oil yield through solvent extraction, but also ensures that the oil meets the requirements for use as a biodiesel feedstock. Through this optimization, the operating conditions should be 100% volume of n-hexane, 5 hours extraction time, and 60°C extraction temperature. Using these conditions, an oil yield of 37.35% can be achieved, with a specific gravity of 0.8860, viscosity of 39.7710 cp, FFA of 2.1185%, and iodine value of 101.51 g/g.

Chapter 3
MATLAB Application in Optimization

3.1 Linear Programming Model for Optimizing Product Mix & Machine Processing Time in a Fast Food Restaurant: A Case Study of Ostrich Bakery –'*linprog*' *[Choong Yi Jie]*

Background:

Linear Programming models is used in the Ostrich Bakery fast food restaurant. They are focus on the Product Mixing model and Machine Processing time model. Both of these models are used to optimize the allocation of limited resources and minimize the processing time of the products, while utilizing the production capacity of machines. There are 4 major products in this study, they are bread, meat pie, hamburger and cake. There are 6 machines including mixer, molder, rotate oven, flat oven, slicing machines and grinder.

Table 3.1: The Daily Processing Time of the Machines for Each Product

Machines	Bread	Meatpie	Hamburger	Cake	Capacity (hours/day)	Capacity (min/day)
Mixer	15	15	15	15	8	480
Molder	5	-	5	-	8	480
Rotate oven	20	-	20	-	10	600
Flat oven	30	30	-	20	10	600
Slicing machine	-	-	15	-	10	600
Grinder	-	15	-	-	4	240
Processing Time	70	60	55	35		

Objective Function:

$$\text{Minimize } T= 70x1 + 60x2 + 55x3 + 35x4$$

Where variables x1, x2, x3 & x4 are the number of bread, meat pie, hamburger and cake respectively.

Subject to constraints:

$$\text{Mixer: } 15x1 + 15x2 + 15x3 + 15x4 \geq 480$$
$$\text{Molder: } 5x1 + 5x3 \geq 480$$
$$\text{Rotate Oven: } 20x1 + 20x3 \geq 600$$
$$\text{Flat Oven: } 30x1 + 30x2 + 20x4 \geq 600$$
$$\text{Slicing Machine } 15x3 \geq 600$$
$$\text{Grinder: } 15x2 \geq 240$$
$$x1, x2, x3, x4 \geq 0$$

```
Command Window
>>  % Define the objective function
f = [70 60 55 35];

% Define the coefficients of the constraints
A = [-15 -15 -15 -15 %Mixer
        -5 0 -5 0        % Moulder
        -20 0 -20 0      % Rotate Oven
        -30 -30 0 -20    % Flat Oven
        0 0 -15 0        % Slicing Machine
        0 -15 0 0        % Grinder
        -1 0 0 0         % No negative value
        0 -1 0 0         % No negative value
        0 0 -1 0         % No negative value
        0 0 0 -1]        % No nagetive value
```

Figure 3.1.1: Objective Function and the Coefficients in this Problem

```
A =

   -15   -15   -15   -15
    -5     0    -5     0
   -20     0   -20     0
   -30   -30     0   -20
     0     0   -15     0
     0   -15     0     0
    -1     0     0     0
     0    -1     0     0
     0     0    -1     0
     0     0     0    -1

>> B = [-480 -480 -600 -600 -600 -240 0 0 0 0];
% All the constraints are in negative as the Objective Function is a Minimization Problem.
% Therefore, the symbol is switching from GREATER THAN or EQUAL TO into LESSER THAN or EQUAL TO.

% The implementation of Linear Programming Formulae.
x = linprog (f,A,B)
fx ...
```

Figure 3.1.2: Linprog Solver Application in this Problem

Linprog function is used to solve this linear programming problem. The solution is stored in variable x, which give the optimal numbers of units for each of the products as shown below.

```
% The implementation of Linear Programming Formulae.
x = linprog (f,A,B)

Optimal solution found.


x =

    4.0000
   16.0000
   92.0000
         0

fx >> % The answer for x1, x2, x3 and x4 are determined.
```

Figure 3.1.3: Optimal Solution Obtained

3.2 Wastewater treatment for water quality management for rivers affected by phosphorus- *'quadprog'* *[Jee Pei Qi]*

Background:

Due to pressure from population increase, industrial expansion, and climate change, environmental issues within river water systems (such as water pollution) have gotten substantially worse in recent years. Planning and effective river water quality management (WQM) are required to promote long-term socioeconomic growth in a watershed. The fluctuation in river water quantity and waste conveyance processes, as well as the stochasticity in climatic and hydraulic circumstances, are some of the system components that significantly impact the WQM system. Furthermore, planning and managing river water quality effectively is a difficult task that is hampered by several complications and uncertainties. In order to plan the seasonal implementation of water quality management (WQM) under a variety of uncertainties, a quadratic programming ('quadprog') model based on simulation has been created. Algal blooms, which are caused by too much phosphorus from manufacturing sectors' effluent, farming runoff, and urban rainwater, pose a hazard to the multiple downstream. The phosphorus in the river also has an impact on several lake water quality. Therefore, the company has employed quadratic programming to formulate a mathematical model to minimize the cost needed for wastewater treatments in the affected rivers with phosphorus. During the operation, there are a total of 30 rivers that need to be treated by using wastewater treatments A and B at the same time for cost and time efficiencies. In addition, the maximum operating capacity for wastewater treatment A is 15 whereas for wastewater treatment B is 20. By solving the model with 'quadprog' solver in MATLAB, the optimal solution for the distribution of wastewater treatment A and B which is able to minimize the cost of wastewater treatment for tackling the problem can be obtained.

The mathematical model can be formulated as shown below:

Decision variable:

x1= Type of wastewater treatment A
x2= Type of wastewater treatment B

Objective function to minimize the cost of wastewater treatment for phosphorus in rivers can be written as:

$$min_x = 32_{x1}^2 + 8_{x2}^2 - 2_{x1} - 10_{x2}$$

Subject to:
x1+x2=30
Boundary constraints:

$$0 \le x1 \le 15$$
$$0 \le x2 \le 20$$

Since the formulated objective function is in the form of a quadratic function and is subject to linear equality and boundary constraints, hence this is a quadratic programming problem. The MATLAB solver, ‘quadprog’ solver will be applied in this case in order to solve this quadratic programming problem. Thus, the objective function and the constraints need to be changed in the form of matrix:

$$\begin{aligned} &\text{minimize} \quad \tfrac{1}{2}\mathbf{x}^{\mathrm{T}} Q \mathbf{x} + \mathbf{c}^{\mathrm{T}} \mathbf{x} \\ &\text{subject to} \quad A\mathbf{x} \preceq \mathbf{b}, \end{aligned}$$

Figure 3.2.1: Objective Function of Quadrative Programming Model

After rewriting the formulated objective function into the form in figure 3.2.1 above, Q and c which represent the objective function terms can be rewritten as below:

$$Q = 2 * \begin{bmatrix} 32 & 0 \\ 0 & 8 \end{bmatrix}$$
$$= \begin{bmatrix} 64 & 0 \\ 0 & 16 \end{bmatrix}$$

$$c=\begin{bmatrix} -2 \\ -10 \end{bmatrix}$$

Linear equality constraints can be written below:
; where: LEC is the linear equality constraint
: BC is the boundary constraint
: LB is the lower boundary of the constraint
: UB is the upper boundary of the constraint
Thus,

$$LEC = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$
$$BC = [30]$$
$$LB = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$
$$UB = \begin{bmatrix} 15 \\ 20 \end{bmatrix}$$

Then, ‘Q’, ‘c’, ‘LEC’, ‘BC’,’LB’, and ‘UB’ is typed into the command window in MATLAB. By letting ‘A’=[] and ‘B’=[] as no inequality constraints, the ‘quadprog’ solver is called to solve the coding by inserting the function of [x, fval, exitflag, output]=quadprog (Q, c, A, B, LEC,BC, LB,UB).

```
Command Window
>> Q=2*[32 0; 0 8];
>> c=[-2;-10];
>> A=[];
>> B=[];
>> LEC=[1 1];
>> BC=[30];
>> LB=[0;0];
>> UB=[15;20];
>> [x,fval,exitflag,output]=quadprog(Q,c,A,B,LEC,BC,LB,UB)

Minimum found that satisfies the constraints.

Optimization completed because the objective function is non-decreasing in
feasible directions, to within the value of the optimality tolerance,
and constraints are satisfied to within the value of the constraint tolerance.

<stopping criteria details>

x =

   10.0000
   20.0000


fval =

   6.1800e+03
```

Figure 3.2.2: Result obtained from MATLAB by solving quadratic programming model using ‘quadprog’ solver

```
Command Window
x =

   10.0000
   20.0000


fval =

   6.1800e+03


exitflag =

     1


output =

  struct with fields:

            message: '↵Minimum found that satisfies the constraints.↵↵Optimization
          algorithm: 'interior-point-convex'
      firstorderopt: 4.3257e-08
    constrviolation: 0
         iterations: 7
       linearsolver: 'dense'
       cgiterations: []
```

Figure 3.2.3: Output Result Obtained

Based on figures 3.2.2 and 3.3.3 obtained from MATLAB, the exitflag=1 represents that the result is a local minimum. Since Q from the problem is a positively defined matrix, therefore this is a convex problem. Other than that, x1=10.0000, x2=20.0000, and fval=6180.00, hence it can be concluded that by treating the affected rivers with different types of wastewater treatments A and B, the cost required can be minimized into RM6180.00.

3.3 Optimization in the Production of Stainless Steel- *‘intlinprog’* *[Khadija Nadim]*

Background:

The objective of this is to make a program that enables a steel company to optimize the quantity of alloys and steel scraps used to minimize the cost in the production of stainless steel. Alloys are more expensive than steel scraps and it is important to sort the quantities of each alloy required, so as to lower the cost. In the following problem, the objective is to obtain 25 tons of steel with a composition of 5% carbon and 5% molybdenum by weight.

The available steel ingots available are given below. There are only one of each.

Ingot	Weight (tons)	% Carbon	% Molybdenum	Cost (per ton)
1	5	5	3	$350
2	3	4	3	$330
3	4	5	4	$310
4	6	3	4	$280

The available alloys available are given below along with their composition. Fractional amounts of alloys and scrap can also be purchased.

Alloy	% Carbon	% Molybdenum	Cost (per ton)
1	8	6	$500
2	7	7	$450
3	6	8	$400
Scrap	3	9	$100

Solution:
5% Carbon and 5% Molybdenum by weight
= 25 x 0.05 = 1.25 tons
= 1.25 tons of carbon and 1.25 tons of molybdenum.
Let x(1) through x(4) be the weights of the ingots purchased and similarly x(5) through x(7) be the respective quantity of alloys purchased in tons. Lastly, x(8) is the quantity of scrap steel purchased in tons.
Constraints:

x(1), x(2), x(3), x(4) = [0,1]	// Either an ingot is purchased or it is not, therefore it results in a binary value
x(5), x(6), x(7) $\geq$ 0	// Non-zero constraint
x(8) $\geq$ 0	// Non-zero constraint

Equality constraints:
Total weight = 25 tons
5*x(1) + 3*x(2) + 4*x(3) + 6*x(4) + x(5) + x(6) + x(7) + x(8) = 25

Weight of Carbon = 1.25 tons
5*0.05*x(1) + 3*0.04*x(2) + 4*0.05*x(3) + 6*0.03*x(4)+ 0.08*x(5) + 0.07*x(6) + 0.06*x(7) + 0.03*x(8) = 1.25
Weight of Molybdenum = 1.25 tons
5*0.03*x(1) + 3*0.03*x(2) + 4*0.04*x(3) + 6*0.04*x(4) + 0.06*x(5) + 0.07*x(6) + 0.08*x(7) + 0.09*x(8) = 1.25
This program does not have any non-linear constraints.
Matlab Solver:

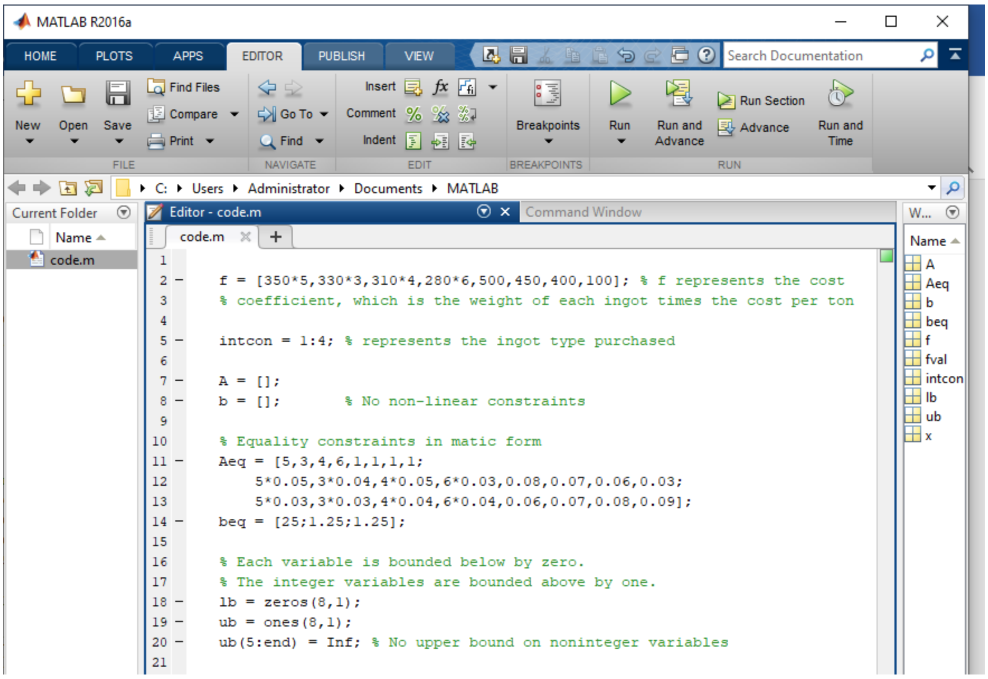


Figure 3.3.1: Code Written in MATLAB

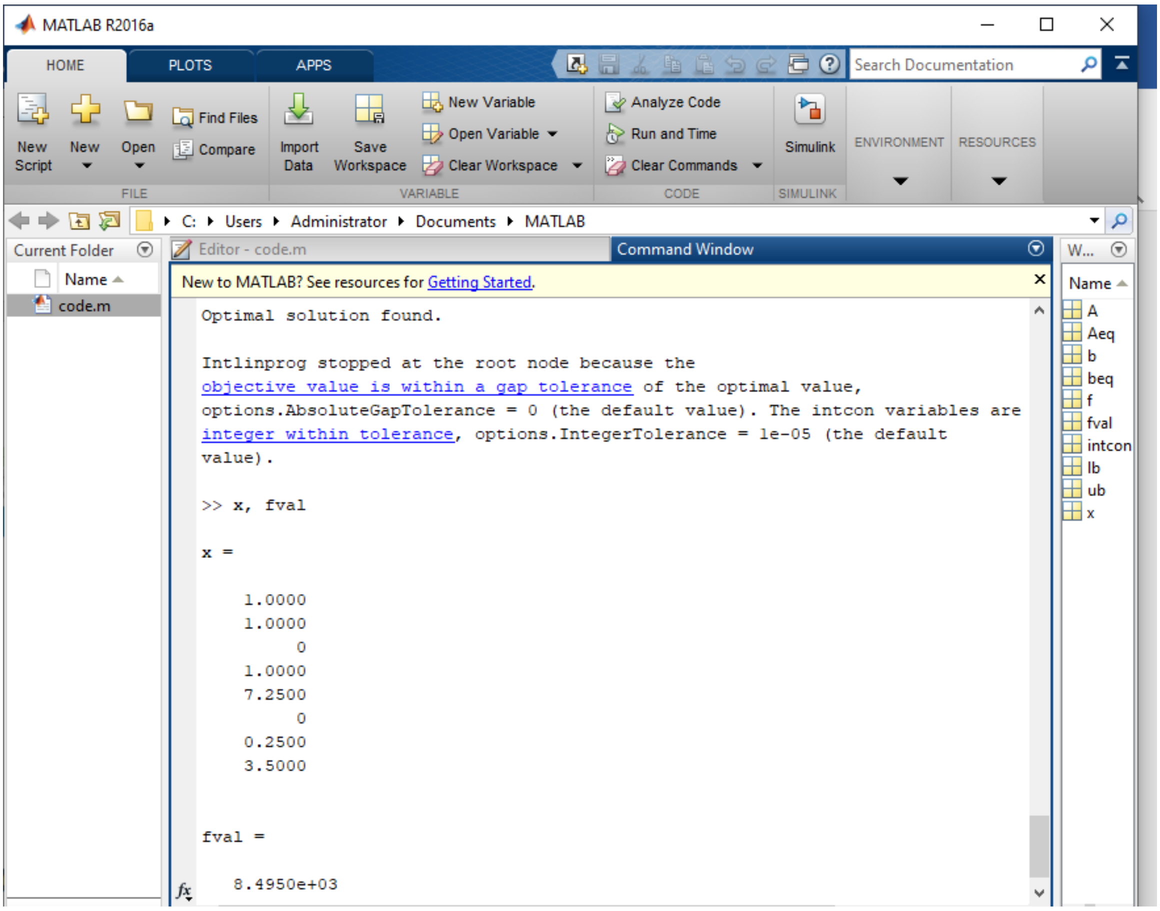


Figure 3.3.2: Optimal Solution by Typing ‘x’ and ‘fval’ in Command Window

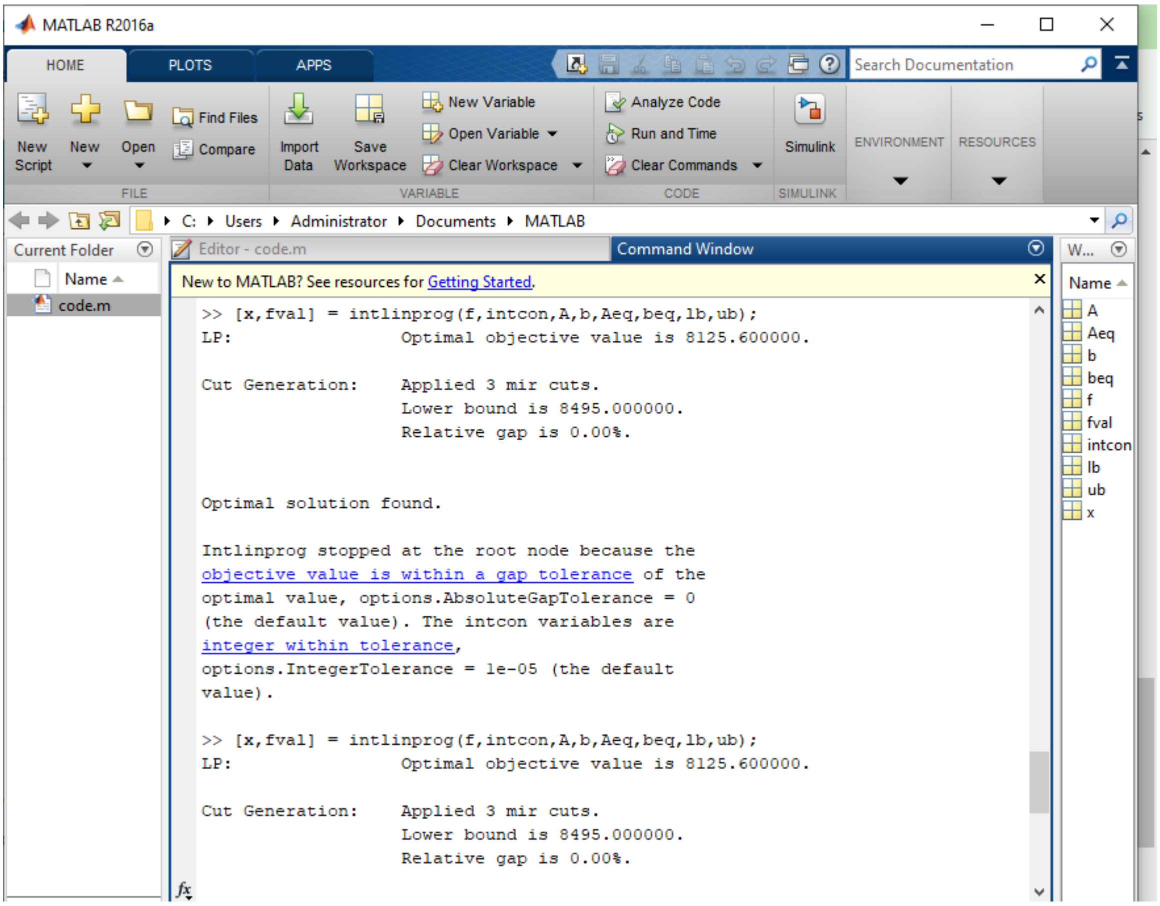


Figure 3.3.3: Optimal Solution

3.4 Production of Thermoplastic in a Chemical Reactor- 'fmincon' *[Tham Ting Woon]*

Background

A chemical plant produces chemical product which required two raw materials, A and B. Production of thermoplastic will be applied in chemical plant. The raw materials needed are Polyethylene (PE) pellets and Ethylene vinyl acetate (EVA) copolymer pellets (Onay, 2022). Weight- based batch process will be applied to produce thermoplastic polymer. In the production of thermoplastic, polymerization, molding, cooling and finishing will undergo.

Raw materials are fed into a chemical reactor and mix with catalyst. During polymerization, the raw materials will undergo chemical reaction of monomer to form the polymer chain (TWI,2023). High-pressure polymerization will be applied in this process. The condition of reactor such as temperature, pressure and residence time should be set and monitored to ensure the quality of thermoplastic resin produced.

After the polymerization process, the thermoplastic resin is purified to remove the impurities and unreacted monomer and catalyst. The thermoplastic resin can be purified through filtration. Filter used in this filtration process such as mesh screens and porous membrane (Mallikarjun, n.d). The different size of impurities will be removed through different filter to ensure the efficiency of filtration.

The purified thermoplastic resin will then undergo molding process. Different technique can be applied depending on the shape of size desired of the product such as injection molding, blow molding and extrusion (Ebnesajjad, 2015). To create a complex shape, injection molding will be suitable to applied. Besides, air is blow in a hot hollow tube to expand and shape it to different form. This is blow molding which usually to produce bottles and container. In extrusion, molten thermoplastic material will be force into a die to create pipes (Blackburn & Szymiczek, 2021).

The last step is cooling and finishing. The thermoplastic product is then cooled down to solidify it. Excess resources and waste will be recycled back to the chemical reactor since thermoplastic is recyclable.

The optimization function used in this scenario is 'fmincon'. 'Fmincon' is a MATLAB nonlinear programming tool that helps to find the minimum constrained nonlinear multivariable function. Two raw materials are mixed in a chemical reactor through batch process. The objective in above scenario is to minimize the production cost of chemical product, subject to constraints. According to Mills (2016), the total amount of raw materials A, polyethylene (PE) pellets in each batch cannot exceed 1000 kg while the total amount of raw materials B, Ethylene vinyl acetate (EVA) copolymer pellets used in batch cannot exceed 800 kg. The final product must contain at least 20 percent and at most 30 percent of raw material, A. For raw material B, the final product must contain at least 50 percent and at most 60 percent. The cost of raw material A is RM 1.50 per kg while the cost of raw material B is RM2.00 per kg (Alibaba,nd). The objective function is a quadratic function while the constraints consist of bound constraints and inequality constraints.

Objective function:

$$Minimize\ C = 1.5\ {x_1}^2 + 2{x_2}^2$$

Where x_1 = raw material A
x_2 = raw material B

Subject to:

$$x_1 \le 1000$$
$$x_2 \le 800$$
$$0.2(x_1 + x_2) \le x_1 \le 0.3(x_1 + x_2)$$
$$0.5(x_1 + x_2) \le x_2 \le 0.6(x_1 + x_2)$$
$$x_1, x_2 \ge 0$$

MATLAB software is used to solve this problem. The method use is OPTIMTOOL.

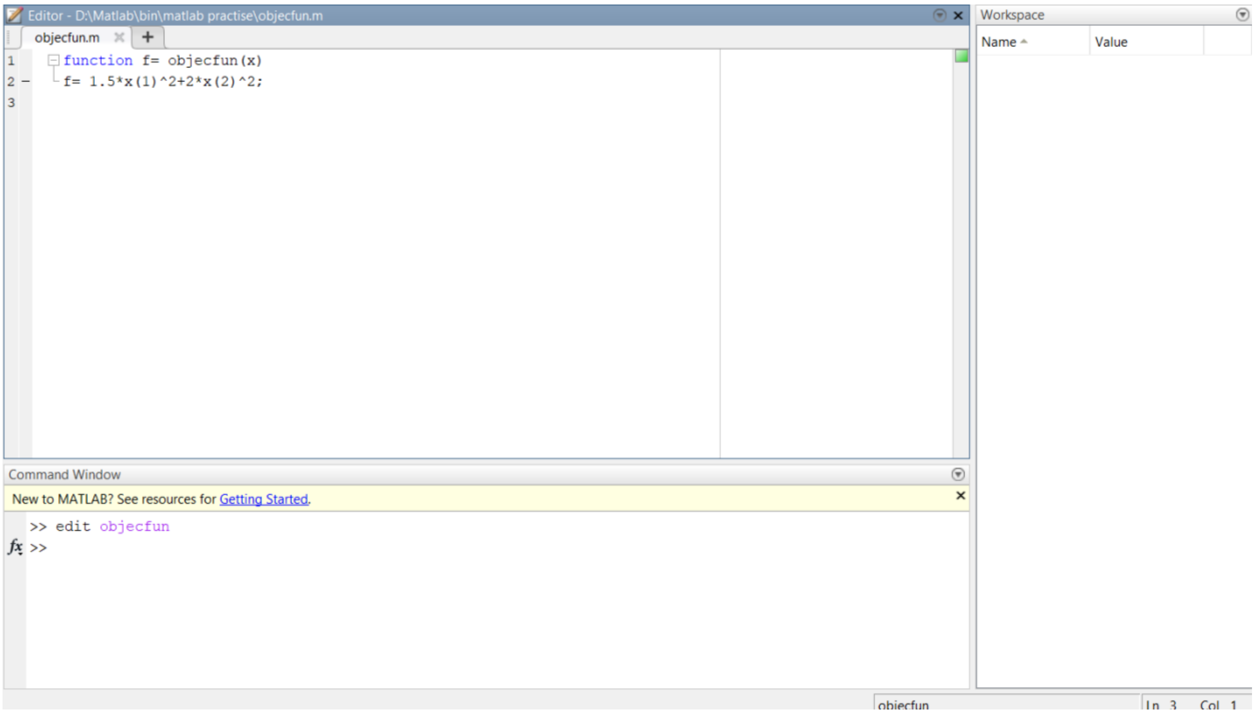


Figure 3.4.1: M-File of Objective Function

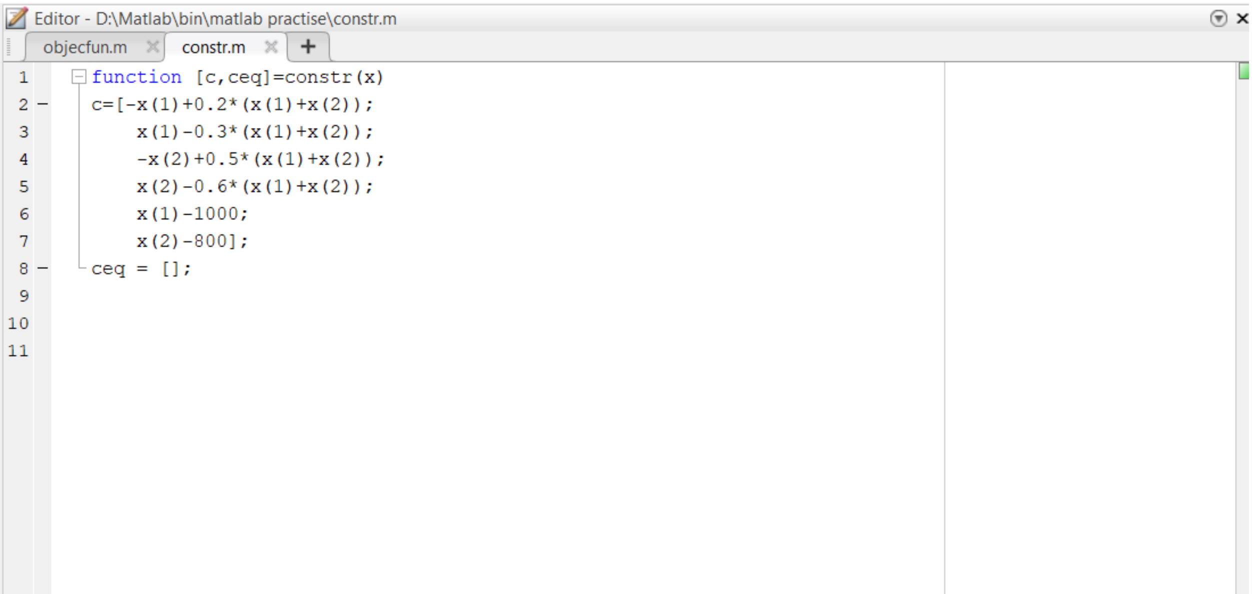


Figure 3.4.2: M- File for Constraints

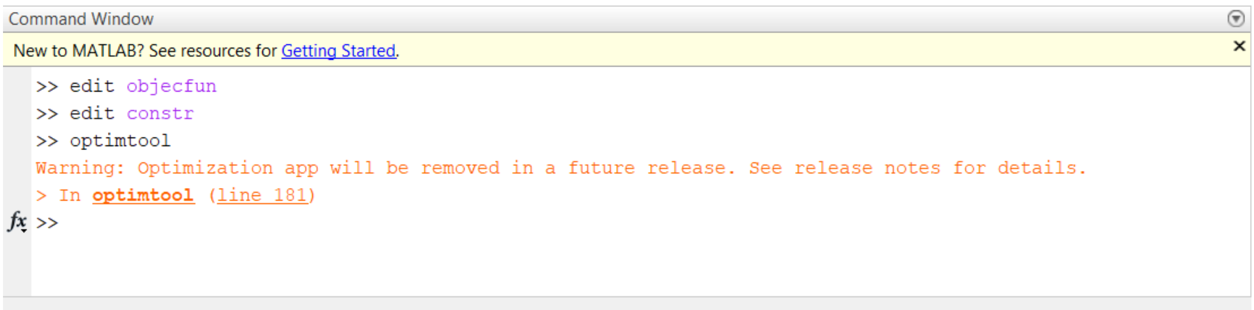


Figure 3.4.3: Command Window

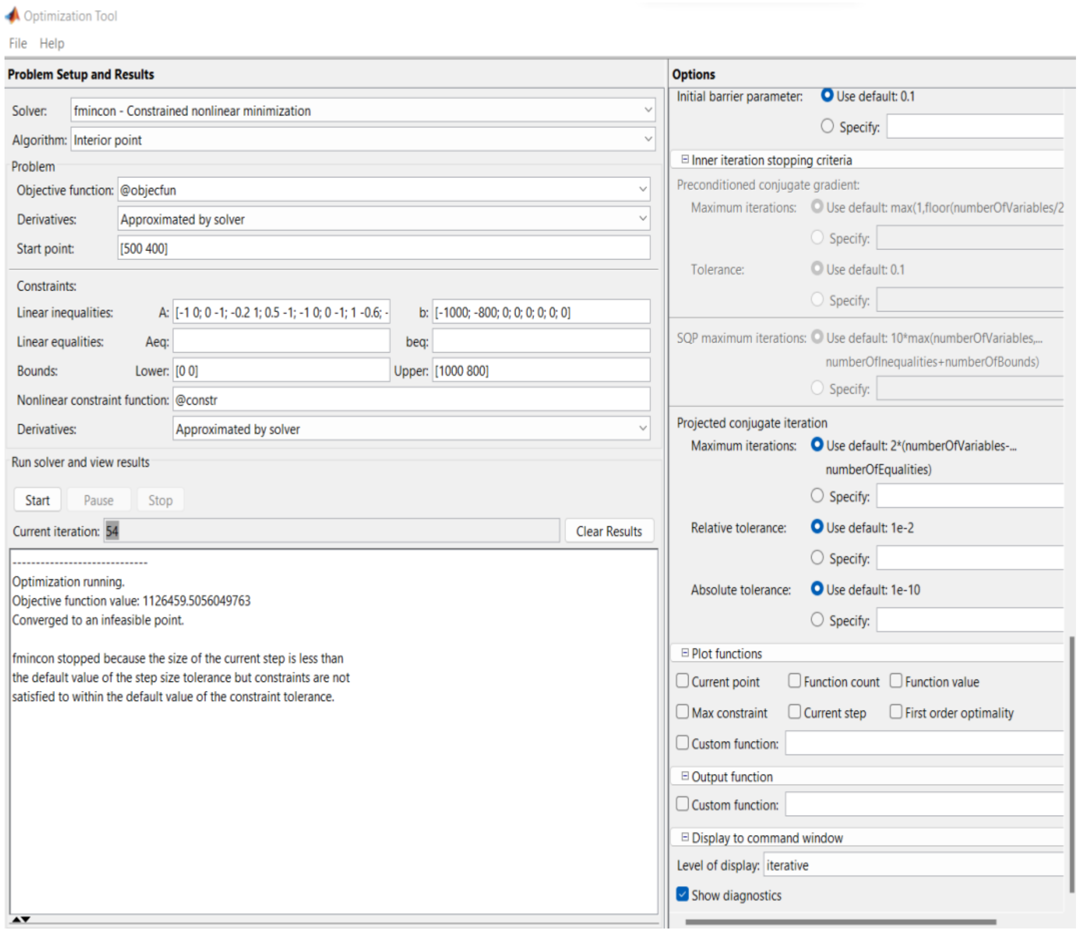


Figure 3.4.4: Details of Optimtool

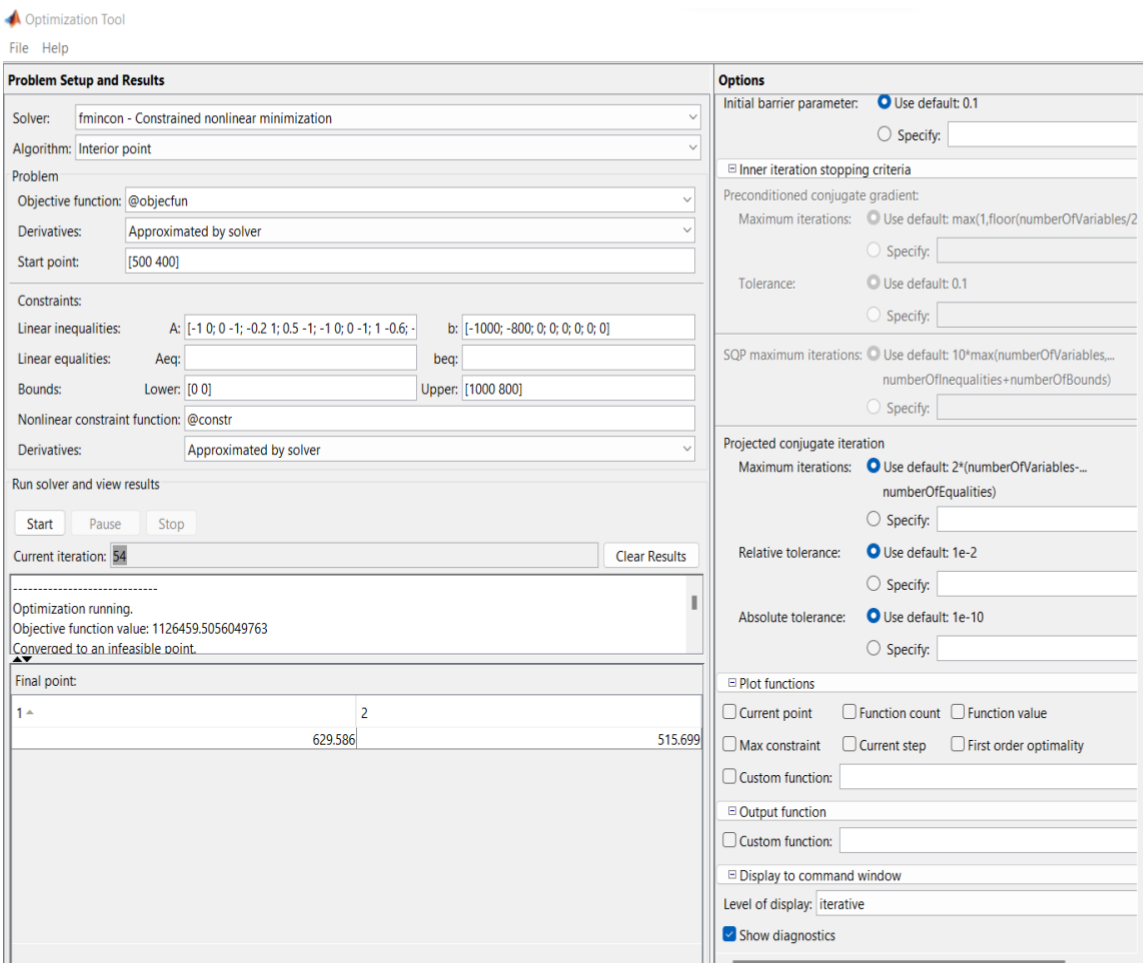


Figure 3.4.5: Details of Optimtool

This problem can be solved by using optimization tool in MATLAB. In Figure 1 and 2, M file are created for objective function and constraints. Edit (file name) are typed in the command window to generate the M-file. After typing in the objective function and constraints in the M-file, save the file were important. Next, optimization tool will be pop out after we typed optimtool in the command window as in Figure 3. This problem is solved by using 'fmincon' in the optimization tool. In algorithms section, interior point is selected to be used to solve this problem. Under the problem section of objective function, the M-file name of objective function is typed into it. The @character means that this is the function handle of the file. The start point is [500 400]. Under constraints parts, @constr is typed into the nonlinear constraint function as the function handle of constraint M-file. The linear inequality of A is set by [-1 0; 0 -1; -0.2 1; 0.5 -1; -1 0; 0 -1; 1 -0.6; -0.5 1] and b is [-1000; -800; 0; 0; 0; 0; 0; 0] according to the constraints in above scenario. The lower bound is set to be zero while the upper bound is set to be [1000 800]. In the option pane which located at the bottom in the center, iterative is selected in the display to command window menu. Last, click the start button to observe the objective function value and final point as in Figure 4 and 5. The objective function value in this scenario is 1126459.5056049763 while the final point for 1 is 629.586 and 2 is 515.699.

3.5 Optimization of Biological Wastewater Treatment Plant – '*fminunc*' (Dutta, 2016)

[Wong Pei Yu]

Background

Wastewater treatment is a crucial process that aims to purify influent wastewater for appropriate disposal or reuse, thereby safeguarding the environment by eliminating harmful substances to an acceptable level. Biological wastewater treatment is an effective technology that utilizes bacteria, microbes and protozoa to purify water. A study conducted by Dutta (2016) focuses on a rectangular tank has been made for biological treatment of wastewater. The process involved is a batch process, where treated water is discharged only after contaminants have been significantly reduced to safe levels. New batches of wastewater must wait until the previous batch has been completely processed and discharged. The sides and bottom of the tank cost, respectively, Rs. 1200/-, and Rs.2500/- per m² area. The operating cost for the tank is Rs.500/- for each batch of water treatment. A maintenance cost of Rs. 100/- for every 10 batch is required. Assuming that the tank will have no salvage value, the cost for treatment of 1000 m^3 of wastewater is needed to optimize. The salvage value of the tank is assumed to be zero after 1000 m^3 of wastewater treatment.

The optimization problem in this case aims to minimize the overall cost of water treatment, which includes the cost of constructing the tank, maintenance and operating the wastewater treatment process. The decision variables in this case include x_1, which refers to the length of the tank x_2, which refers to its width and x_3, which refers to the height of the tank. The objective function in this case is designed to minimize the total cost of the water treatment process.

Objective Function

The total cost of water treatment = cost of the tank operating cost of wastewater treatment + maintenance cost (cost of sides + cost of bottom)+ number of batch x (cost for each batch) + 100 x (number of barch)/10

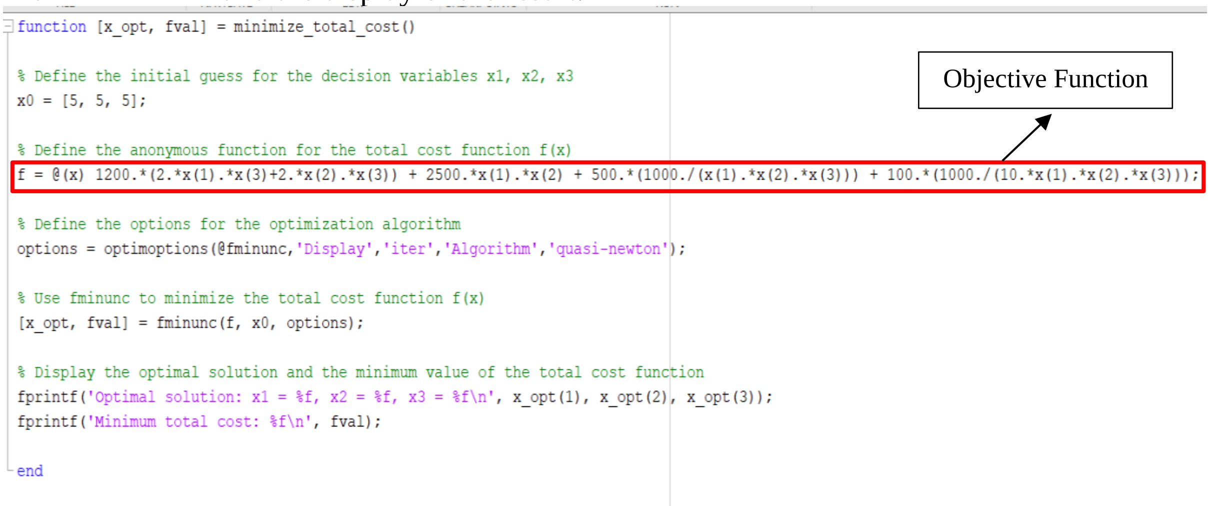
Minimize total cost:

$$f(x) = 1200(2x_1x_3 + 2x_2x_3) + 2500x_1x_2 + 500\left(\frac{1000}{x_1x_2x_3}\right) + 100\left(\frac{1000}{10x_1x_2x_3}\right)$$

Application of MATLAB solvers - fminunc

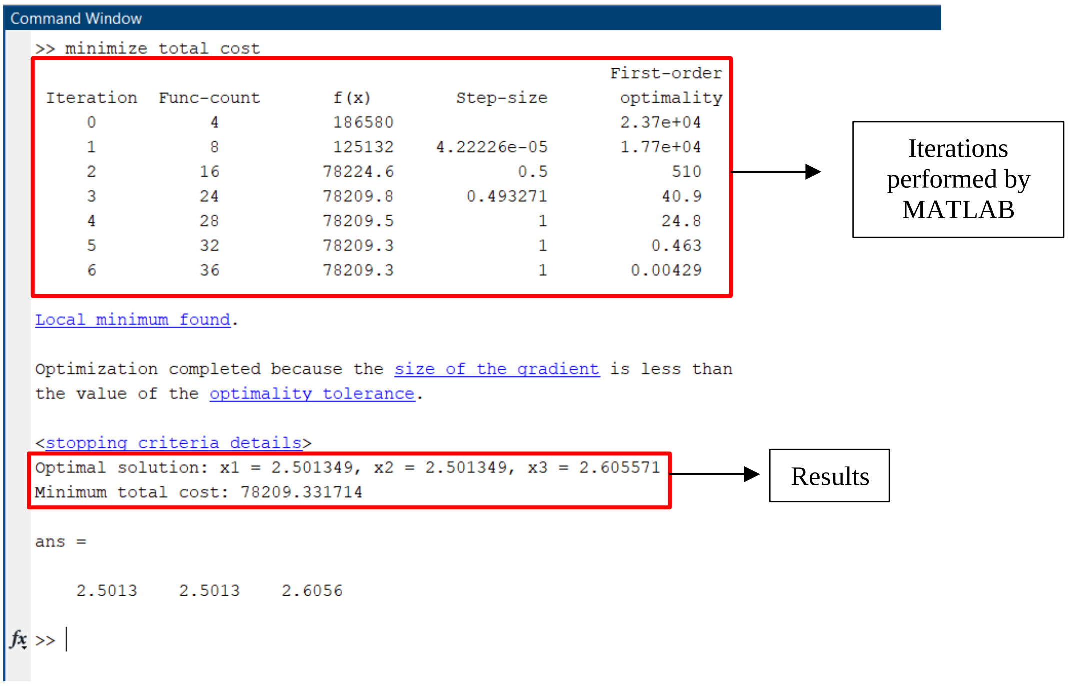
This optimization problem is a unconstrained multivariable problem that can be solved by utilizing the fminunc solver in MATLAB. It is a smooth problem that does not have any explicit constraints. Firstly, a script file is created in MATLAB to write down the code. The initial guess is crucial as it determines the number of iterations required to reach the optimal solution. If the initial guess is closer to the optimal point, fewer iterations will be needed. In this scenario, a starting point of 5 is assumed for each decision variable since the tank's dimensions cannot be less than zero. An anonymous function f(x) is defined for the total cost function, which takes a vector input x and returns the value of the total cost function at that input. Next, the code uses 'optimoptions' to set the display option to 'iter' so that the optimization process is printed to the console during runtime. 'iter' allows to see the detail and number of iterations done by MATLAB and the 'Algorithm' option is set to 'quasi-newton' to use the quasi-Newton algorithm for optimization. The 'fminunc' function is called to minimize the total cost function 'f' with the initial guess 'x0' and the options defined in step 4. The outputs of this function are the optimal values of the decision variables 'x_opt' and the minimum value of the total cost function 'fval'. The optimal values of the decision variables 'x_opt' and the minimum value of the total cost function 'fval' are displayed using the 'fprintf' function. Figure 3.5.1 shows the script for coding in MATLAB.

After calling the function "minimize_total_cost" in the command window, the optimal solution is found after 6 iterations as shown in figure 3.5.2. As a result, the optimal dimension of the biological wastewater treatment tank is the length of the reactor is 2.501 m, width is 2.501 m and height is 2.606 m. The minimum total cost of water treatment is Rs.78209/-. Figure 3.5.2 shows the iterations generated from MATLAB and the display of the result.



```
function [x_opt, fval] = minimize_total_cost()

% Define the initial guess for the decision variables x1, x2, x3
x0 = [5, 5, 5];

% Define the anonymous function for the total cost function f(x)
f = @(x) 1200.*(2.*x(1).*x(3)+2.*x(2).*x(3)) + 2500.*x(1).*x(2) + 500.*(1000./(x(1).*x(2).*x(3))) + 100.*(1000./(10.*x(1).*x(2).*x(3)));

% Define the options for the optimization algorithm
options = optimoptions(@fminunc,'Display','iter','Algorithm','quasi-newton');

% Use fminunc to minimize the total cost function f(x)
[x_opt, fval] = fminunc(f, x0, options);

% Display the optimal solution and the minimum value of the total cost function
fprintf('Optimal solution: x1 = %f, x2 = %f, x3 = %f\n', x_opt(1), x_opt(2), x_opt(3));
fprintf('Minimum total cost: %f\n', fval);

end
```


Figure 3.5.1: Script for Coding in MATLAB



```
Command Window
>> minimize_total_cost
                                                        First-order
 Iteration  Func-count       f(x)        Step-size       optimality
     0           4          186580                        2.37e+04
     1           8          125132      4.22226e-05       1.77e+04
     2          16         78224.6              0.5            510
     3          24         78209.8         0.493271           40.9
     4          28         78209.5                1           24.8
     5          32         78209.3                1          0.463
     6          36         78209.3                1        0.00429

Local minimum found.

Optimization completed because the size of the gradient is less than
the value of the optimality tolerance.

<stopping criteria details>
Optimal solution: x1 = 2.501349, x2 = 2.501349, x3 = 2.605571
Minimum total cost: 78209.331714

ans =

    2.5013    2.5013    2.6056

fx >>
```


Figure 3.5.2: Iterations Generated from MATLAB and The Display of Result

3.6 Script File and Short Video Summary

For script file, please refer to objfun.m, constr.m and script3.m and run them in order to obtain the result. In objfun.m, it variables such as cost_A and cost_B also allows for easy modification of the cost structure if necessary. In constr.m, it defines the constraints that must be satisfied to produce the chemical product, including upper and lower bounds on the values of x1 and x2. While for script3.m, it includes ‘fmincon’ which is an optimization solver that minimizes the objective function subject to the given constraints. Overall, the combination of a well-structured problem, the use of a powerful optimization solver, and clear organization and commenting make this code an excellent choice for solving problems.

Reference

Alibaba (n.d.) *Polypropylene pp granules injection grade recycle pellets, Polypropylene Pp Granules Injection Grade Recycle Pellets - Buy Polypropylene Homopolymer,Plastic Pellets,Pp Polypropylene Product on Alibaba.com*. Available at: https://www.alibaba.com/product-detail/Polypropylene-PP-Granules-Injection-Grade-Recycle_110858052.html?spm=a2700.pc_countrysearch.main07.56.1273322akhFPPm (Accessed: March 6, 2023).

Ashraf, A. and Chowdhury, N.B. (2018) *Esterification of ethanol and acetic acid in a batch reactor in presence of sul - the main purpose, Studocu*. Available at: https://www.studocu.com/row/document/auchi-polytechnic/chemical-engineering-analysis/esterification-of-ethanol-and-acetic-acid-in-a-batch-reactor-in-presence-of-sul/22455685 (Accessed: February 27, 2023).

Author links open overlay panel Tim Robinson *et al.* (2002) *Bioreactor design for protein enrichment of agricultural residues by solid-state fermentation, Chemical Engineering Journal*. Elsevier. Available at: https://reader.elsevier.com/reader/sd/pii/S1369703X02001328?token=FF3B3AC32D0E4381633B6548B4FF80B40EB261AF96F24EED17CB1EE54E2A8DC1ECE5ACCE096DCF68D076177A1F651145&originRegion=eu-west-1&originCreation=20230304051850 (Accessed: March 4, 2023).

Author links open overlay panelQianqian Zhang a b *et al.* (2019) *Development of an interval quadratic programming water quality management model and its solution algorithms, Journal of Cleaner Production*. Elsevier. Available at: https://www.sciencedirect.com/science/article/pii/S0959652619341897 (Accessed: March 2, 2023).

Baeyens, E., Herreros, A. and Perán, J.R. (2016) *A direct search algorithm for global optimization, MDPI*. Multidisciplinary Digital Publishing Institute. Available at: https://www.mdpi.com/1999-4893/9/2/40 (Accessed: February 26, 2023).

Blackburn, S. and Szymiczek, M. (2021) *Extrusion, Extrusion - an overview | ScienceDirect Topics*. Available at: https://www.sciencedirect.com/topics/materials-science/extrusion (Accessed: March 6, 2023).

Brownlee, J. (2021) *A gentle introduction to stochastic optimization algorithms*, *MachineLearningMastery.com*. Available at: https://machinelearningmastery.com/stochastic-optimization-for-machine-learning/ (Accessed: February 26, 2023).

Brownlee, J. (2021) *Iterated local search from scratch in Python*, *MachineLearningMastery.com*. Available at: https://machinelearningmastery.com/iterated-local-search-from-scratch-in-python/ (Accessed: February 26, 2023).

Calvar, N., González, B. and Dominguez, A. (2007) "Esterification of acetic acid with ethanol: Reaction kinetics and operation in a packed bed reactive distillation column," *Chemical Engineering and Processing: Process Intensification*, 46(12), pp. 1317–1323. Available at: https://doi.org/10.1016/j.cep.2006.10.007. (Accessed: February 27, 2023).

D. Poupon. (2023) *What is quadratic programming?, (with picture)*. Available at: https://www.wise-geek.com/what-is-quadratic-programming.htm (Accessed: February 27, 2023).

Dutta, S., 2016. Optimization in chemical engineering. Cambridge University Press. pp. 30,144.

Ebnesajjad, S. (2015) *Injection blow molding*, *Injection Blow Molding - an overview | ScienceDirect Topics*. Available at: https://www.sciencedirect.com/topics/engineering/injection-blow-molding (Accessed: March 6, 2023).

Key, J.A. (2014) *Factors that affect the rate of reactions*, *Introductory Chemistry 1st Canadian Edition*. BCcampus. Available at: https://opentextbc.ca/introductorychemistry/chapter/factors-that-affect-the-rate-of-reactions/ (Accessed: February 27, 2023).

Libretexts (2022) *13.4: Equilibrium Calculations*, *Chemistry LibreTexts*. Libretexts. Available at: https://chem.libretexts.org/Bookshelves/General_Chemistry/Chemistry_1e_(OpenSTAX)/13%3A_Fundamental_Equilibrium_Concepts/13.4%3A_Equilibrium_Calculations (Accessed: February 27, 2023).

Libretexts (2023) *2.5: Reaction rate*, *Chemistry LibreTexts*. Libretexts. Available at: https://chem.libretexts.org/Bookshelves/Physical_and_Theoretical_Chemistry_Textbook_Maps/Supplemental_Modules_(Physical_and_Theoretical_Chemistry)/Kinetics/02%3A_Reaction_Rates/2.05%3A_Reaction_Rate (Accessed: February 27, 2023).

Mallikarjun et al (no date) *US5442041A - removal of volatile substances from thermoplastic resins*, *Google Patents*. Google. Available at: https://patents.google.com/patent/US5442041A/en (Accessed: March 6, 2023).

Mathworks (2013) *What Is Direct Search?*, *Mathworks*. Available at: https://www.mathworks.com/help/gads/what-is-direct-search.html (Accessed: February 26, 2023).

Mills, D. (2016) *Polyethylene Pellet*, *Polyethylene Pellet - an overview | ScienceDirect Topics*. Available at: https://www.sciencedirect.com/topics/engineering/polyethylene-pellet (Accessed: March 6, 2023).

Mitchell, J., (2009). *Branch-and-cut example*. Available at https://homepages.rpi.edu/~mitchj/handouts/bc_eg/[Accessed 1 March 2022].

Mixed-integer linear programming basics: Solver-based. Mixed-Integer Linear Programming Basics: Solver-Based - MATLAB & Simulink. (n.d.). Retrieved March 7, 2023, from https://www.mathworks.com/help/optim/ug/mixed-integer-linear-programming-basics.html

Ogunleye, O.O. and Omodele Eletta (2012). Nonlinear Programming for Solvent Extraction of Jatropha Curcas Seed Oil for Biodesiel Production. [online] ResearchGate. Available at: https://www.researchgate.net/publication/314916319_Nonlinear_Programming_for_Solvent_Extraction_of_Jatropha_Curcas_Seed_Oil_for_Biodesiel_Production [Accessed 25 Feb 2023].

Oladejo, M. O. and Agashua, N. U., 2015. OPTIMIZING THE PRODUCT MIXINGS AND MACHINE PROCESSING TIME OF A FAST FOOD RESTAURANT USING A LINEAR PROGRAMMING MODEL: CASE STUDY OF OSTRICH BAKERY. Asian Journal of Science and Technology, 06(12), pp. 2043-2047

Onay, K. (2022) *What is plastic? what are plastic raw materials?*, *Divan Packaging*. Available at: https://divanpackaging.com/2022/05/07/what-is-plastic-what-are-plastic-raw-materials/ (Accessed: March 6, 2023).

R;, Y.S.S.T.L.J.N.P. (no date) *The removal of phosphorus during wastewater treatment: A Review*, *Environmental pollution (Barking, Essex : 1987)*. U.S. National Library of Medicine. Available at: https://pubmed.ncbi.nlm.nih.gov/15092663/#:~:text=Phosphorus%20removal%20from%20wastewater%20can,various%20mechanisms%20which%20are%20discussed. (Accessed: March 2, 2023).

Skrifvars, H. *et al.* (2003) *Comparison of different MINLP methods applied on certain chemical engineering problems*, *Computers & Chemical Engineering*. Pergamon. Available at: https://www.sciencedirect.com/science/article/abs/pii/009813549600066X (Accessed: March 4, 2023).

TWI.(2023) *What is a thermoplastic? (definition and examples)* (no date) *TWI*. Available at: https://www.twi-global.com/technical-knowledge/faqs/what-is-a-thermoplastic#:~:text=Thermoplastics%20are%20made%20by%20joining,forces%20between%20chains%20are%20weak. (Accessed: March 6, 2023).

Ugwuanyi, J.O., McNeil, B. and Harvey, L.M. (1970) *Production of protein-enriched feed using agro-industrial residues as substrates*, *SpringerLink*. Springer Netherlands. Available at: https://link.springer.com/chapter/10.1007/978-1-4020-9942-7_5 (Accessed: March 4, 2023).

Vedantu. (n.d). Graphical Method Linear Programming. [online] Available: <https://www.vedantu.com/maths/graphical-method-linear-programming> [Accessed 5 March 2022].

Westerberg, C.-H., Bjorklund, B., & Hultman, E. (1977, February). *An application of mixed integer programming in a Swedish steel mill - JSTOR*. JSTOR. Retrieved March 7, 2023, from https://www.jstor.org/stable/25059437