Chapter 1

Introduction

In introducing this study, we shall proceed under the following sub-headings:

- Preamble
- Motivation
- A Sample Problem
- Objective of the study

1.1 Preamble

In everyday human, corporate and industrial activities, we are faced with the challenges of optimal resource assignment. In the face of limited resources, low tolerance for scrap and rework, need for greater degree of accuracy and scalability, optimization of resources is paramount. Over the years, several Mathematical Optimization techniques in Operations Research (OR) specifically, have been developed to help with these class of problems witha good amount of success recorded. Nonetheless, there still remains a great room for improvement on these. Classical approaches to solving assignment problems, among others include the well revered Linear Programming techniques.

1.2 Motivation

The University Timetable Problem: This study, though it doesn't directly proffer a solution to this problem, it attempts to solve a simplified variant of it.

In Universities, there are usually a good number of students taking courses from a large pool, and in varying mixes, based on Faculty and Institutional requirements. Typically, there are limited facilities such as lecture halls to assign to each course. The problem therefore arises in creating an examination timetable that caters to all the courses, eliminating, or at least minimizing stress and ensuring usability of the timetable.
Studies show that, in order to prepare a well usable timetabling system, provision has to be made for such considerations as the following parameters:

1. Time Clash: A student group cannot have more than one exam at one time

2. Semester Clash: Student groups from the same major but in different semesters cannot have exams at the same time

3. Core Exams: A student group cannot take more than one core exam a day

4. Maximum Exams: A student group cannot have more than one exam per day

5. Difficulty Level: A student group cannot have two difficult exams in two consecutive days

6. Capacity: The total number of students seating for exams in a particular time cannot exceed predefined limits

7. One-period Execution: Each course must occur at the same time for all student groups

8. Periodic Unavailability: Some exams may not be scheduled at particular time slots

9. Comprehensiveness of Coverage: All exams on the timetable have to be assigned

10. Pre-assignment Capabilities: A course may be assigned to a particular target time slot

11. Controllable Exams Conclusion: Student in all student groups should conclude their exams at approximately the same time

The foregoing attributes have been cited as some of the desirable properties of a credible timetabling system. Proffering credible solutions to the timetabling problem has been a major challenge in the wide, that annually, the International Timetabling Competition 2019 (www.itc2019.org) holds to select a number of outstanding solutions to the timetabling problem.
Although, this problem has been identified as a NP-Hard class of problems, it is still a long way from being efficiently solved to tackle the highlighted problems

This study takes a whole new look at Mathematical Optimization: what if our optimization goal doesn't involve minimizing (or maximizing) any variable or group of variables such as cost, profit, or even price? What if we're on the otherhand interested in

Allocation of objects to non-intersecting groups, based on

attributes possessed by the objects, and one or more

constraining conditions

Mathematical Optimization techniques, have not been formulated to solve problems as this, and thus the motivaiton for this study.

1.3 A Reduction Problem

Consider the following problem statement: In deploying intending NYSC members to a serving state, the following constraints need be satisfied:

1. State of Origin: No student should be deployed to his/her state of origin
2. State of Study: No student should be deployed to his state of study
3. Comprehensiveness of Coverage: All enlisted students must be deployed
4. Capacity: No state should be allocated more students that it has resources to accommodate

The foregoing problem statement is an attempt at reducing the more complex University Timetabling problem to a simpler form where the parameters involved are better appreciated. This study focuses on resolving the simpler NYSC batching problem, and makes recommendations on applying the concepts herein in tackling the more complex University Timetabling Problem.

1.4 Objective

T he aim of this study is to:

- Develop a mathematical algorithm to solve assignment problems involving the distribution of candidate objects into non-overlapping groups, putting into considerations several constraining conditions
- Create a simulation, exercising this model on a sample problem

- Sketch a path to evolving this class of solution to more complex University Timetabling problem, owing to its more complex constraints formulation.

Throughout this text, all mathematical notations retain their usual meanings, unless otherwise stated.

1.5 Definition of Terms

1.5.1 Membership Probability or Degree

Each assignable object in all cases studied in this text, belong to each assignable slot with varying degrees of membership, usually a value in the interval [0,1]. An object with a membership degree of 0 is referred to as unfit for the class, while an object have a higher value has a better fit, with the value of 1 being a perfect fit into the class in question.

1.5.2 Partition

Let X be a set, and $A_1, A_2, ..., A_n$ be subsets of sets such that $A_1 \cap A_2 \cap \cdots \cap A_n = \emptyset$ and $A_1 \cup A_2 \cup \cdots \cup A_n = X$. The disjoint sets $A_1, A_2, \ldots, A_n$ are referred to as a *partition* of the set X

Throughout this text, all mathematical notations retain their usual meanings, unless otherwise stated.

Chapter 2

Literature Review

In this section we shall proceed to show some theoretical framework for this work.

2.1 Linear Programming (LP)

Although not used in this study, it bears mentioningAs mentioned above, Linear Programming techniques often lend themselves to making optimum resource allocation in many cases. The following definition accurately captures these techniques:

Linear Programming is a mathematical technique for

determining the optimal allocation of resources and obtaining a

particular objective (i.e, cost minimization or inversely profit

maximization) when there are alternative uses of resources:

Land, Capital, Materials, Machines, etc

Solving an assignment problem with Linear Programming techniques, often requires a clear statement of an Objective function, resources constraints, usually expressed as a set of multivariate linear equations.

The general formulation of a Linear Programming problem is given below:

Let Z be a linear function defined by

$$Z = c_1x_1 + c_2x_2 + ... + c_nx_n$$

where c_j's are constants

Let (a_{ij}) be mn constants and let (b_i) be a set of m constants

such that

$$a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n(\leq, =, \geq)b_1$$

$$a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n(\leq, =, \geq)b_2$$

.

.

.

$$a_{m1}x_1 + a_{m2}x_2 + ... + a_{mn}x_n(\leq, =, \geq)b_m$$

$$x_1 \geq 0, x_2 \geq 0, ..., x_n \geq 0$$

The problem of determining the values of $x_1, x_2, ..., x_n$, which makes Z a minimum (or maximum) and which satisfies (ii) and (iii) is called the General Linear Programming Problem

Linear Programming techniques, as well as other Mathematical Optimization techniques have been reputed a great success, nonetheless, there are several yet unaddressed perspectives of optimum assignment problems

2.2 Fuzzy Set

Since its inception in 1965, the theory of fuzzy sets has advanced in a variety of ways and in many disciplines. Applications of this theory can be found, for example, in artificial intelligence, computer science, medicine, control engineering, decision theory, expert systems, logic, management science, operations research, pattern recognition, and robotics. Mathematical developments have advanced to a very high standard and are still forthcoming to day. In this review, the basic mathematical framework of fuzzy set theory will be described.
We begin with a definition of the Fuzzy Set Theory:

If X is a nonempty collection of objects denoted generically by

$x_1, x_2, ..., x_n$ *for* $n = 1,\ 2,\ \ldots,$ *then a fuzzy set A in X is a pair*

$$(A, \mu_A)$$

where $A \subset X$ *and* $\mu_A \colon \boldsymbol{R} \to [0,1]$, *called the* membership

function of A

Examples of Fuzzy Sets

To better understand the concept of Fuzzy Sets, let us consider the following Examples

Example 1 Let us define a fuzzy set

$$A = \{\, x \mid x \in R,\ \text{real numbers near } 0 \ \}$$

The boundary for set real number near 0 is pretty ambiguous. The possibility of real number x to be a member of prescribed set can be defined by the following membership function.
We define the membership function

$$\mu_A \colon \mathbf{R} \to [0,1]$$

Defining the membership of elements of $\mathbb{R}$ in **A**
Defined as:

$$\mu_A(x) = \frac{1}{1+x^2} \tag{2.1}$$

Example 2 Let us define a similar fuzzy set

$$B = \{\, x \mid x \in R,\ \text{real numbers very near } 0 \ \}$$

We define the membership function

$$\mu_B \colon \mathbf{R} \to [0,1]$$

Defining the membership of elements of $\mathbb{R}$ in **B** such that:

$$\mu_B(x) = \left(\frac{1}{1+x^2}\right)^2 \tag{2.2}$$

The Theory of Fuzzy Sets forms the basic framework upon which the solution proposed by this work is formulated

Chapter 3

The Algorithm

In this section we shall proceed to develop the algrithm needed to solve the reduction problem proposed in the previous section.

Let the set (U) be a collection of objects to be assigned to partitioning sets

$$\mathbb{U} = \{u_1, u_2, \ldots, u_n\} \tag{3.1}$$

where each $u_i i = 1, 2, \ldots$,n are the objects possessing characteristics to be assigned to various partition classes, e.g say, the group of graduates to be posted to NYSC serving states

And let

$$p_1, p_2, \ldots, p_k \tag{3.2}$$

be the partitions to which objects u_i's are to be assigned. We observe that there are k such classes, eg, each p_i would be a state where graduates can be posted to for their Service Year

We now let

$$J(p_i) = j_i \tag{3.3}$$

be the maxiumum capacity the partition p_i can contain. That is, the maximum number of u_i's each partition p_i can contain

Let V,

$$V = \sum_{i=1}^{k} J(p_i) = \sum_{i=1}^{k} j_i \tag{3.4}$$

be the total capacity of the system, that is, the total number of u_i's all the partitions can contain combined.

NB: It should be noted that for an optimum solution to be feasible, $\mid U \mid \leq V$ needs be satisfied

We now define the weight, w_i of each partition p_i

$$w_i = \frac{j_i}{V} = \frac{J(p_i)}{\sum_{i=1}^{k} J(p_i)} \textit{ for i=1,2,3,\ldots,k} \tag{3.5}$$

The weight, w_i represents the proportion of each p_i's capacity in the whole

We now define the expected cardinality, $\mid p_i \mid$ of each partition p_i

$$\mid p_i \mid = w_i \cdot \mid U \mid = w_i \cdot n, \textit{for i=1,2,3,\ldots,k} \tag{3.6}$$

The cardinality, $\mid p_i \mid$ represents the number of objects proportionately assignable to each partition p_i

We now proceed to define constraints functions, akin to membership function of a Fuzzy Set. But in this case, they membership functions are of two types: inclusion and exclusion types.

Inclusion Constraints require that the value of a characteristic measured on candidate objects, u_i match that prescribed by the constraints, while on the otherhand, Exclusion constraints define characteristics that once a candidate object u_i posseses, it is disqualified (by assigning a value of 0) from being a probable member of the partition in question

Constraints, like membership functions, are generally of the form

$$\mu \colon U \mapsto [0,1]$$

Inclusion Constraints are defined thus:

$$\mu_j(u_i) = \frac{1}{1 + (j - s(u_i))^2} \tag{3.7}$$

And, Exclusion Constraints are defined thus:

$$\mu_j(u_i) = \frac{(j - s(u_i))^2}{1 + (j - s(u_i))^2} \tag{3.8}$$

for i=1,2,...,n and *j=1,2,...,k*
and

$$s \colon U \mapsto \{1, 2, \ldots, k\} \tag{3.9}$$

is a helper function defined by the individual Constraint function, μ_j to get the index of the partition class p_j with which each object u_i shares the characteristic being measured

For instance, if s is the state of origin helper function, it returns the index of the partition class p_i to which the object u_i maps as a state of origin

NB: $p_1, p_2, \ldots, k$ are indexed based on proximity or similarity. That is, the difference between the indices of any two partitions is directly proportional to the distance between them.

Now, in situations where there are more than one Constraint functions defined, we obtain a single Constraint function by multiplying through viz:

$$\mu_i(u_j) = \prod_{t=1}^{s} \mu_{i_t}(u_j) \tag{3.10}$$

for i=1,2,...,k; j=1,2,...,n and s is the total number of Constraint functions defined

We now define the membership set for each partition. These sets list the probability of each object u_i being a member of the partition p_j

$$\mu_i = \{\mu_i(u_1), \mu_i(u_2), \ldots, \mu_i(u_n)\} \tag{3.11}$$

So, for example

$$\mu_1 = \{\mu_1(u_1), \mu_1(u_2), \ldots, \mu_1(u_n)\} \tag{3.12}$$

and,

$$\mu_2 = \{\mu_2(u_1), \mu_2(u_2), \ldots, \mu_2(u_n)\} \tag{3.13}$$

and,

$$\mu_3 = \{\mu_3(u_1), \mu_3(u_2), \ldots, \mu_3(u_n)\} \tag{3.14}$$

$$\vdots$$

$$\mu_k = \{\mu_k(u_1), \mu_k(u_2), \ldots, \mu_k(u_n)\} \tag{3.15}$$

We now proceed to populate each partition with the objects with the highest membership probability, in turn

Let

$$\mu_{p_i} = \mu_i \backslash \bigcup_{j=1}^{u-1} \{\mu_j(u_x) \forall u_x \in p_j\} \tag{3.16}$$

That is, isolating the already assigned objects u_x from the membership set to avoid multiple membership

Now, sorting out the first $\mid p_k \mid$ objects with the highest membership points:
Let

$$Q_1 = u_x \mid \mu_i(u_x) = \max\{\mu_{p_i}\} \tag{3.17}$$

And

$$Q_2 = u_x \mid \mu_i(u_x) = \max\{\mu_{p_i} \backslash \mu_i(u_y) \mid u_y \in Q_1\} \tag{3.18}$$

And

$$Q_3 = u_x \mid \mu_i(u_x) = \max\left\{\mu_{p_i} \backslash \mu_i(u_y) \mid u_y \in Q_1 \bigcup Q_2\right\} \tag{3.19}$$

And

$$Q_4 = u_x \mid \mu_i(u_x) = \max\left\{\mu_{p_i} \backslash \mu_i(u_y) \mid u_y \in Q_1 \bigcup Q_2 \bigcup Q_3\right\} \tag{3.20}$$

$$\vdots$$

$$Q_{|p_i|} = u_x \mid \mu_i(u_x) = \max\left\{\mu_{p_i} \setminus \bigcup_{j=1}^{|p_i|-1} \{\mu_i(u_y) \mid u_y \in Q_j\}\right\} \tag{3.21}$$

Whence we finally have that

$$p_i = \bigcup_{j=1}^{|p_i|} \{Q_j\} \qquad \textit{for } i\textit{=1,2,}\ldots\textit{,}k \tag{3.22}$$

Chapter 4

Solving A Sample Problem

In this section, we attempt to apply the algorithm developed in the previous chapter to solve a sample problem. The problem is an attempt to assign a group of 20 University graduates to one of 5 states for service.

4.1 Statement of Constraints

1. No students should be posted to their State of Origin
2. No students should be posted to their States of study
3. The maximum capacity of each state must not be exceeded
4. All students should be assigned

The 3rd and 4th constraints are inately adhered to by the algorithm, so we'll focus only on constructing Constraint Functions for the first two constraints.

4.2 The States

The States in this problem stand for the partitions to be filled. In order to simplify the problem, we focus only on 5 states and assume each of the 20 graduates schooled in one of these and originated from one of these states. The data represented here is in the form {State,Capacity}

1. p_1: {Osun State, 3}
2. p_2: {Kwara State, 2}
3. p_3: {Enugu State, 1}

4. p_4: {Akwa-Ibom State, 5}
5. p_5: {Kano State, 10}

4.3 The Graduates

Each graduate has a number of characteristics which include names, State of origin, and Alma Mater. Here is our list of 20 graduates which we shall consider in this sample problem:

1. u_1: {GR01, Kwara(2), OAU(1)}
2. u_2: {GR02, Kwara(2), BUK(5)}
3. u_3: {GR03, Kwara(2), NSUKKA(3)}
4. u_4: {GR04, Kwara(2), UNILORIN(2)}
5. u_5: {GR05, Akwa-Ibom(4), OAU(1)}
6. u_6: {GR06, Akwa-Ibom(4), BUK(5)}
7. u_7: {GR07, Akwa-Ibom(4), NSUKKA(3)}
8. u_8: {GR08, Akwa-Ibom(4), UNILORIN(2)}
9. u_9: {GR09, Osun(1), OAU(1)}
10. u_{10}: {GR10, Osun(1), BUK(5)}
11. u_{11}: {GR11, Osun(1), NSUKKA(3)}
12. u_{12}: {GR12, Osun(1), UNILORIN(2)}
13. u_{13}: {GR13, Kano(5), OUI(1)}
14. u_{14}: {GR14, Kano(5), NSUKKA(3)}
15. u_{15}: {GR15, Kano(5), UNILORIN(2)}
16. u_{16}: {GR16, Kano(5), AKSU(4)}
17. u_{17}: {GR17, Enugu(3), UNIOSUN(1)}
18. u_{18}: {GR18, Enugu(3), BUK(5)}
19. u_{19}: {GR19, Enugu(3), UNILORIN(2)}
20. u_{20}: {GR20, Enugu(3), AKSU(4)}

4.4 States - Capacity, Weights and Cardinality

As indicated in the States data, each state has a capacity, j_i. So in this case,

$$J(p_1) = j_1 = 3$$

$$J(p_2) = j_2 = 2$$

$$J(p_3) = j_3 = 1$$

$$J(p_4) = j_4 = 5$$

$$J(p_5) = j_5 = 10$$

Thus, the maximum capacity of the system is given as:

$$V = \sum_{i=1}^{5} J(p_i) = 3 + 2 + 1 + 5 + 10 = 21 \tag{4.1}$$

Since V is less than the total number of graduates awaiting deployment (20), a feasible assignment is possible.

The weight, w_i of each state is defined thus:

$$w_i = \frac{j_i}{V}, \qquad \textit{for i=1,2,...,5} \tag{4.2}$$

$$w_1 = \frac{j_1}{V} = \frac{3}{21} = 0.143 \tag{4.3}$$

$$w_2 = \frac{j_2}{V} = \frac{2}{21} = 0.095 \tag{4.4}$$

$$w_3 = \frac{j_3}{V} = \frac{1}{21} = 0.048 \tag{4.5}$$

$$w_4 = \frac{j_4}{V} = \frac{5}{21} = 0.238 \tag{4.6}$$

$$w_5 = \frac{j_5}{V} = \frac{10}{21} = 0.476 \tag{4.7}$$

We now define the expected cardinality, $\mid p_i \mid$ of each state. That is, the expected number of Corp members posted to each state

$$\mid p_i \mid = w_i \cdot \text{no. of graduates} = w_i \cdot n = w_i \cdot 20, \qquad \textit{for i=1,2,...,5} \tag{4.8}$$

$$\mid p_1 \mid = w_1 \times 20 = 0.143 \times 20 = 2.86 \approx 3 \tag{4.9}$$

$$\mid p_2 \mid = w_2 \times 20 = 0.095 \times 20 = 1.9 \approx 2 \tag{4.10}$$

$$\mid p_3 \mid = w_3 \times 20 = 0.048 \times 20 = 0.96 \approx 1 \tag{4.11}$$

$$\mid p_4 \mid = w_4 \times 20 = 0.238 \times 20 = 4.76 \approx 5 \tag{4.12}$$

$$\mid p_5 \mid = w_5 \times 20 = 0.476 \times 20 = 9.52 \approx 9 \tag{4.13}$$

4.5 Constraints

The sample problem has two statements of exclusion constraints. We will establish the Constraint functions defining the membership probability of each graduate into each state thus:

Constraint A, μ_A: No student should be posted to their State of Origin

$$\mu_{A_j}(u_i) = \frac{(j - s_A(u_i))^2}{1 + (j - s_A(u_i))^2} \tag{4.14}$$

where $\mu_{A_j}(u_i)$ measures the degree of object u_i's membership in State j under the consideration of the *state of origin* constraint and, $s_A(u_i)$ represents the index of graduate u_i's State of Origin

$$\begin{aligned} \mu_{A_1}(u_1) &= \frac{(1 - s_A(u_1))^2}{1 + (1 - s_A(u_1))^2} \\ &= \frac{(1-2)^2}{1+(1-2)^2} \\ &= 0.5 \end{aligned} \tag{4.15}$$

$$\begin{aligned} \mu_{A_1}(u_2) &= \frac{(1 - s_A(u_2))^2}{1 + (1 - s_A(u_2))^2} \\ &= \frac{(1-2)^2}{1+(1-2)^2} \\ &= 0.5 \end{aligned} \tag{4.16}$$

$$\begin{aligned} \mu_{A_1}(u_3) &= \frac{(1 - s_A(u_3))^2}{1 + (1 - s_A(u_3))^2} \\ &= \frac{(1-2)^2}{1+(1-2)^2} \\ &= 0.5 \end{aligned} \tag{4.17}$$

$$\vdots$$

$$\begin{aligned} \mu_{A_1}(u_{20}) &= \frac{(1 - s_A(u_{20}))^2}{1 + (1 - s_A(u_{20}))^2} \\ &= \frac{(1-3)^2}{1+(1-3)^2} \\ &= 0.8 \end{aligned} \tag{4.18}$$

$$\vdots$$

$$\begin{aligned}\mu_{A_2}(u_1) &= \frac{(2 - s_A(u_1))^2}{1 + (2 - s_A(u_1))^2} \\ &= \frac{(2-2)^2}{1+(2-2)^2} \\ &= 0\end{aligned} \tag{4.19}$$

$$\begin{aligned}\mu_{A_2}(u_2) &= \frac{(2 - s_A(u_2))^2}{1 + (2 - s_A(u_2))^2} \\ &= \frac{(2-2)^2}{1+(2-2)^2} \\ &= 0\end{aligned} \tag{4.20}$$

$$\begin{aligned}\mu_{A_2}(u_3) &= \frac{(2 - s_A(u_3))^2}{1 + (2 - s_A(u_3))^2} \\ &= \frac{(2-2)^2}{1+(2-2)^2} \\ &= 0\end{aligned} \tag{4.21}$$

$$\vdots$$

$$\begin{aligned}\mu_{A_2}(u_{20}) &= \frac{(2 - s_A(u_{20}))^2}{1 + (2 - s_A(u_{20}))^2} \\ &= \frac{(2-3)^2}{1+(2-3)^2} \\ &= 0.50\end{aligned} \tag{4.22}$$

$$\vdots$$

$$\begin{aligned}\mu_{A_3}(u_1) &= \frac{(3 - s_A(u_1))^2}{1 + (3 - s_A(u_1))^2} \\ &= \frac{(3-2)^2}{1+(3-2)^2} \\ &= 0.5\end{aligned} \tag{4.23}$$

$$\begin{aligned}\mu_{A_3}(u_2) &= \frac{(3 - s_A(u_2))^2}{1 + (3 - s_A(u_2))^2} \\ &= \frac{(3-2)^2}{1+(3-2)^2} \\ &= 0.5\end{aligned} \tag{4.24}$$

$$\begin{aligned}\mu_{A_3}(u_3) &= \frac{(3 - s_A(u_3))^2}{1 + (3 - s_A(u_3))^2} \\ &= \frac{(3-2)^2}{1 + (3-2)^2} \\ &= 0.5\end{aligned} \tag{4.25}$$

$$\vdots$$

$$\begin{aligned}\mu_{A_3}(u_{20}) &= \frac{(3 - s_A(u_{20}))^2}{1 + (3 - s_A(u_{20}))^2} \\ &= \frac{(3-3)^2}{1 + (3-3)^2} \\ &= 0\end{aligned} \tag{4.26}$$

$$\vdots$$

$$\begin{aligned}\mu_{A_4}(u_1) &= \frac{(4 - s_A(u_1))^2}{1 + (4 - s_A(u_1))^2} \\ &= \frac{(4-2)^2}{1 + (4-2)^2} \\ &= 0.8\end{aligned} \tag{4.27}$$

$$\begin{aligned}\mu_{A_4}(u_2) &= \frac{(4 - s_A(u_2))^2}{1 + (4 - s_A(u_2))^2} \\ &= \frac{(4-2)^2}{1 + (4-2)^2} \\ &= 0.8\end{aligned} \tag{4.28}$$

$$\begin{aligned}\mu_{A_4}(u_3) &= \frac{(4 - s_A(u_3))^2}{1 + (4 - s_A(u_3))^2} \\ &= \frac{(4-2)^2}{1 + (4-2)^2} \\ &= 0.8\end{aligned} \tag{4.29}$$

$$\vdots$$

$$\begin{aligned}\mu_{A_4}(u_{20}) &= \frac{(4 - s_A(u_{20}))^2}{1 + (4 - s_A(u_{20}))^2} \\ &= \frac{(4-3)^2}{1 + (4-3)^2} \\ &= 0.50\end{aligned} \tag{4.30}$$

$$\vdots$$

$$\mu_{A_5}(u_1) = \frac{(5 - s_A(u_1))^2}{1 + (5 - s_A(u_1))^2} \tag{4.31}$$
$$= \frac{(5-2)^2}{1+(5-2)^2}$$
$$= 0.9$$

$$\mu_{A_5}(u_2) = \frac{(5 - s_A(u_2))^2}{1 + (5 - s_A(u_2))^2} \tag{4.32}$$
$$= \frac{(5-2)^2}{1+(5-2)^2}$$
$$= 0.9$$

$$\mu_{A_5}(u_3) = \frac{(5 - s_A(u_3))^2}{1 + (5 - s_A(u_3))^2} \tag{4.33}$$
$$= \frac{(5-2)^2}{1+(5-2)^2}$$
$$= 0.9$$

$$\vdots$$

$$\mu_{A_5}(u_{20}) = \frac{(5 - s_A(u_{20}))^2}{1 + (5 - s_A(u_{20}))^2} \tag{4.34}$$
$$= \frac{(5-3)^2}{1+(5-3)^2}$$
$$= 0.80$$

The foregoings, based on the first constraint, calculates the degree of membership of each object(graduate) u_i in each state

In a similar manner, we proceed to calculate the degree of membership of each object(graduate) in each state.

Constraint B, μ_B: No student should be posted to their State of Studies

$$\mu_{B_j}(u_i) = \frac{(j - s_B(u_i))^2}{1 + (j - s_B(u_i))^2} \tag{4.35}$$

where $\mu_{B_j}(u_i)$ measures the degree of object u_i's membership in State j under the consideration of the *state of origin* constraint and, $s_B(u_i)$ represents the index of graduate u_i's State of Study

$$\begin{aligned}\mu_{B_1}(u_1) &= \frac{(1 - s_B(u_1))^2}{1 + (1 - s_B(u_1))^2} \\ &= \frac{(1-1)^2}{1+(1-1)^2} \\ &= 0\end{aligned} \tag{4.36}$$

$$\begin{aligned}\mu_{B_1}(u_2) &= \frac{(1 - s_B(u_2))^2}{1 + (1 - s_B(u_2))^2} \\ &= \frac{(1-2)^5}{1+(1-5)^2} \\ &= 0.94\end{aligned} \tag{4.37}$$

$$\begin{aligned}\mu_{B_1}(u_3) &= \frac{(1 - s_B(u_3))^2}{1 + (1 - s_B(u_3))^2} \\ &= \frac{(1-3)^2}{1+(1-3)^2} \\ &= 0.80\end{aligned} \tag{4.38}$$

$$\vdots$$

$$\begin{aligned}\mu_{B_1}(u_{20}) &= \frac{(1 - s_B(u_{20}))^2}{1 + (1 - s_B(u_{20}))^2} \\ &= \frac{(1-4)^2}{1+(1-4)^2} \\ &= 0.90\end{aligned} \tag{4.39}$$

$$\vdots$$

$$\begin{aligned}\mu_{B_2}(u_1) &= \frac{(2 - s_B(u_1))^2}{1 + (2 - s_B(u_1))^2} \\ &= \frac{(2-1)^2}{1+(2-1)^2} \\ &= 0.5\end{aligned} \tag{4.40}$$

$$\begin{aligned}\mu_{B_2}(u_2) &= \frac{(2 - s_B(u_2))^2}{1 + (2 - s_B(u_2))^2} \\ &= \frac{(2-2)^5}{1+(2-5)^2} \\ &= 0.9\end{aligned} \tag{4.41}$$

$$\begin{aligned} \mu_{B_2}(u_3) &= \frac{(2 - s_B(u_3))^2}{1 + (2 - s_B(u_3))^2} \\ &= \frac{(2-3)^2}{1+(2-3)^2} \\ &= 0.5 \end{aligned} \tag{4.42}$$

$$\vdots$$

$$\begin{aligned} \mu_{B_2}(u_{20}) &= \frac{(2 - s_B(u_{20}))^2}{1 + (2 - s_B(u_{20}))^2} \\ &= \frac{(2-4)^2}{1+(2-4)^2} \\ &= 0.8 \end{aligned} \tag{4.43}$$

$$\vdots$$

$$\begin{aligned} \mu_{B_3}(u_1) &= \frac{(3 - s_B(u_1))^2}{1 + (3 - s_B(u_1))^2} \\ &= \frac{(3-1)^2}{1+(3-1)^2} \\ &= 0.8 \end{aligned} \tag{4.44}$$

$$\begin{aligned} \mu_{B_3}(u_2) &= \frac{(3 - s_B(u_2))^2}{1 + (3 - s_B(u_2))^2} \\ &= \frac{(3-2)^5}{1+(3-5)^2} \\ &= 0.8 \end{aligned} \tag{4.45}$$

$$\begin{aligned} \mu_{B_3}(u_3) &= \frac{(3 - s_B(u_3))^2}{1 + (3 - s_B(u_3))^2} \\ &= \frac{(3-3)^2}{1+(3-3)^2} \\ &= 0 \end{aligned} \tag{4.46}$$

$$\vdots$$

$$\begin{aligned} \mu_{B_3}(u_{20}) &= \frac{(3 - s_B(u_{20}))^2}{1 + (3 - s_B(u_{20}))^2} \\ &= \frac{(3-4)^2}{1+(3-4)^2} \\ &= 0.50 \end{aligned} \tag{4.47}$$

$$\vdots$$

$$\begin{aligned}\mu_{B_4}(u_1) &= \frac{(4 - s_B(u_1))^2}{1 + (4 - s_B(u_1))^2} \\ &= \frac{(4-1)^2}{1+(4-1)^2} \\ &= 0.5\end{aligned} \tag{4.48}$$

$$\begin{aligned}\mu_{B_4}(u_2) &= \frac{(4 - s_B(u_2))^2}{1 + (4 - s_B(u_2))^2} \\ &= \frac{(4-2)^5}{1+(4-5)^2} \\ &= 0.5\end{aligned} \tag{4.49}$$

$$\begin{aligned}\mu_{B_4}(u_3) &= \frac{(4 - s_B(u_3))^2}{1 + (4 - s_B(u_3))^2} \\ &= \frac{(4-3)^2}{1+(4-3)^2} \\ &= 0.5\end{aligned} \tag{4.50}$$

$$\vdots$$

$$\begin{aligned}\mu_{B_4}(u_{20}) &= \frac{(4 - s_B(u_{20}))^2}{1 + (4 - s_B(u_{20}))^2} \\ &= \frac{(4-4)^2}{1+(4-4)^2} \\ &= 0\end{aligned} \tag{4.51}$$

$$\vdots$$

$$\begin{aligned}\mu_{B_5}(u_1) &= \frac{(5 - s_B(u_1))^2}{1 + (5 - s_B(u_1))^2} \\ &= \frac{(5-1)^2}{1+(5-1)^2} \\ &= 0.94\end{aligned} \tag{4.52}$$

$$\begin{aligned}\mu_{B_5}(u_2) &= \frac{(5 - s_B(u_2))^2}{1 + (5 - s_B(u_2))^2} \\ &= \frac{(5-2)^5}{1+(5-5)^2} \\ &= 0\end{aligned} \tag{4.53}$$

$$
\begin{aligned}
\mu_{B_5}(u_3) &= \frac{(5 - s_B(u_3))^2}{1 + (5 - s_B(u_3))^2} \\
&= \frac{(5-3)^2}{1 + (5-3)^2} \\
&= 0.80
\end{aligned} \tag{4.54}
$$

$$\vdots$$

$$
\begin{aligned}
\mu_{B_5}(u_{20}) &= \frac{(5 - s_B(u_{20}))^2}{1 + (5 - s_B(u_{20}))^2} \\
&= \frac{(5-4)^2}{1 + (5-4)^2} \\
&= 0.50
\end{aligned} \tag{4.55}
$$

Now, Since our sample problem set defines two Constraints, $\mu_{A_i}(u_j)$ and $\mu_{B_i}(u_j)$ *for i=1,2,...,5 and j=1,2,...,20,*
We will now proceed to estimate a single membership probability $\mu_i(u_j)$ *for i=1,2,...,5 and j=1,2,...,20,* which is estimated by the following:

$$\mu_i(u_j) = \mu_{A_i}(u_j) \cdot \mu_{B_i}(u_j) \tag{4.56}$$

$$
\begin{aligned}
\mu_1(u_1) &= \mu_{A_1}(u_1) \cdot \mu_{B_1}(u_1) && (4.57) \\
&= 0.50 \times 0 \\
&= 0 \\
\mu_1(u_2) &= \mu_{A_1}(u_2) \cdot \mu_{B_1}(u_2) && (4.58) \\
&= 0.50 \times 0.94 \\
&= 0.47 \\
\mu_1(u_3) &= \mu_{A_1}(u_3) \cdot \mu_{B_1}(u_3) && (4.59) \\
&= 0.50 \times 0.50 \\
&= 0.25 \\
&\vdots \\
\mu_3(u_4) &= \mu_{A_3}(u_4) \cdot \mu_{B_3}(u_4) && (4.60) \\
&= 0.50 \times 0.80 \\
&= 0.40 \\
&\vdots \\
\mu_5(u_{20}) &= \mu_{A_5}(u_{20}) \cdot \mu_{B_5}(u_{20}) && (4.61) \\
&= 0.80 \times 0.50 \\
&= 0.40 \\
& && (4.62)
\end{aligned}
$$

Which we can more concisely represent in the following sets:

$$\mu_i = \{\mu_i(u_1), \mu_i(u_2), \mu_i(u_3), \ldots, \mu_i(u_20)\} \qquad \textit{for i=1,2,\ldots,5} \tag{4.63}$$

So that

$$\begin{aligned}
\mu_1 &= \{\mu_1(u_1), \mu_1(u_2), \mu_1(u_3), \ldots, \mu_1(u_20)\} &(4.64)\\
&= \{0, 0.47, 0.40, \ldots, 0.90\}\\
\mu_2 &= \{\mu_2(u_1), \mu_2(u_2), \mu_2(u_3), \ldots, \mu_2(u_20)\} &(4.65)\\
&= \{0, 0, 0, \ldots, 0.40\}\\
\mu_3 &= \{\mu_3(u_1), \mu_3(u_2), \mu_3(u_3), \ldots, \mu_3(u_20)\} &(4.66)\\
&= \{0.40, 0.40, 0, \ldots, 0\}\\
\mu_4 &= \{\mu_4(u_1), \mu_4(u_2), \mu_4(u_3), \ldots, \mu_4(u_20)\} &(4.67)\\
&= \{0.40, 0.40, 0.40, \ldots, 0\}\\
\mu_5 &= \{\mu_5(u_1), \mu_5(u_2), \mu_5(u_3), \ldots, \mu_5(u_20)\} &(4.68)\\
&= \{0.84, 0, 0.72, \ldots, 0.40\}
\end{aligned}$$

4.6 Populating the States(Partitions)

Having achieved a well defined membership probability for each of the graduates into each of the states, in the previous section, we now proceed to assigning the most qualifying candidates to each state, while not exceeding each State's defined expected partition size, capacity that is. Recall that from eq(4.9) to eq(4.18), we defined the expected assignment size of each state to be:

$$\begin{aligned}
\mid p_1 \mid &= 3\\
\mid p_2 \mid &= 2\\
\mid p_3 \mid &= 1\\
\mid p_4 \mid &= 5\\
\mid p_5 \mid &= 9
\end{aligned}$$

Now, for each State,p_i, we select the first $\mid p_i \mid$ candidates, u_j with the highest membership $\mu_i(u_j)$ probability in order, for each State i=1,2,...,5:

$$p_1 = \{u_6(0.85), u_{16}(0.84), u_{14}(0.74)\} \tag{4.69}$$

$$p_2 = \{u_{10}(0.45), u_{13}(0.45)\} \tag{4.70}$$

$$p_3 = \{u_9(0.64)\} \tag{4.71}$$

$$p_4 = \{u_{12}(0.72), u_4(0.64), u_{11}(0.45), u_{17}(0.45), u_1(0.45)\} \tag{4.72}$$

$$p_5 = \{u_3(0.72), u_{19}(0.72), u_8(0.45), u_5(0.47), u_7(0.40), u_{20}(0.40), u_2(0), u_{18}(0), u_{15}(0)\} \tag{4.73}$$

After successfully applying the algorithm developed in the previous section to make the assignment on the sample problem, we have been able to reach an assignment explained thus:

4.6.1 The Assignment Table

Name	State of Origin	College	Posting	Membership Degree
GR06	Akwa-Ibom	BUK, Kano	Osun	85%
GR16	Kano	AKSU, Akwa-Ibom	Osun	84%
GR14	Kano	NSUKKA, Enugu	Osun	74%
GR10	Osun	BUK, Kano	Kwara	45%
GR13	Kano	OUI, Osun	Kwara	45%
GR09	Osun	OAU, Osun	Enugu	64%
GR12	Osun	UNILORIN, Kwara	Akwa-Ibom	72%
GR04	Kwara	UNILORIN, Kwara	Akwa-Ibom	64%
GR11	Osun	NSUKKA, Enugu	Akwa-Ibom	45%
GR17	Enugu	UNIOSUN, Osun	Akwa-Ibom	45%
GR01	Kwara	OAU, Osun	Akwa-Ibom	45%
GR03	Kwara	NSUKKA, Enugu	Kano	88%
GR19	Enugu	UNILORIN, Kwara	Kano	72%
GR08	Akwa-Ibom	UNILORIN, Kwara	Kano	72%
GR05	Akwa-Ibom	OAU, Osun	Kano	47%
GR07	Akwa-Ibom	NSUKKA, Enugu	Kano	45%
GR20	Enugu	AKSU, Akwa-Ibom	Kano	40%
GR02	Kwara	BUK, Kano	Kano	0%
GR18	Enugu	BUK, Kano	Kano	0%
GR15	Kano	UNILORIN, Kwara	Kano	0%

Table 4.1: Showing the NYSC Deployment

Chapter 5

Discussion and Conclusion

5.1 Discussion of Results

From the assignment achieved in the previous Chapter, the total optimal assignment is 17 out of 20, which is a score of 85%, with an average membership score of 51.6%. This algorithm can be assessed to have performed good on the task, nonethless, we point several points of possible improvements that can be explored to enhance the result.

5.2 Improvement Suggestions

Although, the sample problem outlined in the previous chapter got a 90% success assignment, by making a few adjustments, we observe that the accuracy can be raised as high as 100%

- When choosing objects into each partition, keep a tab on the number of non-zero assignable members left for each class, if this number is equal to the expected partition size of any class, proceed to make assignments into the class in question. This precaution ensures that each class gets the opportunity to get the maximum number assignments with fitness value different from zero.

5.3 Further Work

As must have been observed in the algorithm designed in this text, an attempt was made to solve a simplified version of the University Timetabling problem.

Here, the Constraints are of a simpler construct with no inter-constraint interaction. Such that, by simpley multiiplying out the individual constraints, we were able to estimate the degree of membership of each object in each partition. In the case of the University Timetabling problem, several constraints have more complex interactions which will be further studied.

Thank you.