Imperial College London

Department of Electrical and Electronic Engineering

Interim Project Report 2019

Project Title:	**Student Project Allocation using 0-1 Integer Programming**
Student:	**Mwanakombo Hussein**
CID:	**00936105**
Course:	**EIE4**
Project Supervisor:	**Prof. Alessandro Astolfi**
Second Marker :	**Dr Fei Teng**

Table of contents

Nomenclature

Input Parameters

T_C total number of project choices/preferences

T_L total number of lecturers

T_P total number of projects

T_{PS} total number of projects supervised

T_S total number of students

Superscripts

i student , $i \in I = \{1, \ldots, T_S\}$

j project, $j \in J = \{1, \ldots, T_P\}$

k lecturer, $k \in K = \{1, \ldots, T_L\}$

Acronyms / Abbreviations

CBC Coin-or Branch and Cut

EEE Electrical and Electronic Engineering

EIE Electronic and Information Engineering

IP Integer Programming

MILP Mixed Integer Linear Programming

NP Non-deterministic Polynomial-time

SPA-P Student Project Allocation with Preferences

SPA Student Project Allocation

Chapter 1

Project Overview

1.1 Project Specification

The student allocation problem is one faced by countless universities as research projects are usually a requirement for graduating purposes at higher education institutions.[1] The allocation of these research projects is usually modelled with a set of lecturers, student and projects. Usually the lecturers at these institutions propose available projects, and the students then choose the projects they deem acceptable as their project preferences. In some cases, lecturers can also choose the students they deem acceptable to pursue the their project. Dependent on the institution or department certain constraints are imposed on this optimization problem such as a maximum number of projects each lecturer may supervise or the minimum number of projects a student must choose as a preference. The thesis here-on-out will refer to the student project allocation problem as the 'SPA problem' [10].
As a current final year Electronic and Information engineering student at Imperial College London, I've observed the student project allocation challenge personally. As a result, I noticed that the allocation solution presented to students was feasible but not optimal as certain aspects of the algorithm were hard coded based on particular decisive factors and preferences. The current allocation solution at the department may result in certain students being unallocated at the end of the process.
The Electrical and Electronic Engineering Department at Imperial College London, aims to automate the allocation of projects to students, due to the large number of students in the graduating class and in the process, optimize the project allocation algorithm. Although the current algorithm is automated, as realized, some aspects of the optimization were hard coded hence rendering the SPA algorithm semi-automated. The need to hard coded certain constraints is a challenge as certain lecturers and students require a different allocation objective function without altering the current allocation solution. Hence, this thesis will

explore several models with distinct objective functions and use dynamic programming to implement objective functions with multiple goals which will meet the challenge of fully automating the optimization of SPA problem. This need to automate the allocation process is also observed at several other universities and their departments such as the Department of Computing at Imperial College London, the School of Computing Science at the University of Glasgow [8], the Department of Computer Science at the University of Manchester [12], School of Chemical Engineering and Analytical Science at the University of Manchester [14], the Department of Computing Science at the University of York [13], to mention a few. Automating the SPA process reduces the time and effort required to obtain a feasible solution enabling re-evaluation and re-optimization of the allocation solution when required. Automation being a necessity with solving the SPA problem, Integer Programming will be used as zero-one variables enable modelling of logical relationships and decisions in a simple manner that can be solved with powerful commercial solvers such as Gurobi [34].

1.2 Why 0-1 Integer Programming?

Existing literature and research on the SPA problem indicates the majority of the existing algorithms for the SPA problem considering student preferences, will only produce an approximate solution [2][10]. Hence, this paper seeks to utilize Integer Programming - from here-on-out denoted as IP, as it's capable of producing optimal solutions to a range of optimization problems even those that are NP-hard with optimization solvers such as Gurobi [34], CPLEX [35], GLPK [36]. These commercial solvers allow for IP models to be solved relatively fast for real-world applications as they are not open-source. Furthermore, IP formulations make it simpler to encode decisions and constraints with binary variables making it flexible and accommodating of distinct decisions. In addition, IP formulations enables multi-objective optimization [6][15].

1.3 Deliverables

The project will deliver Integer Programming optimization models that will optimize the Student Project Allocation (SPA) problem specifically for individual student project as opposed to group projects. The variant of the SPA problem this thesis will focus on will model lecturers proposing available projects and students required to provide a preference over the proposed projects as a basis. The thesis will not explore the SPA model whereby lecturers have preferences over the students who've shortlisted their proposed projects as it's a model that is not implemented often in higher education institutions for reasons delved in further in the next chapter. The thesis will then introduce complexity by further defining the SPA problem by considering constraints such as lecturer capacity and distinct decision factors.

1. **Mathematical Formulations**
   The optimization models explored will be described with mathematical formulations that aim to optimize the SPA problem based on various objective functions that a university may want to achieve, such as:
   (a) Allocate projects to students solely based on their preferences
   (b) Allocate the highest preference project to as many students as possible based on their preference rankings
   (c) Minimize the number of projects each lecturer supervises
   (d) Allocate the highest preference project to as many students as possible whilst minimizing number of projects each lecturer supervises
   (e) Allocate the highest preference project to as many students as possible whilst putting an upper bound on the number of projects each lecturer can supervise
   (f) *Opportunities of possible further models that may emerge from analysis and further literature research*

2. **Constraint Controls**
   Distinct constraint controls will be analyzed to gauge how minor changes to certain constraints impact the optimal allocation results and objective function behavior. One of the controls will be student's project preference methodologies where different project preference approaches will be explored and varied throughout the thesis. By student preference methodology, we refer to the manner a student creates a project preference list, which could be one of the following:
   - A set of project preferences with no ranking (thereby, of equal ranking)
   - A set of project preferences with ranking indicated by fixed range of integers. For example a project ranked 1 would be a student's highest preference whilst a project ranked 5 would be a student's lowest preference.
   - A set of project preferences with ranking indicated by weightings in percentages. For example a set of projects preferences weighted: $50\%, 30\%, 10\%, 20\%$, would have respective proportional impact on the allocation algorithm. Hence from above, the student indicated that their highest preference project is the

project weighted 50%. Hence, this project should have greater impact on the allocation algorithm than any of the other project preferences as their weighting are below 50%.

3. **Model Simulation**
   Each model will be simulated using MATLAB and/or python's optimization library PuLP for simulation purposes with manufactured datasets. The simulations are to test and analyze whether the designed model functions as expected. Python's optimization library PuLP is well designed to enable the utilization of multiple solvers using the same code such as: Gurobi [34], CPLEX [35], GLPK[36] etc. All the solvers mentioned prove to have detailed results that provide information such as problem solve time, upper-bounds, feasibility status which allow for further analytical diagnosis on the solutions found compared to the default PuLP solver, CBC. For these reasons, python will be the primary programming language used for simulation of the optimization models of the SPA problem.

4. **Post-optimal Analysis**
   Each model and it's respective allocation solution will be analyzed in order to establish a performance criteria to rank the effectiveness of each model with respect to their objective function:

   Analysis Aspects for each Model :

   - Speed of Convergence
   - Degree at which objective function is met i.e number of students allocated their 1st choice for a model whose objective function is to allocate the highest preference project to as many students as possible.
   - Remodelling post analysis
   - Other analysis metrics

5. **Optimal Solution Analysis**
   Post optimal analysis should yield the most effective models and these models will utilize the realistic data-set used during 2018-2019 EEE/EIE Final Year Project Allocations.

1.4 Project Extension Specification

The group project SPA problem will be explored if time allows to build an Integer Programming optimization model of the SPA problem for group projects in addition to the individual SPA problem. Group projects are challenging to model as they create a different set of constraints that are not considered in the individual case of the SPA problem.
Such constraints are:

- Maximum and minimum number of students that can be allocated to a group
- Module prerequisites or knowledge requirements needed to implement the group project as observed for 3rd year EEE/EIE students in the EEE department at Imperial College London.
- Certain group dynamic constraints that may be required for allocation such as: Student gender ratio, student department diversity ratios, cultural ratio etc.

Hence, delving into the group project SPA problem would introduce new and distinct challenges. Furthermore, group project allocation is a problem faced at several universities throughout all education years, not only during the final year as seen with the individual SPA problem.

Chapter 2

Project Planning

2.1 Implementation Plan

Required : Individual SPA problem
To implement the thesis on SPA the following needs to be done for all exploration optimization models discussed in the project specification deliverables 1 :

1. Forming the distinct SPA problem descriptions
2. Building mathematical formulations for the individual SPA
3. Modelling the mathematical formulations with software
4. Performing post-optimal analysis
5. Optimal solution analysis

Project Extension: Group SPA problem

6. Repeat the 5 actions for the group SPA problem.

Step 0 : In-progress
Step 0 : Project overview that defines the interim report requirements and background research.
Currently, step 0 has completed all interim report requirements and the background research is in-progress at 45% completion. Overall, step 0 is currently at 72% completion with a completion period for the background research component predicted as per Fig 2.1, is November 2018 - April 2019.

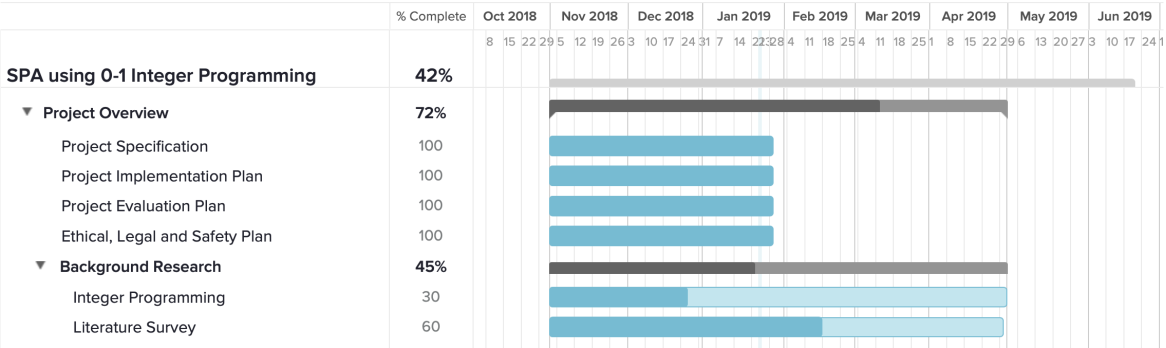

Fig. 2.1 Gantt Chart Progress: Step 0 - 72%

Step 1-2 : Complete

Step 1-2 : Forming the SPA problem descriptions and building their respective mathematical formulations.

Currently, steps 1-2 have been completed for all models listed 1, except item (f) referred to as Model 6 in Fig 2.2 since its optional. The lack of completion of Model 6 results in the 86% completion of steps 1-2 as Model 6 would result from further analysis during the project.

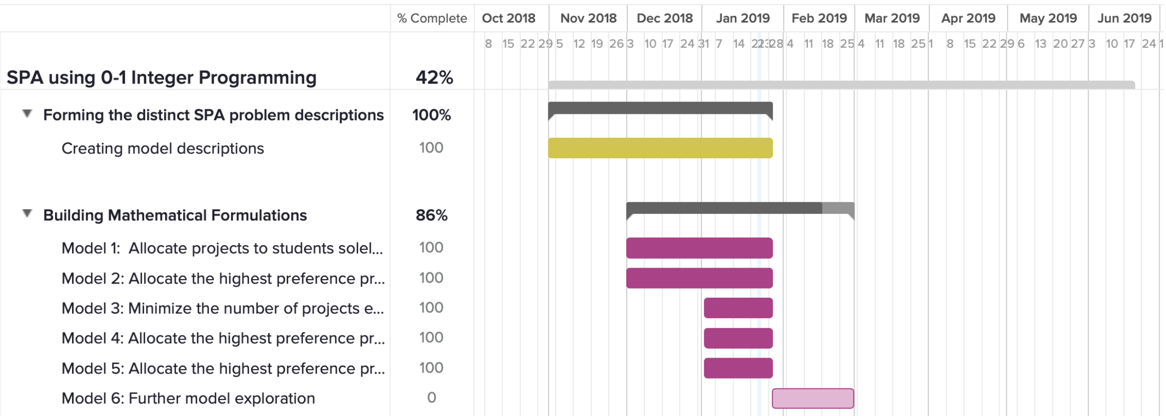

Fig. 2.2 Gantt Chart Progress: Steps 1-2 - 100% and 86% respectively

Status of Step 1 - 2 :
Forming the SPA descriptions and building their respective mathematical formulation,

- ☑ Allocate projects to students solely based on their preferences
- ☑ Allocate the highest preference project to as many students as possible based on their preference rankings
- ☑ Minimize the number of projects each lecturer supervises
- ☑ Allocate the highest preference project to as many students as possible whilst minimizing number of projects each lecturer supervises
- ☑ Allocate the highest preference project to as many students as possible whilst putting an upper bound on the number of projects each lecturer can supervise
- ☒ *Opportunities of possible further models that may emerge from analysis and further literature research*

Step 3 : Incomplete

Step 3 : Modelling the mathematical formulations with software.

Currently, step 3 is incomplete at 8% completion. Each model will be evaluated by running data scenarios to test whether the model functions as anticipated with respect to their objective function. The completion period predicted as per Fig 2.3, is January 2019 -May 2019.

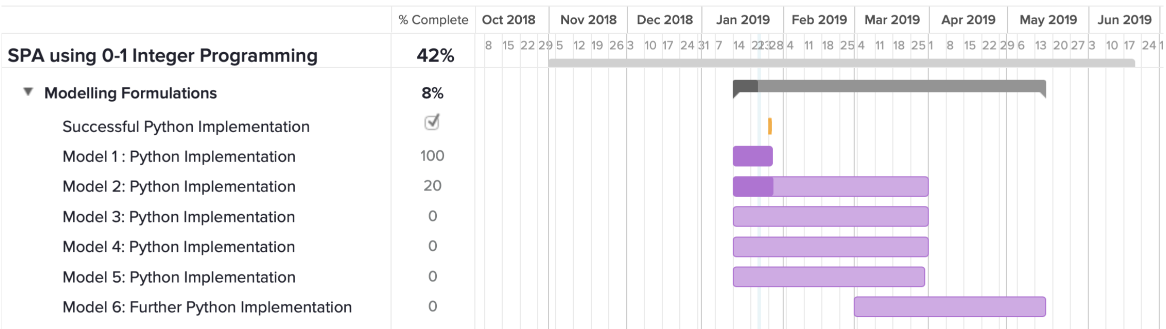

Fig. 2.3 Gantt Chart Progress: Step 3 - 8%

Status of Step 3 : Modelling the mathematical formulations with software,

- ☑ Allocate projects to students solely based on their preferences
- ☒ Allocate the highest preference project to as many students as possible based on their preference rankings

- ☒ Minimize the number of projects each lecturer supervises
- ☒ Allocate the highest preference project to as many students as possible whilst minimizing number of projects each lecturer supervises
- ☒ Allocate the highest preference project to as many students as possible whilst putting an upper bound on the number of projects each lecturer can supervise
- ☒ *Opportunities of possible further models that may emerge from analysis and further literature research*

Whilst modelling, the student preference methodology listed 2, will be explored. The outstanding methodology yet to be explored will be implemented during January 2019 - February 2019 as per Fig 2.4 below. Status of Constraint Controls,

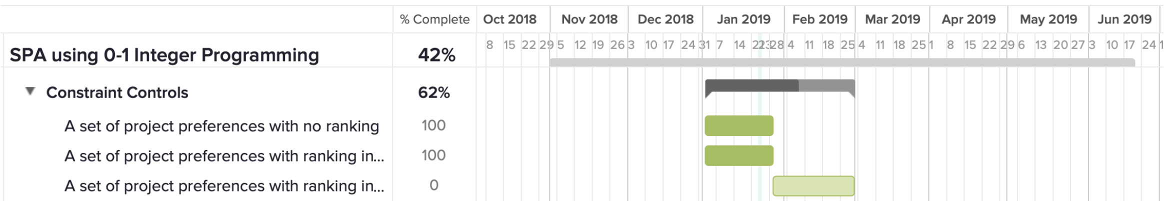

Fig. 2.4 Gantt Chart Progress: Constraint Control - 62%

- ☑ A set of project preferences with no ranking (of equal ranking)
- ☑ A set of project preferences with ranking indicated by fixed range of integers. For example a project ranked 1 would be a student's highest preference whilst a project ranked 5 would be a student's lowest preference.
- ☒ A set of project preferences with ranking indicated by weightings in percentages as listed 2.

Step 4 : Outstanding

Step 4 : Performing post-optimal analysis.

Currently, step 4 is outstanding at 0% completion as this step is dependent on the completion of step 3. The completion period predicted as per Fig 2.5, is March 2019 - April 2019.

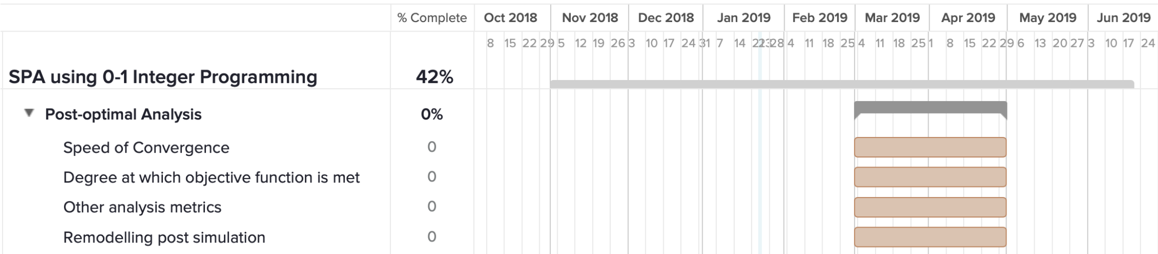

Fig. 2.5 Gantt Chart Progress: Step 4 - 0%

Status of Step 4 : Performing post-optimal analysis,

- ☒ Speed of Convergence
- ☒ Degree at which objective function is met
- ☒ Remodelling post simulation
- ☒ Other analysis metrics

Step 5 : Outstanding

Step 5 : Optimal solution analysis

Currently, step 5 is outstanding at 0% completion as this step is dependent on step 4. This step will be done concurrently with step 4 as the results from step 4 will infer analytical decisions in step 5, hence the completion period predicted as per Fig 2.6, is also March 2019 - April 2019. Status of Step 5 : Optimal solution analysis,

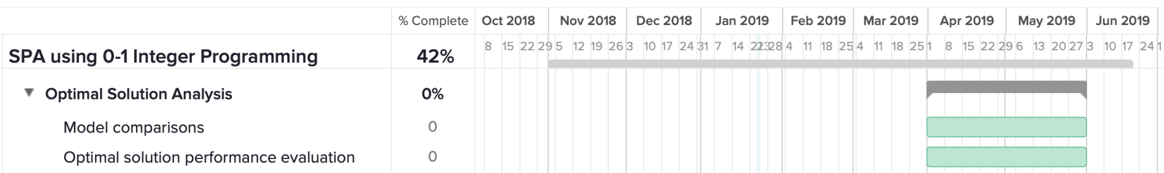

Fig. 2.6 Gantt Chart Progress: Step 5 - 0%

- ☒ Model optimal solution comparisons
- ☒ Optimal solution performance evaluation against current allocation implementation in EEE department

Overall Implementation Plan:

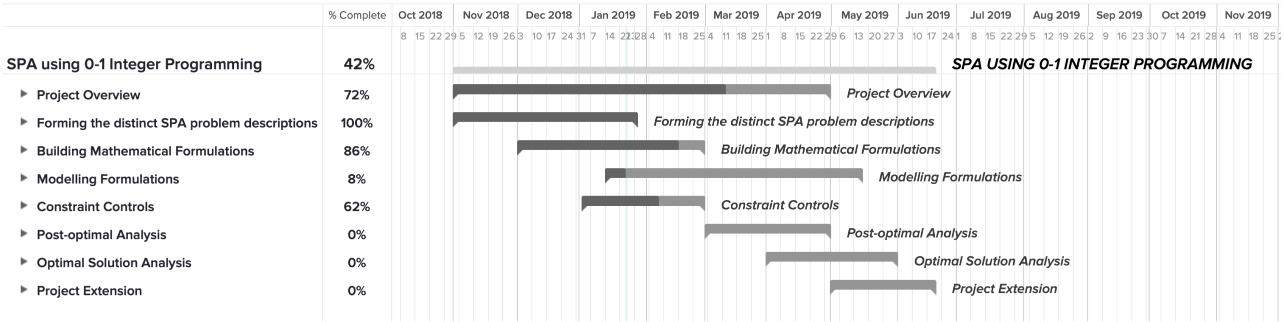

Fig. 2.7 Overall Project Progress : 42%

Milestones

- ✎ Background Research
- ☑ Mathematical formulation of the basic SPA problem
- ☑ All individual SPA models mathematically formulated
- ☑ Implementation of basic SPA problem in python
- ☒ All individual SPA models implemented with python
- ☒ Post-optimal analysis
- ☒ Optimal solution analysis
- ☒ Project Extension

Overall, as can be seen in Fig 2.2 - 2.6, the project is at 42% completion currently with the target of completion as mid June 2019.

Fallback position

If any stage of development is delayed or goes awry the fallback position would be to only implement the five individual SPA problems and not explore further models as indicated 1 as item (f). Furthermore, the project extension detailed in section 1.4, will not be explored.

2.2 Evaluation Plan

My thesis optimal allocation solution to the individual SPA problem will be gauged and compared against the current allocation algorithm currently being used in the EEE department at Imperial College London. The current algorithm may render certain students as 'unallocated' whereas, my optimal allocation solution strives to allocate all students and have zero unallocated students.

Evaluation Metrics:

- Number of students allocated their 1st preference
- Number of students allocated their 1st, 2nd or 3rd preference
- Number of students rendered unallocated.
- Maximum number of projects each lecturer will supervise

In order to evaluate the project against the metrics, the following experiments will be implemented.

- Modelling each mathematical formulation with python utilizing different data scenarios to test whether the model functions as expected.
- Remodelling and re-implementation based on results if modification is required to the initial mathematical formulation of the model or perhaps the problem description as detailed in Fig 2.8.
- Modelling each mathematical formulation with python utilizing a 'realistic data-set'.The 'realistic data-set' refers to the 2018-2019 EEE/EIE final year project allocations. This enables explicit evaluation with respect to current SPA implementation.

The methodology for testing the models listed 1, is as follows:

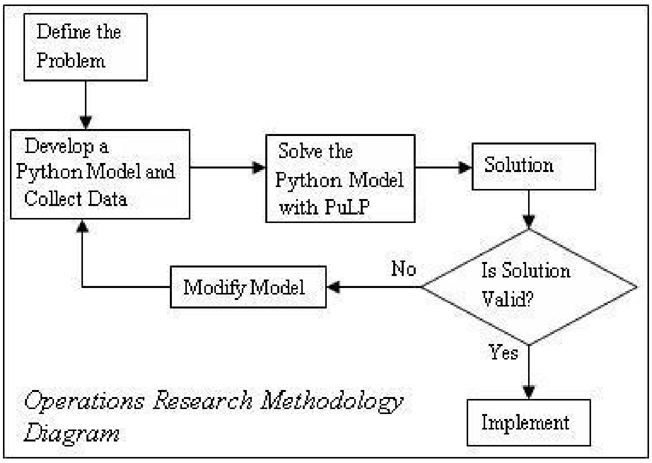

Fig. 2.8 Model Implementation Methodology [21]

To deem the project successful, the optimal allocation solution provided by the thesis should improve on the evaluation metrics 2.2, compared to the current allocation implemented in the EEE department at Imperial College London.

2.3 Ethical, Legal, and Safety Plan

The *Student Project Allocation using 0-1 Integer Programming* thesis will not require any laboratory use or hardware. Due to the software nature of the thesis, all the possible considerations listed below are not an ethical, legal or safety issue throughout the implementation of the project.

Considerations taken into account:

- *Electrical Safety* - No electrical components will be used except the charging supply of the laptop used for software development. To safeguard against electrical shocks, only a certified charging cable will be utilized for charging the laptop.
- *Physical Safety* - No fast-moving objects are built or utilized during this project.
- *Chemical Safety* - No chemical materials are utilized during this project.
- *Fire Safety* - No use of batteries or a power supply that causes element heating.
- *Biological Safety* - No biological materials are utilized during this project.

- *Animal Safety* - No animals are utilized during this project.
- *Appliance Safety* - No portable lab equipment utilized during this project.
- *Airspace Safety* - No drones or rockets utilized during this project.
- *Study Participant Safety* - People will not be required for any surveys or tests during this project.
- *Data Infrastructure Safety* - No access to College or student IT systems will be required for this project.

With the considerations above, it's observed with the respective explanations as to why they will not apply to the project and hence render the project free of any ethical, legal or safety issues.

Chapter 3

Background

3.1 Literature Overview

The generalization of the SPA problem is the allocation of junior doctors to hospitals nationwide in the US [5] coined as the Hospitals/Residents problem (HR) [3]. The HR problem is a two-sided matching problem where the aim is to form a stable matching of residents to hospitals considering constraints such as hospital capacities and the preference of residents over hospitals and vice versa [10]. As such,the generalization of the SPA problem can be applied to several real-world challenges such as allocation students to secondary schools and higher education institutions across Europe [4]. Hence, exploring the optimal solution of the SPA problem would have a definite impact.
There are two Student Project Allocation problem models recurrent in literature: the first simply considers the student's preferences [6][7][8] and the second incorporates the preference of the lecturers as well [9][3]. The first variant views the SPA problem as a resource allocation problem whereby the finite resource are the projects and lecturers, [1] and the second variant as a matching problem whereby the challenge is to allocate a set of agents to another set of agents based on preferences [2],[3]. The latter scenario is one whereby the SPA problem defines that students have preferences over proposed projects whilst the lecturers have preferences over the students. It has been proved when both students and lecturers have preferences then the critical property the matching should satisfy is stability [10].
Conducting an interview with the Head of the Student Project Allocation Algorithm at the EEE department at Imperial College London, Dr. Tom Clarke, depicts it's not desirable to impose a constraint that lecturers have preference over students. The reason being it imposes unfair project allocation conditions on certain students if the lecturer's student preference for instance, is deemed solely on academic merit. It's simple to deduce that students who are

less academically inclined would be less likely to be allocated to their preferable projects if they are also preferable to more academically inclined students. This reasoning is also seen at other institutions such as the University of Manchester [14], whereby conducting a survey yields that there is a high student dissatisfaction rate experienced especially by less academically-inclined students with the latter approach. This two-sided matching introduces new and distinct complications such as what factors will be used to derive the lecturer preference list and how will the factors affect the optimal allocation solution without bias. However, it should also be considered that exceptions can be made to this situation and certain projects would require a two-sided matching approach as occurred with one project proposal during this year's project allocation process for the EEE/EIE 2019 graduates. This is attributed to the fact that some projects require certain skills and prerequisites of knowledge in which case a lecturer would have a preference list on the student they deem acceptable to take on their project. Whether or not lectures have preferences over students depends on different institutions and departments alike. As mentioned in the project specification, this thesis will focus on the variant of the SPA problem whereby the student imposes a preference on projects only. Hence, the SPA problem in this thesis will not be a matching problem but a resource allocation optimization problem which can also be viewed as an assignment problem [14].

Of the two most recurrent SPA models in literature, Anwar and Bahaj [6] follow the first SPA model in literature mentioned. They only consider the student's preference as they proposed two Integer Programming models: 1) An individual project model that requires students to select a subset of the proposed projects and rank their choices. The model aims to make lecturer workload in the supervision of individual student projects as even as possible and also strives to allocate students their first project preference. This was implemented with dynamic programming that defined two models: one that minimized the lecturer workload and another that maximized the student preferences, in that order. The two models were combined sequentially such that objective function result of the previous model acted as a constraint for the following model. This is dynamic programming approach was deemed effective. The second IP model, 2) A group project model that has the same requirements of students as mentioned in the individual model and, by optimally allocating each individual student a project, implicitly allocates projects to groups of students. It's worth noting that although the SPA problem is one that has been researched through out the years, the concept of approaching the SPA problem using Integer Programming is seen rarely in literature.

It's observed that Anwar and Bahaj combined two distinct objective functions to achieve a dual goal objective function. Pan et al [15], expanded on the multi-objective concept by defining a goal programming formulation with three objective functions. The three objective

functions : 1) Maximizing the number of students allocated, 2) Maximizing students' satisfaction and 3) Maximizing supervisors' satisfaction, in that order. Similar to Anwar and Bahaj's approach, these three objective functions were sequentially solved considering the previous model's objective function solution as the constraint of the following model.
Other researchers follow the second SPA model in literature viewing the SPA problem as a matching problem and strive to find stable matching between student and projects such as Abraham, Irving and Manlove [3]. They proposed two algorithms that find a stable matching in the SPA problem where students have preferences over projects, whilst lecturers have preferences over students. The stable matching produced by the first algorithm is student-optimal (that is, students have the best possible projects that they could obtain in any stable matching) while the one produced by the second algorithm is lecturer-optimal (that is, lecturers have the best possible students that they could obtain in any stable matching).
Manlove and O'Malley's [10] research also viewed the SPA problem as a matching problem similar but different to the second SPA model in literature, as they did not explore the SPA problem where students have preferences over projects and lecturers have preferences over students.
Instead Manlove and O'Malley [10] explore an SPA problem where both students and lecturers have preferences over projects which is distinct to the approach usually observed in literature. Their paper considers the variant of SPA in which lecturers rank in strict order of preference the projects that they offer which implicitly implies that each lecturer is indifferent among those students who find acceptable a given project that he/she offers. They also show that stable matchings can have different sizes,and prove that finding the maximum cardinality stable matching referred to as MAX-SPA-P is NP-hard and not approximable within δ, for some $\delta > 1$, unless P=NP. Lastly, they give an approximation algorithm with a performance guarantee of 2 for MAX-SPA-P further showing that an integer programming approach is more effective as it would result in an optimal solution rather than an approximation especially for problems that are NP-hard. Hence, reiterating why following an integer programming approach is far more effective when solving the SPA problem rather than a linear programming approach.
A completely different approach to either of the two recurrent SPA models in literature is one R. Calvo-Serrano et al [14] proposed, a mixed-integer linear programming (MILP) model for the SPA problem. Their MILP model is uniquely based on a flexible definition of the SPA problem that allocates students to project categories and project supervisors, rather than to projects only as seen in previous research. This aims to maximize student satisfaction by fulfilling two criteria rather than one: Allocating students to their preferred supervisor

and/or to their preferred project category, which provides the usual SPA model with more flexibility leading to better solutions that maximize student satisfaction.
Other than the IP SPA approach Anwar and Bahaj [6] brought to the table, the only other IP SPA research approach I could acquire in literature is one written by Manlove, Mline and Olaosebika [2]. Their IP SPA model enables MAX-SPA-P to be solved optimally and present a correctness result. By comparing the feasible solutions produced by the approximation algorithms and the optimal solution produced by their IP model w.r.t the size of that table matching constructed, their paper shows that the $\frac{3}{2}-$approximation finds stable matching that are very close to maximum cardinality. Hence, showing that an IP model can run SPA-P instances that appear in practice, to find maximum cardinality matchings that admit no blocking pair.

Chapter 4

Individual Student Project Allocation Modelling

The interim report will only include the first of the five completed individual SPA models as it's been fully implemented and tested. The other four models have implementation and testing pending as indicated in the Planning Section, Step 3 Status(2.1)
The modelling process starts with a well-defined model description. The model description is then translated into a mathematical program. The next step is to use solvers such as Gurobi [34], CPLEX [35], GLPK [36], PuLP's CBC [37] to solve the model. The solution is then utilized to make decisions that support the model description.
The modelling process can be broken down into five steps[21]:

1. Forming the problem description
2. Formulating the mathematical program
3. Solving the mathematical program
4. Performing post-optimal analysis
5. Presenting the solution and analysis

The first two steps will be explored in this chapter whilst the last three steps will be explored in the following chapter.

4.1 Model 1: Naively allocate projects to students based solely on their preferences

Step 1 - Problem Description:
As mentioned in Section 2.1, there two main recurrent SPA models in literature : The first simply considers the student's preferences and the second incorporates the preference of the lecturers as well. Model 1 follows from the first model recurrent in literature, the model allocates projects to students from a subset of project preferences the students shortlist from the entire list of project proposals. This model is naive as it's far from optimal but allows for an intuitive problem formulation which will then be improved for optimality. Model 1 is designed such there are T_S students. Each student can choose a subset of T_C projects from T_P proposed projects. In this model, a student can only be allocated to one project and a project can only be allocated to one student - although in distinct real-world applications such as group project allocations, a project can be allocated to multiple students.
Where,

- T_S is the total number of students.
- T_L is the total number of lecturers.
- T_C is the total number of projects required to be ranked by each student.
- T_P is the total number of projects proposed by lecturers.
- T_{PS} is the total number of projects supervised by each lecturer.

Step 2 - Mathematical Program Formulation:

- <u>*Decision Variables*</u>
  - $X_{i,j}$: A student can only be allocated a project that has been proposed by a lecturer. This decision can be defined with the binary variable, $X_{i,j}$:

$$X_{i,j} = \begin{cases} 1, & \text{if student } i \text{ is allocated to project } j \\ 0, & \text{otherwise} \end{cases} \tag{4.1}$$

Note : The index, i represents the i^{th} student, $\{i|1 \leq i \leq T_S, i \in \mathbb{Z}\}$. The index, j represents the j^{th} project, $\{j|1 \leq j \leq T_P, j \in \mathbb{Z}\}$

- $C_{i,j}$: A student can only be allocated a project that has been chosen as one of their preferences.

$$C_{i,j} = \begin{cases} 1, & \text{if project j is chosen as a preference by student i} \\ 0, & \text{otherwise} \end{cases} \tag{4.2}$$

- <u>*Constraints*</u>

1. **Each student should only be allocated one project** Where student i has to be allocated to 1 of the total T_P proposed projects,

$$X_{i,1} + \cdots + X_{i,T_P} = 1 \tag{4.3}$$

This is generalized as:

$$\sum_{j=1}^{T_P} X_{i,j} = 1 \tag{4.4}$$

2. **Each project should be allocated to at-most one student**

$$\sum_{i=1}^{T_S} X_{i,j} \leq 1 \tag{4.5}$$

3. **Each student can select a maximum of Tc projects as part of their preferences subset**
Where, student i has to choose T_C project preferences,

$$C_{i,1} + \cdots + C_{i,T_P} = T_C \tag{4.6}$$

Notice, if project j, is not chosen by student i then $C_{i,j} = 0$ and project j would not be applicable for allocation to student i. This is generalized as:

$$\sum_{j=1}^{T_P} C_{i,j} = T_C \tag{4.7}$$

4. **Each student can only be allocated to a project that is part of their preferences subset**
   Out of the T_c possible preferences, student i can only be allocated to 1 of their project preferences.

$$C_{i,1} \times X_{i,1} + \cdots + C_{i,T_P} \times X_{i,T_P} = 1 \tag{4.8}$$

   This is generalized as:

$$\sum_{j=1}^{T_P} C_{i,j} \times X_{i,j} = 1 \tag{4.9}$$

- <u>*Objective Function*</u>

$$Z_1 = \sum_{i=1}^{T_S} \sum_{j=1}^{T_P} (C_{i,j} \times X_{i,j}) \tag{4.10}$$

Reference

[1] H. O. Salami and E. Y. Mamman, "A Genetic Algorithm for Allocating Project Supervisors to Students," International Journal of Intelligent Systems and Applications, vol. 8, (10), pp. 51-9, 2016. Available: http://dx.doi.org/10.5815/ijisa.2016.10.06. DOI: 10.5815/ijisa.2016.10.06.
[2] D. Manlove, D. Milne and S. Olaosebikan, "An integer programming approach to the student-project allocation problem with preferences over projects," in 5th International Symposium on Combinatorial Optimization, ISCO 2018, April 11, 2018 - April 13, 2018, pp. 313-325.
[3] D.J. Abraham, R.W. Irving and D.F. Manlove, "Two algorithms for the Student- Project allocation problem." Journal of Discrete Algorithms, pp. 79-91, 2007.
[4] K. C. Ágoston, P. Biró and R. Szántó, "Stable project allocation under distributional constraints," Operations Research Perspectives, vol. 5, pp. 59-68, 2018. Available: https://www.sciencedirect.com/science/article/pii/S2214716018300022. DOI: 10.1016/j.orp.2018.01.003.
[5] A. E. Roth, "The Evolution of the Labor Market for Medical Interns and Residents: A Case Study in Game Theory," Journal of Political Economy, vol. 92, (6), pp. 991-1016, 1984. Available: https://www.journals.uchicago.edu/doi/10.1086/261272. DOI: 10.1086/261272.
[6] A. A. Anwar and A. S. Bahaj, "Student project allocation using integer programming," IEEE Transactions on Education, vol. 46, (3), pp. 359-67, 2003. Available: http://dx.doi.org/10.1109/TE.2003.811038. DOI: 10.1109/TE.2003.811038.
[7] P.R. Harper et al, "A genetic algorithm for the project assignment problem." Computers and Operations Research, pp. 1255-1265, 2005.
[8] A. Kwanashie et al, "Profile-based optimal matchings in the Student–Project Allocation problem." In Proceedings of IWOCA '14: The 25th International Workshop on Combinatorial Algorithms, Volume 8986 of Lecture Notes in Computer Science, pp. 213-225, 2015.
[9] D. Kazakov, "Co-ordination of student-project allocation," 2001.
[10] D. F. Manlove and G. O'Malley, "Student-Project Allocation with preferences over Projects," Journal of Discrete Algorithms, vol. 6, (4), pp. 553-560, 2008. Available: https://www.sciencedirect.com/science/article/pii/S1570866708000476. DOI: 10.1016/j.jda.2008.07.003.
[11] K. Iwama, S. Miyazaki and H. Yanagisawa, "Improved approximation bounds for the student-project allocation problem with preferences over projects." Journal of Discrete Algorithms,13, pp. 59-66, 2012.
[12] S. Lightfoot, "Solving the Student Project Allocation Problem", Studentnet.cs.manchester.ac.uk, 2016. [Online]. Available: http://studentnet.cs.manchester.ac.uk/resources/library/3rd-year-projects/2016/samuel.lightfoot.pdf. [Accessed: 21- Jan-2019].
[13] J. Dye, "A Constraint Logic Programming Approach to the Stable Marriage Problem and its Application to Student Project Allocation," University of York, 29 June, 2001.
[14] R. Calvo-Serrano et al, "Mathematical programming approach for optimally allocating students' projects to academics in large cohorts," Education for Chemical Engineers, vol. 20, pp. 11-21, 2017. Available: https://www.sciencedirect.com/science/article/pii/S1749772817300301. DOI: 10.1016/j.ece.2017.06.002.

[15] Li Pan Sydney C. K. Chu Guangyue Han and Joshua Z. Huang, "Multi-criteria Student Project Allocation: A Case Study of Goal Programming Formulation with DSS Implementation," 2009.
[16] Hamza O Salami and Esther Y Mamman, "A Genetic Algorithm for Allocating Project Supervisors to Students," International Journal of Intelligent Systems and Applications, vol. 8, (10), pp. 51-59, 2016. Available: https://search.proquest.com/docview/1884171923. DOI: 10.5815/ijisa.2016.10.06.
[17] R. Hoshino and C. Raible-Clark, "The quest draft: An automated course allocation algorithm," in 28th AAAI Conference on Artificial Intelligence, AAAI 2014, 26th Innovative Applications of Artificial Intelligence Conference, IAAI 2014 and the 5th Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, July 27, 2014 - July 31, 2014, pp. 2906-2913.
[18] Y. Gu et al, "Student admission matching based content-cache allocation," in 2015 IEEE Wireless Communications and Networking Conference, WCNC 2015, March 9, 2015 - March 12, 2015, pp. 2179-2184.
[19] R. Garfinkel, Integer Programming. New York; London: Wiley, 1972.
[20] (Sept 26). Allocating final year projects to students. Available: https://vknight.org/unpeudemath/math/2017/09/26/allocating-final-year-projects-to-students.html.
[21] (Jan 15 2019). Optimization with PuLP . Available: https://pythonhosted.org/PuLP/.
[22] S. Varone and D. Schindl, "Course opening, assignment and timetabling with student preferences," in 2nd International Conference on Operations Research and Enterprise Systems, 2013, pp. 153-8.
[23] D. Gale and L. S. Shapley, "College Admissions and the Stability of Marriage," The American Mathematical Monthly, vol. 120, (5), pp. 386-391, 2013. Available: https://www.jstor.org/stable/10.4169/amer.math.monthly.120.05.386. DOI: 10.4169/amer.math.monthly.120.05.386.
[24] R. W. Irving, D. Gusfield and P. Leather, "An Efficient Algorithm for the “Optimal” Stable Marriage " vol. 34, .
[25] B. Kassa, "A linear programming approach for placement of applicants to academic programs," SpringerPlus, vol. 2, (1), pp. 1-7, 2013. Available: https://www.ncbi.nlm.nih.gov/pubmed/24386626. DOI: 10.1186/2193-1801-2-682.
[26] D. F. Manlove and F. Cooper, "Two algorithms for the Student-Project Allocation problem," Journal of Discrete Algorithms, vol. 5, (1), pp. 73-90, 2007. Available: https://www.sciencedirect.com/science/article/pii/S1570866706000207. DOI: 10.1016/j.jda.2006.03.006.
[27] S. Pudaruth, M. Bhugowandeen and V. Beepur, "A Multi-Objective Approach for the Project Allocation Problem," International Journal of Computer Applications, vol. 69, (20), pp. 26-30, 2013. . DOI: 10.5120/12088-8256.
[28] A. H. Chown, C. J. Cook and N. B. Wilding, "A simulated annealing approach to the student-project allocation problem," American Journal of Physics, vol. 86, (9), pp. 701-708, 2018. Available: http://dx.doi.org/10.1119/1.5045331. DOI: 10.1119/1.5045331.
[29] D. Gusfield and R. W. Irving, "TheStable Marriage Problem", Cambridge, Mass. MIT Press,1989.
[30] S. Olaosebikan and D. Manlove, "Super-stability in the Student-Project Allocation Problem with Ties," 2018. Available: https://www.openaire.eu/search/publication?articleId=od18::1220ef66e6dd97e0a0b6c88ffa136821.

[31] E. M. Nazri et al, "Selecting university undergraduate student activities via compromised-analytical hierarchy process and 0-1 integer programming to maximize SETARA points," in 13th IMT-GT International Conference on Mathematics, Statistics and their Applications (ICMSA2017), 2017, Available: http://dx.doi.org/10.1063/1.5012211. DOI: 10.1063/1.5012211.
[32] Y. Cheung, G. Meng Hong and K. Keng Ang, "A dynamic project allocation algorithm for a distributed expert system," Expert Systems with Applications, vol. 26, (2), pp. 225-232, 2004. Available: https://www.sciencedirect.com/science/article/pii/S0957417403001374. DOI: 10.1016/S0957-4174(03)00137-4.
[33] Anonymous "MATLAB," .
[34] Anonymous "Gurobi Optimization," .
[35] (Jan 15 2019). CPLEX Optimizer. Available: https://www.ibm.com/analytics/cplex-optimizer.
[36] (Jan 15 2019). GLPK (GNU Linear Programming Kit). Available: https://www.gnu.org/software/glpk/.
[37] (Jan 15 2019). CBC (Coin-or branch and cut). Available: https://projects.coin-or.org/Cbc.
[38] L. G. Proll, A Simple Method of Assigning Projects to Students. 197223.
[39] M. Chiarandini, R. Fagerberg and S. Gualandi, "Handling preferences in student-project allocation," Annals of Operations Research, pp. 1-40, 2017. Available: https://search.proquest.com/docview/1965733980. DOI: 10.1007/s10479-017-2710-1.
[40] F. Diebold and M. Bichler, "Matching with indifferences: A comparison of algorithms in the context of course allocation," European Journal of Operational Research, vol. 260, (1), pp. 268-282, 2017. Available: https://www.sciencedirect.com/science/article/pii/S037722171631030X. DOI: 10.1016/j.ejor.2016.12.011.
[41] V. Sanchez-Anguix et al, "A near Pareto optimal approach to student–supervisor allocation with two sided preferences and workload balance," Applied Soft Computing Journal, vol. 76, pp. 1-15, 2019. Available: https://www.sciencedirect.com/science/article/pii/S1568494618306811. DOI: 10.1016/j.asoc.2018.11.049.
[42] M. I. Moussa and A. H. A. El-Atta, "A Visual Implementation of Student Project Allocation," International Journal of Computer Theory and Engineering, pp. 178-184, 2011. . DOI: 10.7763/IJCTE.2011.V3.302.
[43] F. Cavdur et al, "A two-phase binary-goal programming-based approach for optimal project-team formation," Journal of the Operational Research Society, pp. 1-18, Apr 25, 2018.
[44] T. A. M. Toffolo et al, "An integer programming approach to the multimode resource-constrained multiproject scheduling problem," Journal of Scheduling, vol. 19, (3), pp. 295-307, 2016.