International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Evaluating Effectiveness of Student Final Year Project Allocation: A Case Study at Universiti Brunei Darussalam

Nur Qalbi Hj. Mohd. Salleh[1], Shiaw Yin Yong[2]

[1]Bachelor of Science graduate, Mathematical Sciences, Faculty of Science, Universiti Brunei Darussalam, Brunei Darussalam
[2]Assistant Professor, Mathematical Sciences, Faculty of Science, Universiti Brunei Darussalam, Brunei Darussalam

Abstract
The allocation of final year projects is a critical task for universities to ensure that students receive the best possible academic experience. The allocation problem is often complex, and universities face several challenges, including balancing staff workload, taking into consideration students' preferences as well as allocating resources optimally. This paper presents a Linear Programming approach for optimising student final year project allocation to address the mentioned challenges. By considering students' preferences, staff workload, and project quality, the proposed model provides an optimal and fair project allocation scheme. The model's effectiveness was demonstrated through a case study in Mathematical Sciences, Faculty of Science, UBD, using real data. The results obtained from the LP model are both feasible and optimal, indicating that the approach provides an unbiased allocation of supervisors to project students satisfying all constraints. The model allows for flexibility in fine-tuning the allocation scheme, as it can accommodate different levels of quality preferences and student priorities. It is believed that the proposed LP model can provide necessary support for the project coordinator to make informed decisions regarding final year project allocation. The model can also save time and resources by providing solution quickly and efficiently.

Keywords: Project allocation, Linear Programming, optimisation, constraints

1. Introduction

Final Year Projects (FYPs) are often considered a crucial component of many higher education programs, as they provide students with an opportunity to exhibit their knowledge and skills acquired throughout their academic program by working on a substantial project in a particular subject area. These projects are generally challenging, requiring students to apply the theories, concepts, and technical skills they have learnt. The successful completion of a FYP can be a significant accomplishment, providing students with valuable experience, skills, and knowledge that can have a positive impact on their future prospects and career endeavours. As such, selecting a final year project and supervisor can be competitive among students, who may face difficulty in securing their preferred choices. On the other hand, allocation of FYPs is a complex task for universities that must balances the preferences and needs of students with the availability and expertise of academic staffs. This involves ensuring that the allocation process is transparent, fair, and takes into account staffs' university workload and availability of resources. Allocating FYPs becomes increasingly difficult as the number of students grows [2], making it harder to ensure equal opportunities for all students.

Every semester at Universiti Brunei Darussalam (UBD), there are typically between 15 and 75 students on the Mathematical Sciences bachelor degree, and fewer than 12 academic staff members available for project supervision. Over the years, the department has adopted an equitable approach for allocating FYPs to students. This approach disregards students' past performance, and solely takes into account their preferences for the supervisor. To facilitate this student-oriented approach to project allocation, the department requires each student to submit a list ranking academic staff members, based on their preference, as their potential project supervisors. The student-supervisor allocation process has been done manually by a single program coordinator. This process is intensive and time consuming, particularly when dealing with large cohorts, and when one or two academic staff members are excessively in demand during the ranking process. Automating the project allocation process can be helpful to streamline the allocation process, and reduce the time and effort required to allocate supervisors to respective students. In this paper, a simplified linear programming model is proposed to address the student-oriented allocation problem in UBD, and then conduct ROC analysis to assess the effectiveness of the model. This study is a component of the development plan for an automated system that will be utilised by the department.

2. Student Project Allocation (SAP) problem

Student Project Allocation (SAP) problem has a long history in the educational community, and the problem has been studied by researchers and practitioners in various fields, including Operations Research (OR), Computer Science, and Education, etc. The SPA problem was first formulated as a combinatorial optimisation problem, or better known as an assignment problem [6, 9]. The Hungarian algorithm is a classical algorithm that finds the optimal assignment solution in polynomial time, but assumes that the number of students and projects is equal. Later, as SPA became more widely studied, researchers began to incorporate additional realistic features into the problem to make the allocation process more representative of the actual environments. These additional features often increased the complexity of the problem and made it more challenging to compute an optimal solution. Teo, et al. (1998) proposed a two-phase approach modified from the Hungarian Algorithm. The algorithm first created a preference matrix that mapped students' project area preferences to supervisors' availability. Then, the algorithm used the preference matrix to determine the assignment of projects to students using branch and bound techniques, resulting in an efficient allocation of student projects at NTU. Other significant developments in the solution of the SAP includes application of Linear Programming [1, 3], and heuristic algorithms such as Greedy algorithms [8], simulated annealing [4], genetic algorithms [5, 12], and hybrid algorithms that combine both greedy and simulated annealing techniques [7].

Heuristic algorithms improve an initial allocation solution iteratively employing specific search strategies tailored to the problem's characteristics. The algorithms continue to refine the initial solution until no further improvement is possible or when a termination condition is met. The greedy algorithm works by greedily selecting the best candidate for each allocation decision, but this approach may sometimes result in suboptimal allocation [11]. Simulated annealing algorithms and genetic algorithms are more sophisticated; these algorithms explore the search space more efficiently by using probabilistic and evolutionary search strategies. Simulated annealing algorithms use random moves to explore the search space and gradually decrease the randomness over time. This approach allows the algorithm to break out of local optima and find better solutions, but it can be computationally expensive. Genetic algorithms use principles of evolution to explore the search space by generating a population of potential solutions and continually improving them through processes such as mutation, selection, and crossover. This approach can quickly converge on high-quality solutions, but can also be computationally expensive. In general, heuristic algorithms provide efficient and scalable solutions to the SPA problem, and can be effective for solving NP-hard problems that are difficult to solve using exact methods. These methods can handle large problem instances and quickly coverage to good-quality solutions. However, they may not guarantee optimality in all cases, and may require tuning of multiple parameters, which can be time consuming.

On the other hand, exact methods under linear programming such as integer programming (IP) and constraint programming (CP) can guarantee optimality and can handle smaller problem instances. However, they may be computationally expensive and may not scale up well to larger problem sizes. In general, there is no single method of project allocation that can be considered most effective for all situations. Each method has its own strengths and weaknesses, and the effectiveness of a method can vary depending on the specific characteristics and constraints of the problem.

3. Problem formulation

The problem formulation process involves creating a mathematical model that incorporates all the constraints and objectives of the problem, and using algorithms to determine the optimal solution. The student-supervisor allocation problem in UBD is a polynomial problem, therefore a simplified linear programming model provides a fast and efficient approach to optimise student-supervisor assignment while satisfying all constraints.

Let us define S as the set of students where $i = \{1,2,\cdots,S\}$, and P be the set of academic staff members where $j = \{1,2,\cdots,P\}$, the final year student allocation problem can be formulated as a linear programming problem with the following binary decision variable:

$x_{ij} = 1$ if student i is assigned to staff j, and
$x_{ij} = 0$ otherwise

The objective function of the model is to minimise the sum of priority, given by

$$\text{Min } \sum_{i=1}^{S}\sum_{j=1}^{P} C_{ij}x_{ij} \qquad (1)$$

where C_{ij} is the preference rating of student i for staff j.

The constraints include:

- Each student can be assigned to only one supervisor, given by

$$\sum_{j=1}^{P} x_{ij} = 1, \text{ for } i = 1,2,\cdots,S \qquad (2)$$

- All students are assigned a supervisor, given by

$$\sum_{i=1}^{S} x_{ij} = S, \text{ for } j = 1,2,\cdots,P \qquad (3)$$

- Individual staff workload is taken into consideration, given by

$$\sum_{i=1}^{S} x_{ij} \leq W_j, \text{ for } j = 1,2,\cdots,P \qquad (4)$$

where W_j denotes the maximum workload capacity of staff j

- Each student-supervisor combination is non-negative, given by

$$x_{ij} \geq 0, \text{ for } i = 1,2,\cdots,S; j = 1,2,\cdots,P \qquad (5)$$

4. Computational result and discussion

The proposed algorithm for student-supervisor allocation was implemented in R using the lpsolve package, and the results were obtained on a real dataset of 34 students and 9 academic staff members. Figure 1 presents the frequencies of students' selections for each of the 9 academic staff members as their 1st and 2nd preferences. It appears that P8 was the most popular among students as potential supervisor. Despite the variation in student preference across the academic staff members, the algorithm was able to provide an optimal solution that minimised the workload imbalances among the staff members. The allocation process was deemed successful, with an objective function value of 60 indicating that the constraints were met appropriately. Further, the average assigned preference score of 1.76 reflects that most of the 34 students were able to obtain one of their first or second choice preferred supervisors, as shown in Figure 2. This suggests that the allocation process aligned well with the student preferences, and resulted in a satisfactory solution for both the staff members and the student population.

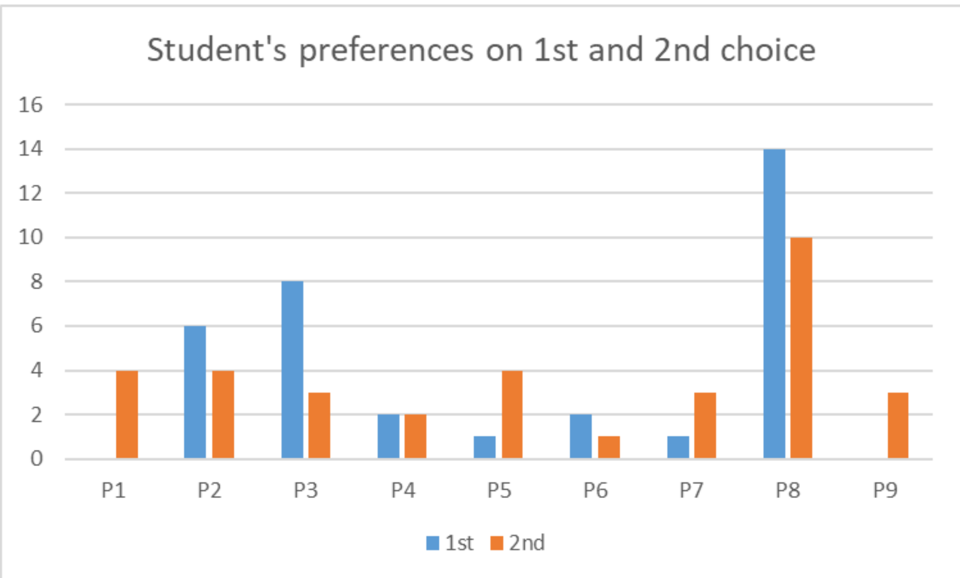

Figure 1: Distribution of student preferences for academic staff members as potential supervisors

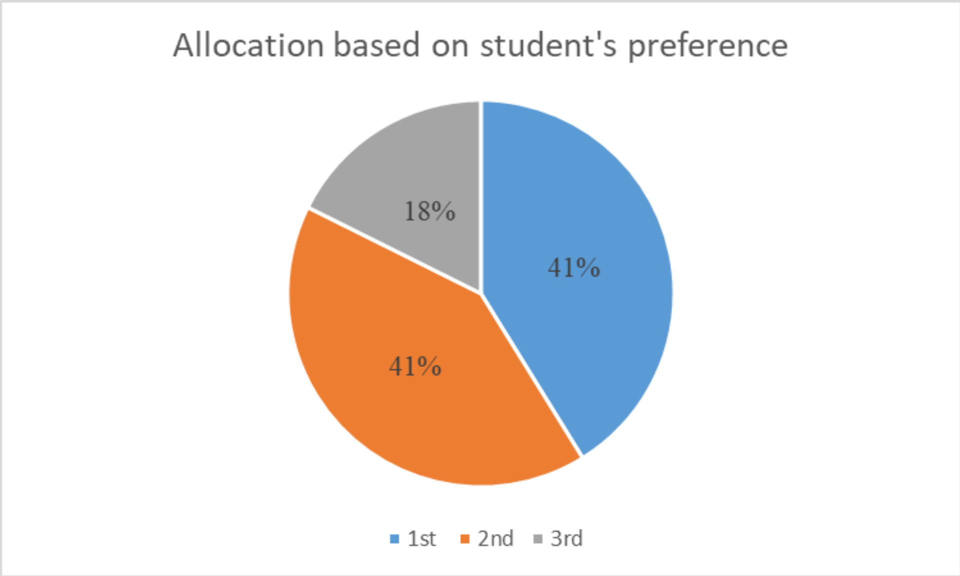

Figure 2: Distribution of supervisors based on student's preferences

The quality of the solution was evaluated based on the number of students assigned to their 1st and 2nd preferences, using Receiver Operating Characteristic (ROC) analysis. The True Positive Rate (TPR) is calculated as the proportion of students who were assigned to one of their top two preferred supervisors, and the False Positive Rate (FPR) as the proportion of students assigned to supervisors that were not one of their top two preferences. Figure 3 depicts a ROC curve that represents the trade-off between the TPR and FPR of the allocation output. The resulting ROC curve is located above, and deviating from the reference line, where the TPR is high and the FPR is low, indicating that the algorithm's overall performance is commendable; it successfully matches a sizable proportion of students to their preferred supervisors, while efficiently minimising the number of students assigned to their less preferred supervisors.

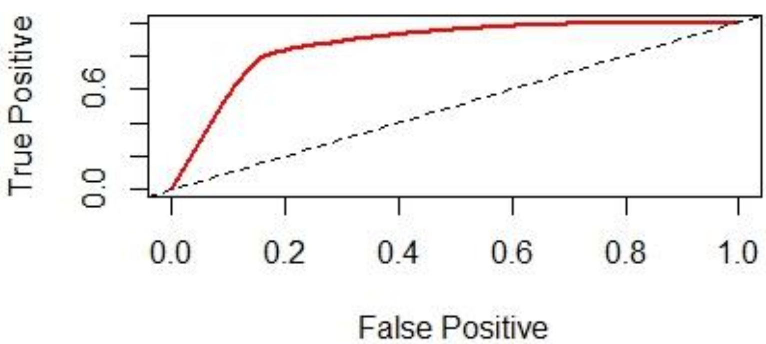

Figure 3: ROC curve showing the trade-off between correctly allocated students and those not correctly allocated based on their desired preferences.

The results obtained from the model, tested on the real data from UBD as a case study, showed an optimal allocation scheme that allocated supervisors to students unbiasedly, allowing for transparency in allocation. The model allowed for flexibility in that the preference of project supervisors can be fine-tuned to cater to various student needs. When compared with the current manual allocation method used for assigning supervisors to students, the linear programming model stands out in various aspects that make it superior. To begin with, the linear programming model proposed in this paper provides an unbiased allocation of supervisors to students, eliminating potential human errors and biases that may occur during manual allocation. This ensures transparency and fairness in the allocation process, preventing any case of favouritism or discrimination. In addition, the model offers flexibility in assigning to students their preferred supervisors by considering workload constraints of supervisors and preferences of students. This results in an allocation that minimises dissatisfaction among both parties and ensures the best fit between each student and supervisor. Lastly, the linear programming model is scalable and efficient in handling larger allocation problems, unlike manual allocation processes that may be prone to errors and inconsistencies when dealing with larger cohort. Overall, the linear programming model optimises the allocation process to yield the best possible outcome, without compromising on quality.

5. Conclusion

This paper proposed a linear programming model for final year student project allocation that aims to minimise the total priority while satisfying imposed constraints. The suggested model showed impressive results in finding an optimal allocation scheme, considering student's preferences, and staff workload in a fair manner. Additionally, the model allowed for flexibility in adjusting the allocation process based on changing needs and circumstances, a feature that proved especially valuable given the unpredictable nature of the university environment. The result from the case study illustrates the potential of the model to address the challenges inherent in traditional manual allocation schemes, and to provide fair, transparent and efficient allocation of supervisors to project students. With the continued use and development of this model, it is expected that the allocation process will become even more streamlined, accurate, and beneficial for both students and supervisors, and the program coordinator. Further research can focus on exploring the scalability of the model, its sensitivity to input data, and potential integration with other relevant systems or processes.

References

1. Anwar A.A., Bahaj A. S., "Student Project Allocation Using Integer Programming", IEEE Transactions On Education, 2003, 46(3), 359-367. DOI: 10.1109/TE.2003.811038
2. Budish E., Cantillon E., "The Multi-unit Assignment Problem: Theory and Evidence from Course Allocation at Harvard", American Economic Review, 2012, 102(5), 2237-2271. DOI: 10.1257/aer.102.5.2237
3. Chiarandini M., Fagerberg R., Gualandi S., "Handling preferences in student-project allocation", Annals of Operations Research, 2019, 275(1), 39-78. DOI: 10.1007/s10479-017-2710-1
4. Chown A.H., Cook C.J., Wilding N.B., "A simulated annealing approach to the student-project allocation problem", American Journal of Physics, 2018, 86(9), 701-708. DOI: 10.1119/1.5045331
5. Harper P.R., de Senna V., Vieira I.T., Shahani A.K., "A genetic algorithm for the project assignment problem", Computers & Operations Research, 2005, 32, 1255-1265. DOI: 10.1016/j.cor.2003.11.003
6. Kuhn H.W., "The Hungarian method for the assignment problem", Naval Research Logistics Quarterly, 1955, 2(1–2), 83-97.
   DOI: 10.1002/nav.3800020109

7. Kwanashie A., Irving R.W., Manlove D.F., Sng C.T.S., "Profile-Based Optimal Matchings in the Student-Project Allocation Problem", Combinatorial Algorithms: 25th International Workshop on Combinatorial Algorithms (IWOCA 2014), 2015, 213-225.
DOI: 10.1007/978-3-319-19315-1_19
8. Maashi M.S., Almanea G., Alqurashi R., Alharbi N., Alharkan R., Alsadhan F., "Solving Student-Project Research Assignment Problems Using a Novel Greedy Linear Heuristic Algorithm: A Case Study at King Saud University, Riyadh Saudi Arabia", Bioscience Biotechnology Research Communications, 2020, 13(2): 1168-1173.
DOI: 10.21786/bbrc/13.3/27
9. Munkres J., "Algorithms for the Assignment and Transportation Problems", Journal of the Society for Industrial and Applied Mathematics, 1957, 5(1), 32-38.
10. Teo C.Y., Ho D. J., "A Systematic Approach to the Implementation of Final Year Project in an Electrical Engineering Undergraduate Course", IEEE Transactions On Education, 1998, 41(1), 25-30.
DOI: 10.1109/13.660783
11. Underhill L.G., "Optimal and suboptimal reserve selection algorithms", Biological Conservation, 1994, 70(1), 85-87.
DOI: 10.1016/0006-3207(94)90302-6
12. Wilson J.A., "A genetic algorithm for the generalized assignment problem", Journal of the Operational Research Society, 1997, 48, 804-809. DOI: 10.1057/palgrave.jors.2600431