Exploratory Analysis in Application of Machine Learning for Optimal Resource Allocation

PROJECT REPORT

SHREYAS VINAYAK SIRDESHMUKH ID : 219004113
SUPERVISOR : A/PROF. BARDO FRAUNHOLZ

Table of Contents

Introduction

This report deals with a specific dimension of the project management i.e. the resource allocation to manage the scope of the project. Benefit the organization by assessing its problems and assisting in fluent functioning by providing it advanced AI/ML tools to achieve its objectives. Organizational growth is a function of reducing errors, time, and costs. This trifecta of growth requires a lot of effort and also scrutiny in the way resources are utilized to gain the most value out of any exercise. Project management is one such discipline that enables an organization to streamline its resources to achieve its objectives in terms of value and assist the organization to grow (Kerzner, 2018). The adaptation of modern project management standards is growing with the realization of the high project complexity and the inflated project failure rates in many leading organizations across the globe. It is observed that 9.9% of every dollar goes unutilized due to poor project management and 2/3rds of the organizations invest in outsourcing and contract-based project management practices (PMI, 2018).

The ever-growing discipline of project management has become smarter and the professionals in the discipline have become even more competent in deriving value through the management principles. With such growth technology has also taken over and brought finesse to the style of managing and planning resources. ERP systems and database management integration with businesses is one standard example of the technological advancements in resource management.

Artificial intelligence has also changed the way businesses function and leverage the data reserves they possess. The (PMI, 2019) survey data shows that organizations that implement a combination of the technology quotient and project management skills are 76% more likely to achieve project objectives than those that don't (61%). Also, organizations are more likely to complete projects without going over budget or exceeding timelines. Coming to the case in focus, the Deakin University faculty allocation is one such task that when automated will go hand in hand with the vision of the organization.

The report at length discusses the impediments in a way of Deakin University while allocating the faculty to Units and subsequent activities. The traditional approaches to the resource allocation are detailed in the report and a prototype is created to address a real-world workload management problem in Deakin University. The feasibility of incorporation on the more advanced technology to solve problems such as resource allocation in an organization such as Deakin university is increasing. With a literal blast in data point the university generates every academic year this is an area that will benefit the organizations and is worth exploring. The effort is directed towards finding such solutions and propose a viable option to the university. Utilizing the power of the prototype equipped with artificial intelligence and leveraging the information in the right manner is the motivation behind the proposed solution to the constraint management and resource allocation challenges faced in an organization

Methodology

<u>Research Design</u>:
The project is exploratory in nature and investigates the potential of taking an alternative approach to resource allocation by creating a prototype algorithm that can be used to allocate tasks to the resources using machine learning. Though the research is exploratory it is primarily quantitative it is quantitative in nature. The research is based on the possibility of innovation in the field of project management and specifically the resource allocation using an automated system that uses AI concepts. The Deakin university's methods to allocate its faculty to the units is understood and an opportunity is sought to progress and automate the manual faculty allocation. Various approaches to

the resource allocation problem are researched and analyzed and a single approach s selected by exclusion and a functional prototype is built to validate the feasibility of application of the AI and ML techniques in the specific problem domain. The results are evaluated using well established metrics and a reasoning to reach a conclusion to the primary questions of the research.

Data Collection:
The data is readily obtained from the data reserves of Deakin University. The raw data represents the faculty and unit and subsequent unit activity allocation for the academic year of 2019-20. The available data is across multiple domains of the university such as Management, Health Sciences, Business and Law etc. The data scope is kept at a single domain of management as to manage the complexity of cross domain faculty restrictions and other University policies. Then the data is processed to be divided into entities and discussed in depth down to the volume and variables of the data.

Data Analysis:
With the help of the Deakin University website and the support of the faculty the data is analyzed to understand its nuances. The data is tailored as per the requirements of the algorithm/selected solution approach. The genetic algorithm requires the data to divided into its basic entity form which helps the algorithm recognize the faculty and units and its activities separately. The entities are separated using excel and manual filtering. The faculty and unit data outliers are removed, and new necessary fields are created. The data is then made static and used to help the prototype algorithm learn the constraints and produce an optimal allocation path.

Prototype Design:
The literature review is conducted for the existing approaches in the resource allocation to realize the potential to innovate and approach he problem from a different perspective. The solution approaches are then discussed, and one suitable approach i.e. Genetic Evolutionary Algorithm is chosen as to create a prototype over. The prototype is created using python programming language and few data manipulation libraries with an object-oriented design approach. Prototype algorithm phases are presented in a sequential manner and the machine learning factors explained, as a result of which we reach the desire optimal faculty unit allocation. The application is created using Spyder IDE platform and uses excel data files as an input. Algorithm performance evaluation metrics are developed with respect to the standard practices. The results are analyzed to justify the selection and use of the solution approach in the University's problem setup.

Business Problem

Organization Background and Challenges in Faculty-Task Allocation

Deakin University is one of the leading Australian universities and accommodates over 60000 students in an academic year. The University is a place of competence and excellence through all disciplines. To manage such a huge number of students the university needs the best of the staff with best of skills. The faculty shapes the status of the university and quality of education. The university needs a competent way to overcome the challenges it faces while allocating the subjects to the teaching staff. The allocation must happen with critical detail and should also consider the quality and status of the university. The university allocates the teaching staff with the subjects manually as to keep the value and the quality at its best. The university does so by understanding the resources and their capabilities deliver the subject the students. The teaching hours are managed and taken out of the contracted teaching hours to attain the maximum value at the best possible cost to the organization. The staff and subject task allocation is of high importance to keep the university with high value and low costs. The allocation needs to become faster and competent enough to give the University the ability to repeat the process any number of times and get the best possible and optimal allocation options. In the setup there are faculty members that will be allocated the subjects and the subsequent tasks for that subject for the academic year. In doing so there are a few constraints to be managed and bring the maximum value to the organization out of the allocation. Driving the motivation and perception of the Deakin University's digital brand automation in resource allocation will benefit the university in overcoming its challenges.

Business Constraints for Faculty Work Allocation

A constraint emerges with a limitation, restriction or an objective of an organization that needs to be taken into consideration in an optimization problem. The maze of the constraints is traced to an optimality by finding a way to manage these constraints by trade-offs and omissions. In context of the Deakin University constraints are the organizational rules made for the faculty members that take into consideration the rights of the faculty members and the performance expectations as well. The optimal solution is a unit-faculty allocation that brings the most value to the organization. This creates an objective of maximum value and hence the capabilities of a faculty member to teach a unit are to be assessed and managed.

I] Capability Constraint: This constraint contributes towards the quality and value of the unit-faculty allocation.

1. Capability of a resource which in our case is the domain expertise which is formulated as a constraint while allocating a subject to the faculty members. The domain expertise is loosely defined by the department that the faculty works for example, Prof. Michael ABC works in HR management is formulated as a constraint while subject allocation. In the scope of the business case selected we have management department which has three sub-domains(HR, General Mgt., Art).

II] Organizational Constraints: Every organization has policies; these policies are to be observed while allocating tasks to the resources.

1. The contract teaching hours for every staff member vary with respect to their contract research percentage and service percentage in an academic year. The unit-faculty allocation happens after taking the available teaching hours and the mandatory research and services hours in an academic year.
2. Workload distribution for the faculty is taken into the set of rules while allocation. The academic year is considered as one project and it has three teaching periods: T1,T2 and T3. In any teaching period no staff member should be allocated to more than 2 units. This constraint manages the exhaustion factor of the staff members.
3. Every Unit should have a dedicated unit head before the optimization of the other unit activities happens.

Faculty Task Allocation Data

The data is obtained from the Deakin University faculty workload allocation from year 2019. The workload allocation data is broken down to the two entities faculty and the Units. Each entity is separately discussed using snippets below. For simplicity only one Domain(Management) is selected which has three sub-domains that are considered in capability constraints.

1. Faculty – The faculty is the resource which is to be allocated to the units and its subsequent activities. The faculty has its research work and organizational services percentages fixed with contract. The contract maximum teaching hours are also fixed. The total number of hours a staff commits to in an academic year is 1690. Faculty data also has the details regarding their domain and campus location.

Domain	Sub-Domair	Staff ID	Staff	Campus	Research	Service	Teaching Hours
Management	Mgt	302753	Achinto Roy	Burwood	0.4	0.11	828.1
Management	Mgt	400530	Alexander Newman	Burwood	0.6	0.5	338

Figure I Snippet of the staff data

2. Units – The Units are the offerings made available by the university in all the three teaching periods of the academic year. A unit has a sub-domain which will be related to the capability of the staff to teach that unit. A single unit might be offered in one or more teaching periods.

Domain	Sub-Domair	Unit Code	Offering
Management	Art	MMM707	T1
Management	HR	MMH250	T1

Figure II Snippet of the Unit offering data

3. Activities – Each unit has the activities as listed below and each activity expects certain number of hours from the staff in a single teaching period for a single unit. Few activities have fixed number of hours and few have a range that can be optimized for best value.

Tasks	Min Hours	Max Hours
Unit Chair	25	25
Review	4	4
Class	99	99
Seminar	66	66
Cloud Seminar	22	22
Unit Development	1	50
Online Resource Managem	40	40
Consultation	1	20
Marking 1st	1	4

Figure III Snippet of the Unit Activities data

The data includes 42 faculty resources, 76 Units offered across 3 teaching periods during one academic year. The faculty as well as the units share One domain i.e. Management domain, and its three sub domains HR , Mgt (General), Art. Each unit has 9 designated activities with hour limits and the value they bring to the organization. The faculty members are to be allocated to the Units and its activities in an optimized way using a machine learning algorithm which learns the constraints and does the optimization of the allocations.

Resource Allocation Solution Approaches

Solution Type	Approach	Limitation
Mathematical	Integer Programming	These approaches are not scalable when multiple variables are added to the problem and the volume of the datapoints increases.
	Mixed Integer Programming	
	Branch and Bound	
	Dynamic Programming (Matching)	
	Multi-Objective Optimization (Trade Off)	
Heuristics	Minimum total slack	Heuristics are used to obtain a suboptimal solution where problem space is near infinity and we require an optimal solution approach
	Least total slack	
	Earliest Late Start	

Machine Learning	DRL- Subproblem decomposition	The Machine learning approaches are adaptable to the problem at hand but need a lot of tweaking and modifications with respect to the complexity of the problem.
	DQN Optimal Critical Path	
	Incremental Reinforcement Learning	
	Genetic Algorithm	

Resource allocation has become a frequent obstacle as it is faced by many industries and sectors. Its wide spectrum ranges from a production line resource management problem in manufacturing facilities to the project planning in IT firms.

Due to the nature of the problem many works have gone into building solutions to the issue with similar set of objectives and slight shifts and innovations with practical application of the solution. As the technology is making its progress new innovations are made in the arena of the resource allocation problem solutions. Traditional approaches are purely mathematical and with the data evolution more sophisticated and advanced approaches in data science are introduced using the power of the machine learning. A brief discussion on the approaches used in practice is detailed below.

Traditionally purely mathematical approaches were devised as a solution to the resource allocation problem. Integer linear programming(ILP) is a very straightforward approach that uses basic constraints and as the name suggests it is limited to integer solutions only(Zoltners and Sinha, 1980). More developed versions of ILP would be Mixed Integer Linear Programming and Branch and Bound algorithm which uses similar logic as ILP. Mathematical approaches also include Dynamic programming which is based on subdividing the problem into smaller sub-problems to arrive at the solution (Zufryden, 1986). Moreover, Multi-Objective Optimization can be considered as an advanced approach that uses tradeoff between the constraints and gives the optimal solution (Devarajan et al., 2012). The computational power of the modern computer limits the application of these approaches to large-scale problems that have higher volume and variety of variables.

Heuristics are known to be helpful in the project management set-up and resource allocation process. Essentially heuristics help in making decisions that are not entirely optimal but suffice the constraints for a timeframe. (Gordon, 1983) discusses the heuristics approaches that are used to solve the resource allocation problems. However, the allocation strategy would suffer from combinatorial explosion considering the high concurrency possibility that multiple tasks may arrive and require resources simultaneously. Moreover, the nature of the constraints also limits the use of these methods.

Relatively new approaches are Deep Reinforcement learning with subproblem decomposition(Li et al., 2020) , Deep Q- Networks for finding an optimal critical allocation path with multi-agent problem also show promising applications in cloud based networking and allocation(Fu et al., 2020). The performance of resource allocation is decided not only by instant cost but also future influence, we can formulate the resource allocation problem into Markov Decision Processes (Basic principle of the Q-Learning and Reinforcement learning) since the problem satisfies the MDP property that the decisions always depend on the current state rather than the historical states. The MDP principles can be applied to the problem space.

The scalability and speed to arrive at the solution are given importance with more modern and sophisticated approaches that have been developed using the sheer power of the data volume and high-speed processing.

In this report, we propose the Genetic evolutionary algorithm as the solution which is a part of the Artificial Intelligence umbrella. The genetic algorithms are used in the resource allocation using the evolutionary approach which resembles the human gene evolution through the time as discussed in (Alcaraz and Maroto, 2001). The algorithm uses a fitness function which ensures that the learned strategy compliant with the given operation rules or constraints of the business problem. The Genetic algorithm approach is chosen against the MDP principles because the trained MDP processes will not be easily adaptable to multiple layers of complexities that might arise due to the layers of additional details in the problem space. The Genetic evolutionary approach is deemed fit for the problem at hand and the solution design based on the genetic resource allocation is furnished below.

Solution Design

The resource allocation solution space is huge as the constraints and the involved elements are usually dynamic in nature. There can be multiple optimal solutions of one allocation problem. We have selected the genetic algorithm approach to solve Deakin University's Faculty Workload allocation problem using the principle of parsimony in comparison with the Deep Reinforcement Learning and Deep Q-Networks approaches. Also, the algorithm has advantages over the traditional mathematical approaches that guarantee performance in terms of scalability and speed.

Genetic Algorithm and Faculty Task Allocation

The genetic algorithm uses the concept of evolution that is observed in the genes of the biodiversity. A comprehensive version of the algorithm is depicted in the figure 1. The process of evolution of the genes is broken down into 5 phases which are mentioned and discussed in detail below. In this section the 5 phases are cross-referred to the corresponding problem statement and how the optimal allocation path is obtained (Alcaraz and Maroto, 2001),(Liu et al., 2005).

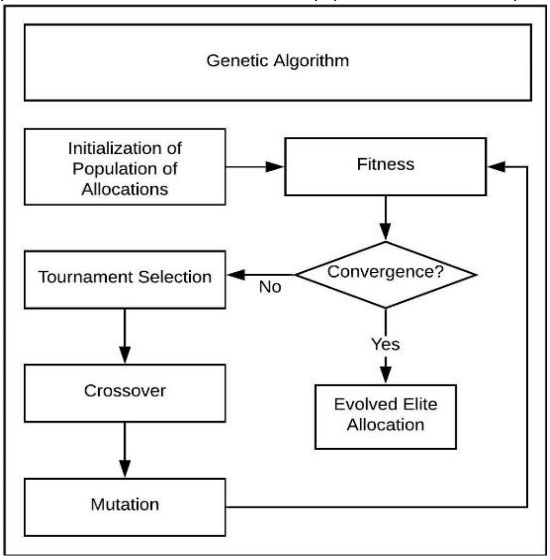

Figure IV Genetic Algorithm

Adjustable Parameters of Algorithm

As an AI/ML algorithm the genetic allocation algorithm has a set of parameters and tuning them to the most perfect of conditions is necessary for the best results. Here we introduce the parameters and discuss how they affect the final solution.

1. <u>Population Size</u> (Integer): Number of allocations created initially before the evolution of the allocations is initiated also called as Generation 0 . The population size determines the time taken for realization of the fittest allocation. The population size should be optimized as a large population size will take longer time to get to the stable state and a smaller population size will lack the variance in the initial population which is a key ingredient of a genetic evolution.
2. <u>Allocations Tournament Size</u> (Integer): The tournament selection process in Phase II depends on this static parameter. A tournament is number of allocations selected before the crossover happens. This parameter modifies the selection of the allocations for crossover populations. The higher the number size of the tournament the less the chances of selection of a new allocation and vice versa. Hence the tournament size is also an optimally chosen parameter.

3. Number of Elite Allocations (Integer): The number of elite populations decides how many top fittest allocations out of a population are to be selected in the crossover population. This population can be initial Gen 0 population or a population that is evolved further i.e. Gen 1 and further.
4. Mutation Rate (Fraction): This is a fractional parameter that decides what amount of mutation is to be brought into the generations after Gen 0. It essentially brings in single faculty subject allocation that is not from the initial population which introduces a factor of new feature that is from other population. This parameter has the highest significance in the genetic allocation as it helps reach convergence by introducing the concept of evolution.

The Allocation Conflict

An allocation path **Conflict** is a mismatch in terms of the business constraints that are to be followed by the allocations. In the business case problem, there are 3 types of conflicts. One if the domain expertise of the faculty does not match with the subject domain, two if the total teaching hours in an academic year exceed the contract hours, three if the faculty is allocated with more than two units in a teaching period out of three teaching periods in the academic year which is an organizational policy constraint. The algorithm evolves through crossover and mutation to omit these conflicts and bring in the features that make an allocation better suited to move across generations. As the conflicts are managed the prototype inches closer to the optimal solution.

The Fitness Function

The Genetic algorithm is based on the concept of biological evolution of the genes. A gene in the business case translates to a single complete faculty to task allocation path where all the units have been allocated a faculty member. Such multiple allocation paths form a population and we need a metric to measure the eligibility of an allocation path to survive through to the crossover population. The fitness function facilitates us with a numeric fitness of each allocation path in the population using which we can determine how fit is the allocation path to produce an offspring or a child allocation path. The fitness function in the business case is based off the conflicts that arise due to non-alignment with one of the business constraints mentioned in the previous section.

The fitness of an allocation path is inversely proportional to the number of conflicts in the allocation of the faculty to the tasks. An allocation path will have a lower fitness if it has higher number of conflicts and vice versa. The genetic evolution manages these conflicts to produce more stable generations that are fitter and have less conflicts.

Allocation Path Fitness = 1/((1.0* Total Number of Conflicts in Allocation Path+ 1))

Note: The fitness of the allocation path will always range from 0 to 1. The ideal or stable allocation path will have 0 conflicts in allocation and result in a fitness of 1.

Genetic Faculty Task Allocation Phases

I] Population Initialization Phase

In this phase an initial population of multiple genes is created similar to the very first set of human genes that did not have all the traits and features that we possess today. A gene in the context of the faculty and unit allocation is one random allocation path of all the faculty to all the units. The random allocation is done adhering to the capability and contract hour constraints as discussed in the business problem section. Multiple such allocation paths constitute a population of allocations which is initialized before the evolutionary process begins. The number of allocation paths is represented by the population size parameter. The fitness of all these allocations is calculated using a fitness function.

II] Tournament Selection Phase

Elite allocation(s) are selected that have the best fitness out of the initial population into the crossover population. The number of Elite allocation(s) selected depend on the ‘Number of Elite Allocations’ parameter. Depending on the elite allocation parameter firstly the elite allocation(s) go into crossover population. After selection of elite allocation(s), 2 tournaments are selected for a single crossover. The size of tournament depends on the parameter ‘Allocation Tournament Size’ from 2 such tournaments 1 fittest allocation each is selected, and the crossover happens in between those two. The crossover population size is kept constant as the initial population size.

III] Crossover Phase

The crossover phase is where the new generation of multiple allocation paths is created as a crossover population. It is done so using the allocation tournament elite allocations as parents.

The crossover of the allocations means that few faculty allocations from one parent and few from another are mixed to form a child/crossover allocation. This process continues until the population size limit is reached and we have a similar size of crossover population with elite allocation(s) from previous population and few allocations from crossover between multiple fittest allocations selected from the tournament’s selection phase.

IV] Mutation Phase

When the crossover phase is under process one allocation is initialized which is not a part of the initial population and that population features are used to mutate the generation or skip. Mutation is a part of the crossover phase. While the crossover of the parents is happening the mutation rate parameter decides whether an unknown allocation feature to the previous population is to be introduced. The mutation phase is the most granular level of the allocation where it brings in the single allocation features unknown to the population that can work either in benefit or loss of the next population. This phase is the most important as it helps the newer populations evolve over time.

V] Elite Generation

Fitness of each crossover population is calculated which is generated through elite allocation selection from the initial population and further process of tournament selection , crossover process and mutation. The algorithm runs for such multiple generations of crossover population. The convergence here is an allocation path in a generation that has 0 conflicts. And is the first stable allocation that can be used as a solution to the resource allocation problem. The generation which has the allocation with 0 conflicts is considered as the elite generation obtained through evolution.

The Machine Learning Factor

This section is an additional detail to outline the functioning of the genetic evolution in resource allocation and why it works and produces an optimal solution.

The phases of the Genetic evolution bring in various best features of the allocations that work for the organization in attaining its objectives, avoid conflicts and survive multiple generations into the subsequent crossover populations. The bad features or conflicts keep on getting omitted as they have a greater number of conflicts and reach the bottom of the future generations.

The algorithm in other terms learns the business constraints that are keeping an initial allocation from becoming a stable/optimal allocation path that is optimized against the adversities of conflicts. The mutation rate is the rate at which an initial population evolves, it is the key that introduces the new features and is responsible for the betterment of the allocation path in log run. The population size determines the problem space and the time to convergence. The tournament size is also crucial in determining the speed of the algorithm and number of elite allocations determines the best faculty task allocation features in every population.

The algorithm essentially is a function of its parameters which are tuned to perfection which yields an optimal faculty to task allocation path.

Solution Evaluation

In this section we discuss the quality of the genetic algorithm in terms of its speed, probability that the algorithm produces viable results and look at the conflict management capacity of the algorithm at convergence. Moreover, this section will provide a brief about the issues in the solution and further advancements in the prototype. The evaluation of the performance of the algorithm is done using three measures. (Sugihara, 1997)

Performance Metrics

I] Time to Convergence :

The time of convergence depicts the speed of the algorithm to reach the desired outcome. This particular metric gives a picture of how well the algorithm is processing the constraints and evolving as the time is passed. We can observe multiple runs of an algorithm and understand the agility of the algorithm to reach convergence. The time is calculated for each generation after Gen0 is obtained. When the algorithm returns an allocation of the resources and tasks with the best possible fitness of 1 the clock returns the time lapsed and its plotted for each run until **n** such convergence runs are finished.

II] Likelihood of Optimality:

Assuming that at the convergence of a run the algorithm is producing the optimal solution, we have to understand the quality of the genetic algorithm to produce optimal solution frequently. The likelihood of optimality enables us to gauge the quality of the algorithm. A generation cap is introduced (**k generations**) for each of the runs of the algorithm out of **n** runs. The genetic evolution is executed for **k** generations and out of **n** runs. The number of runs that produce optimal solution out of **n** runs are denoted as **m.** The likelihood of the optimality(LOpt% (k)) is calculated as the estimated probability [m/n].

III] Average Fitness :

The fitness of an allocation makes it less vulnerable to face elimination during crossover and mutation phases of evolution. The Average fitness is calculated at kth generation taking all the allocations in that generation for aggregation of fitness. The algorithm is executed for **n** number of runs for **k** generations. At the Kth generation the arithmetic average of the fitness of all the allocations in that kth generation is calculated. All the averages are plotted across the **n** runs to understand the power of the algorithm.

Performance Evaluation for Developed Solution Prototype:

The performance metrics are used to establish the quality of the solution produced by the developed prototype. Here we engage the algorithm across multiple runs and at various generation caps to speculate and analyze the complexity, optimality, and overall performance of the program. The four parameters of the algorithm are set at the mentioned values and their relevance discussed when necessary in assessing the performance of the algorithm.

Parameter	Value
Population Size	10
Number Of Elite Allocations	1
Allocations Tournament Size	3
Mutation Rate	0.1

I] Time to Convergence:

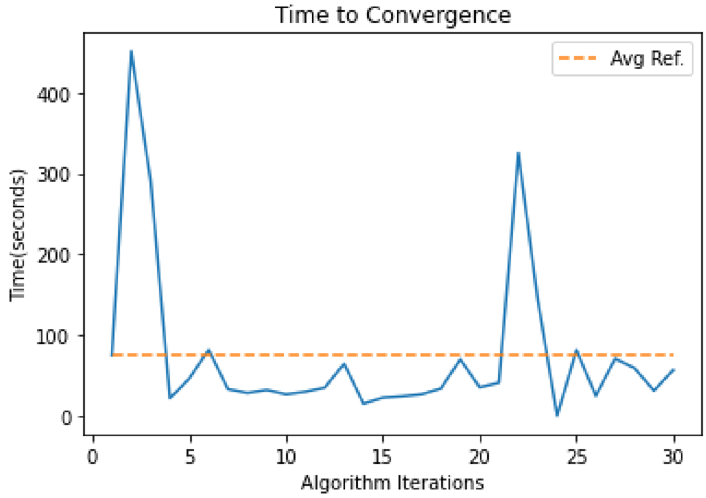

Figure V Time across iterations of Allocation algorithm.

This is a time metric and give a notion as to how quick the program reaches its expected stable state. In the given case the algorithm is evolving over time to obtain an optimal faculty to unit task allocation. It becomes crucial when the genetic algorithm stabilizes or reaches convergence as one of the challenges in resource task allocation is the dearth of time. Figure V shows that the algorithm was executed 30 times without a cap on number of generations and the time in each iteration can be observed as a line plot. Assuming that the algorithm at convergence produced an optimal solution the average time it takes to converge is approximately 80-90 seconds. The two spikes in the chart can be assumed to be outliers and it can be speculated that the randomized allocation at the beginning of each iteration which is the Gen0 can cause such time delays with higher number of conflicts. Overall, the time taken is minimal if compared with the human effort that goes into manual resource allocation. The time taken by the algorithm to converge is heavily dependent on the volume of data and population size. In our case a smaller population size was elected as the problem size is manageable with that. We might need a bigger allocation population size if the number of resources and tasks increase enabling diversity in the initial population.

II] Likelihood of Optimality:

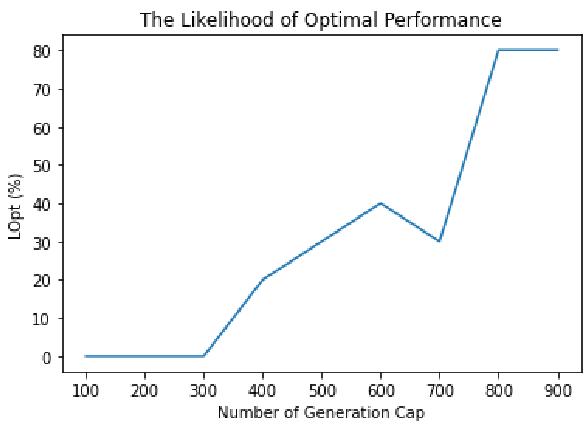

Figure VI LOpt Graph I (k= 100 to 900, n= 10)

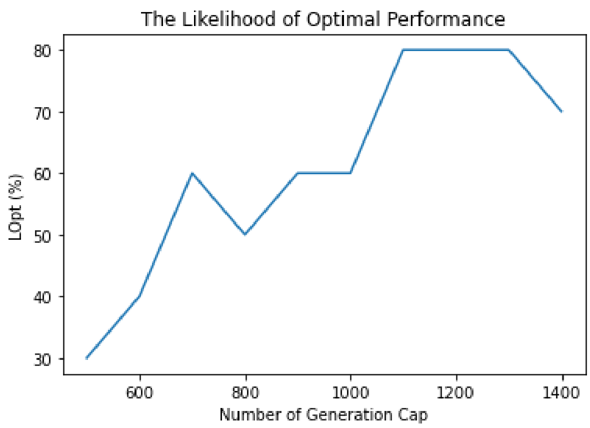

Figure VII LOpt Graph I (k= 600 to 1400, n = 10)

The metric is a measure to analyze the ability of the algorithm to produce the desired optimal faculty to task allocation at the convergence. At set parameters the algorithm takes a certain number of generations to produce the optimal output. The Figure VI and VII show the likelihood of optimality from generation cap of 100 till 1400 in two parts, every likelihood calculated across 10 iterations of algorithm at every increment of generation cap by 100 generations. It is evident from the figure that as the algorithm is given more leeway with the number of generations it produces better results with more optimal allocations. If the algorithm reaching the optimality is observed in Figure VI, at cap of 800 generations the prototype produced a desired optimized faculty to task allocation 8 times out of the 10 iterations. Similarly, it can be observed in the Figure VII that the algorithm reaches 80% LOpt score at 1100 generation cap and stabilizes for the next two increments and drops down from 1300 and further. Speculation can be made for the drop in LOpt after 600 generations in Figure VI and a significant drop at 700 and 1300 in Figure VII. It is possible that at the drop is purely coincidental or it might be causing due to the elected population size. Our initial allocations are based on the concept of natural selection and the fitness of the allocations. It is speculated that at the mentioned generation caps above the allocations are losing good features that cause an allocation to be a fitter allocation(Phases of Genetic Evolution). The loss in such features is causing the allocations to drop in the fitness and move to the bottom of the population and the likelihood of optimality drops as well. It can be considered as a “sampling error” in crossover population generation due to randomized allocation.

III] Average Fitness:

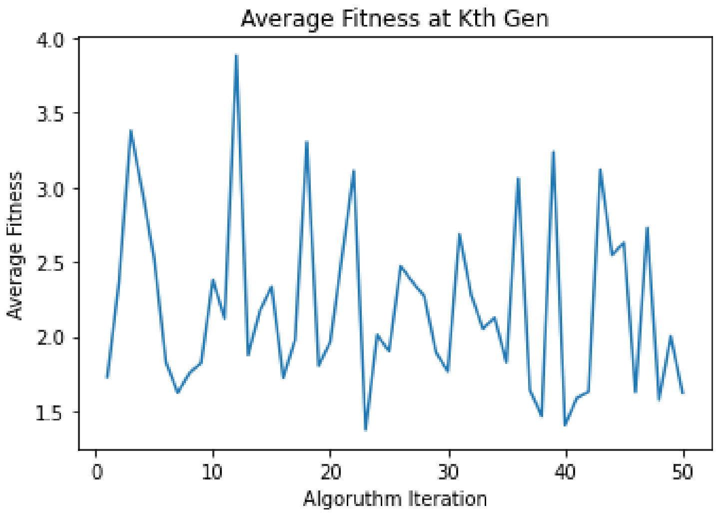

Figure VIII Capped Performance of the Algorithm(k=500)

This metric gives a simple arithmetic measure that helps the user understand the ability of the algorithm to manage the conflicts and evolve to reach the solution. For this the algorithm is capped at **k** number of generation and we observe the average fitness of all the allocation at the completion of **k**th generation. In Figure VI it can be seen that the algorithm has been executed for 50 iterations. The minimum average fitness is appx. 1.4 and maximum at 3.9, the numbers are quite promising as the ideal fitness of a single allocation at convergence would be 1. This represents that the algorithm is performing in a promising manner and the average fitness does not show abnormal behavior across all the 50 runs. The cap of 500 generations is set after the likelihood of optimality at different caps is observed.

Drawbacks and Future Work

The algorithm function and results are tested against the performance metrics and the drawbacks are documented and advancement that can be made are articulated in this section.

1. Overallocation: A few faculty members have more than 70 % of the contract hours for teaching. These faculty members are allocated to more than one unit considering their availability compared to other faculty members. The algorithm allocates a greater number of units that exceed the capacity of a faculty in terms of teaching hours causing overallocation.

2. Evaluation Time: The algorithm evaluation is based on test and run concept where a set of algorithm parameters are evaluated with respect to the performance metrics and . In evaluation of algorithm the performance metrics are not calculated in concurrence with the algorithm iterations but for evaluation we require to keep the parameters the same and test each metric one by one which consumes large amount of time which defeats the purpose of the algorithm. A more dynamic approach can be taken here that allows user to understand the performance of the algorithm at the end of iteration.

The algorithm can be made more robust and adaptable to any problem that resembles an allocation problem by solving the underlying issues and adding multiple features that simplify the process and accelerates the speed at which the algorithm generates the desired results. Such advancements are noted below.

1. Data synthesis : The algorithm expects the input data in a certain format to function. This process can be automated by creating a data pipeline that generates the required features and feeds the data to the algorithm.
2. Auto Cut-Off Generations: Using the Likelihood of Optimality(LOpt) feedback can be given to the algorithm to set generation caps cut off to save time. This will require generation numbers parameter set in the algorithm and a feedback loop that makes the parameter dynamic.
3. Executable Application: User interface or executable application with parameter choices based on the prototype can be built to give the user an experience and ease to access the application.

Conclusion

The potential shown by the AI and ML technology in the recent decade is promising. The applications of the technology have an ever-widening scope and innovation in the same fields keeps justifying the power of AI. The case studied in this project was analyzed and an alternative approach was proposed that was anticipated to bring the solution to the problem. The evolutionary genetic algorithm has given promising results when we look at the Lopt performance metric. The Lopt metric when observed for the Deakin University's and it proves the ability of the algorithm to tackle the challenges of resource allocation problem and produce an optimal solution repeatedly and faster than the manual allocation. The scale of the problem can be magnified by including multiple domains and creating another layer of complexity. The application prototype can be concluded to have answered the primary question of the research i.e. whether AI/ML techniques be used in resource allocation under project management domain.

References

ALCARAZ, J. & MAROTO, C. J. A. O. O. R. 2001. A robust genetic algorithm for resource allocation in project scheduling. 102**,** 83-109.

DEVARAJAN, R., JHA, S. C., PHUYAL, U. & BHARGAVA, V. K. J. I. T. O. W. C. 2012. Energy-aware resource allocation for cooperative cellular network using multi-objective optimization approach. 11**,** 1797-1807.

FU, S., YANG, F. & XIAO, Y. J. I. A. 2020. AI Inspired Intelligent Resource Management in Future Wireless Network. 8**,** 22425-22433.

GORDON, J. H. 1983. Heuristic methods in resource allocation. *International Journal of Project Management,* 1**,** 163-168.

KERZNER, H. 2018. *Project management best practices: Achieving global excellence*, John Wiley & Sons.

LI, K., ZHANG, T. & WANG, R. J. I. T. O. C. 2020. Deep Reinforcement Learning for Multiobjective Optimization.

LIU, Y., ZHAO, S.-L., DU, X.-K. & LI, S.-Q. Optimization of resource allocation in construction using genetic algorithms. 2005 International Conference on Machine Learning and Cybernetics, 2005. IEEE, 3428-3432.

PMI 2018. The Project Manager of the Future: Developing digital-age project management skills to thrive in disruptive times. PMI.org: Project Management Institute.

PMI 2019. Pulse of the Profession 2019 The Future of Work : Leading the Way With PMTQ. PMI.

SUGIHARA, K. Measures for performance evaluation of genetic algorithms. Proc. 3rd. joint Conference on Information Sciences, 1997. Citeseer, 172-175.

ZOLTNERS, A. A. & SINHA, P. J. M. S. 1980. Integer programming models for sales resource allocation. 26**,** 242-260.

ZUFRYDEN, F. S. J. J. O. T. O. R. S. 1986. A dynamic programming approach for product selection and supermarket shelf-space allocation. 37**,** 413-422.

Sener, O. and Koltun, V., 2018. Multi-task learning as multi-objective optimization. arXiv preprint arXiv:1810.04650.

Ahuja, K., 2019. Optimization Methods for Resource Allocation and Machine Learning Applications (Doctoral dissertation, UCLA).

Ergin, H.I., 2002. Efficient resource allocation on the basis of priorities. Econometrica, 70(6), pp.2489-2497.

Werner, H. and Ehn, G., 2018. Reinforcement learning for planning of a simulatedproduction line. Master's Theses in Mathematical Sciences.

Hegazy, T., 1999. Optimization of resource allocation and leveling using genetic algorithms. Journal of construction engineering and management, 125(3), pp.167-175.

Gardner, S., Golovidov, O., Griffin, J., Koch, P., Thompson, W., Wujek, B. and Xu, Y., 2019, October. Constrained Multi-Objective Optimization for Automated Machine Learning. In 2019 IEEE International Conference on Data Science and Advanced Analytics (DSAA) (pp. 364-373). IEEE.