ARTIFICIAL INTELLIGENCE TOOLS FOR ACADEMIC MANAGEMENT: ASSIGNING STUDENTS TO ACADEMIC SUPERVISORS

Victor Sanchez-Anguix[1], Rithin Chalumuri[2], Juan M. Alberola[1,3], Reyhan Aydogan[3]

[1]Universitat Politècnica de València (SPAIN)
[2]Coventry University (UNITED KINGDOM)
[3]Florida Universitària (SPAIN)
[4]Ozyegin University (TURKEY)

Abstract

In the last few years, there has been a broad range of research focusing on how learning should take place both in the classroom and outside the classroom. Even though academic dissertations are a vital step in the academic life of both students, as they get to employ all their knowledge and skills in an original project, there has been limited research on this topic. In this paper we explore the topic of allocating students to supervisors, a time-consuming and complex task faced by many academic departments across the world. Firstly, we discuss the advantages and disadvantages of employing different allocation strategies from the point of view of students and supervisors. Then, we describe an artificial intelligence tool that overcomes many of the limitations of the strategies described in the article, and that solves the problem of allocating students to supervisors. The tool is capable of allocating students to supervisors by considering the preferences of both students and supervisors with regards to research topics, the maximum supervision quota of supervisors, and the workload balance of supervisors.

Keywords: Apps for education, New projects and innovation, Academic management.

1 INTRODUCTION

Every year, students in Higher Education face the challenge of carrying out a long-term project that encompasses a wide breadth of the skills and competences developed during their respective degrees. These projects usually come in the form of undergraduate, master, or PhD. dissertations. While there is a wide range of strategies, tools, and methodologies that have been studied with the purpose of conducting good in-class and module/course experiences [1, 2], there is, comparatively, very little effort on the matter of conducting the experience and guidance of students with respect to their individual dissertations [3]. Yet, we believe that this is an important event in every student's academic life, as it usually entails their degrees' completion and an opportunity to exhibit their accomplishments.

Generally, the responsibility of assigning students to academic supervisors lies on the shoulders of decision makers in academic management (i.e., department heads, coordinators, etc.). Albeit it may initially seem straightforward, decision makers face an intricate problem influenced by a variety of criteria such as the number of students to allocate, the number of supervisors from where to choose, both students' and supervisors' preferences on research topics, work load constraints by academic staff, and even the department's social climate. As a result, the decision on how students should be allocated to supervisors is arduous, time consuming, and, quite often, stressful.

This decision has almost countless possibilities and, making an optimal decision on the matter, may be unfeasible for human decision makers Despite this problem, Artificial Intelligence has proved to be a helpful tool in supporting decision making in a variety of complex domains such as emergency response [4], analytics [5], health [6], sports [7], or even education [8, 9, 10]. Its ability to manage complex decision problems (i.e., involving many variables), provide time effective responses, and learn from experience has put artificial intelligence to the forefront of today's ICT solutions. We believe that the problem of assigning students to supervisors is a complex one, and it would benefit from the support of Artificial Intelligence tools.

ISBN: 978-84-09-17939-8

In this paper, we present an artificial intelligence tool that aims to support members in academic management in the decision of allocating students to supervisors for their academic dissertations. On the one hand, the tool takes into consideration the preferences of both students and supervisors on research topics to ensure that students are advised by supervisors with experience on their desired research topic, and that supervisors feel comfortable advising and guiding students throughout their dissertations. On the other hand, the tool also takes into consideration the different workload levels of individual supervisors (i.e., the maximum number of students that each supervisor may be able to supervise). In order to foster a good work climate, the tool also ensures that all supervisors have a similar workload in the final allocation. We have implemented some pilot experiences, with positive perspectives, to test its feasibility as a decision support tool for members in academic management.

2 STRATEGIES FOR ALLOCATING STUDENTS TO SUPERVISORS

When one faces the task of allocating students for their undergraduate, master, or PhD dissertations (or even summer internship projects), there are multiple alternatives that one may follow. In most institutions, there is a single individual that is responsible for the management of the allocation of students to their respective dissertation projects or supervisors. The general strategy followed by the manager to carry out this decision can be analyzed on different dimensions. In the next lines, we analyze and discuss some of the most relevant dimensions related to the decision of allocating students to dissertations.

- **Distributed vs. Centralized:** If the decision is to be made entirely by supervisors and students, we say that we have a distributed decision, as involved individuals make their own decisions. This approach may have the disadvantage of some students ending up without a dissertation or supervisor, a fatal and unfair consequence for the satisfaction of students with their dissertation project. In addition to this, some supervisors may end up with an excessive workload due to their popularity, and some other supervisors may not be able to supervise any student. Considering that supervising students is a good starting point for starting new research projects, exploring new research areas, or learning on new topics, a distributed decision making process may hinder the research work of some staff members in comparison to those staff members that are the most popular. A centralized decision is one that is entirely carried out by individual in charge of the allocation process, considering the inputs of both students and staff members. A centralized decision may overcome all the previous problems. Nevertheless, the decision faced by this individual is complex, large, and it may involve too many criteria to handle without any external support. In many schools, departments, and faculties the decision is at some point in the spectrum between fully distributed and fully centralized. That is, some individual arrangements between students and supervisors may be allowed, but the academic manager still faces a complex and large decision-making problem.
- **Preference elicitation on items:** This dimension affects centralized decisions, as a purely random allocation may be highly unsatisfactory for both students and supervisors. When making a decision that involves several stakeholders, one should aim to approximately know the preferences of all the stakeholders. This knowledge facilitates making a decision that may satisfy all the parties. The preferences of students and supervisors may be expressed in different terms: individual-based preferences, project-based preferences, and topic-based preferences.
    - Individual-based preferences [11, 12, 13] force students to rank supervisors, and supervisors to rank students. We argue that this preference elicitation method may be biased by the academic merit of specific students and the popularity of specific staff members. In the former case, students with highest academic merits may be more popular among supervisors and end up being preferred by most supervisors. We argue that making decisions purely based on academic merit is against the spirit of modern learning institutions, whose aim should be to promote the learning of all students, providing them with equal opportunities. The latter suffers from the same problems for staff members than those mentioned for distributed decisions. In addition to this, one should also consider that many students and staff members may not be known by their counterparts.

- Project-based preferences [11, 14, 15] consists of staff members proposing a catalogue of projects, and students ranking their preferences over those projects. The expression of students' preferences in terms of projects avoids creating disadvantage for supervisors that are unknown by students, as, supposedly students choose projects based on the project topics. It may also avoid unfair allocations by merit. Despite this, there is some inherent disadvantage for this approach, as usually supervisors need to propose a project catalogue that includes a wide range of choices. Usually, a range of projects that is larger than the number of students. Proposing a set of projects may be a time-consuming process for staff members, with many of the projects ending up unassigned due to the lack of interest from students.
- Topic-based preferences [16] involve students and staff members expressing their preferences in terms of dissertation topics or research (sub)areas. For instance, students and supervisors may provide a list of research topics in which they are interested. Then, the matching is purely based on common research interests of the students and his/her allocated supervisor. This allocation strategy avoids unfair allocation of students based on merit and the issue of (un)popular students/supervisors. On top of that, this method does not require staff members to invest significant time to provide a broad range of projects, as they can negotiate the projects with their allocated students based on the common interests of both.

- **Single/Multi-objective decision:** Another dimension to be considered for the allocation process is whether to consider multiple objectives. Some proposals [14, 15] only consider the objective of satisfying students' preferences, ignoring staff members' preferences. While student satisfaction is important for every academic institution, one should not forget the other side of the coin: staff members. In fact, one could argue that unhappy workers may affect the quality of the teaching and supervision process. One way to overcome this problem is considering several objectives when allocating students to supervisors. In fact, we propose that at least two objectives should be considered, one being the satisfaction of students with the allocation process, and the other being supervisors' satisfaction with the same process. From this point, other considerations may be included such as the distribution of students across different departments.
- **Supervision constraints:** Another dimension consists of whether the allocation considers supervision constraints. In the real world, not every single allocation is possible. Normally, there are realistic constraints that preclude some allocations from being possible. One of these constraints may be realistic workload. Staff members can only supervise a limited number of students, as they have other arrangements and teaching commitments. Another constraint that may be considered is the maximum number of students that may work on the same dissertation topic/project. Some other constraints may include personal incompatibilities or conflicts between specific students and supervisors.

In our perspective, an adequate allocation process should be centralized, multi-objective, consider constraints, and be based on unbiased preferences such as topic-based preferences and project-based preferences.

3 AN ARTIFICIAL INTELLIGENCE TOOL FOR ASSIGNING STUDENTS TO SUPERVISORS

We have developed a student-supervisor allocation tool based on an Artificial Intelligence algorithm that allows to find high quality allocations even for large student cohorts (e.g., 400 students). It is based on a genetic algorithm that finds solutions that are near optimal. The tool is freely available for download[1]. It aims to overcome most of the aforementioned problems. In order to do that, the tool has the following characteristics:

- It is a centralized decision-making tool whereby the user provides through the software all the data needed to solve the problem. Due to this fact, we avoid the case of students not being assigned a supervisor.

[1] The tool is currently available for download at https://github.com/rithinch/pareto-optimal-student-supervisor-allocation

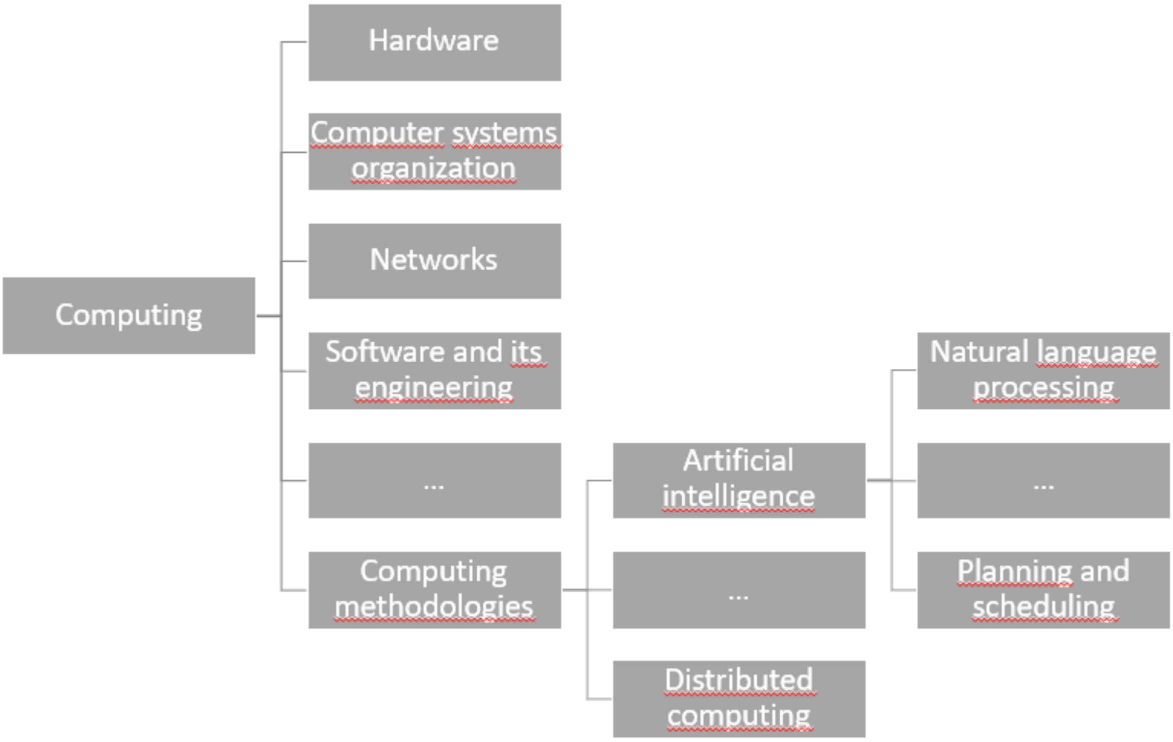

Figure 1. An example of a hierarchy of research topics

- The tool takes into consideration the fact that supervisors may only supervise a limited number of students, and the fact that they may have different teaching commitments. Thus, we provide the ability for the user to individually specify, for each staff member, the maximum number of students to be supervised. The allocation provided by the algorithm never produces allocations that exceed the maximum supervision quota of each supervisor.
- The students' and supervisors' preferences are provided by means of a ranked list of five keywords, from the most preferred topic to the least preferred topic. The keywords are part of a hierarchy of topics that represents the body of knowledge in a discipline. An example of this kind of organization can be found in Figure 1. We chose this representation as it is easy to obtain or construct one for any discipline, and it provides extra and valuable information to the artificial intelligence algorithm. We may use the information encoded in the hierarchy to improve allocations. For instance, if a student has expressed its preference to work on *Natural Language Processing* dissertation and there is no supervisor with such a keyword, it may be possible to find supervisors that have provided the more general keywords of *Artificial Intelligence* and *Computing Methodologies*. These supervisors, even though they may not be experts in the topic they may have knowledgeable experience in the broader field to supervise dissertations in the student's choice of topic. As of now, the tool includes the ACM classification taxonomy for computing topics and the PhySH classification scheme for physics. However, users can provide their own taxonomies in a file, making the tool applicable to almost any discipline.
- The software provides allocations by taking into consideration both the students' and supervisors' preferences. The valuation of the quality of the matching of a student to a specific supervisor is calculated taking into consideration the keywords provided by each of them, the position of those keywords in the hierarchy, and the position of the keywords in the ranked list of preferences. The more similar the ranked lists, the higher quality for the matching. One of the objectives aims to maximize the average quality of the allocation from the students' perspective. The other objective aims to maximize the average quality of the allocation from the point of view of supervisors.
- The software also strives to find an allocation in which supervisors have a similar level of supervision workload. This avoids some supervisors being overutilized while other supervisors being underutilized.
- The tool provides a range of prospective solutions to the decision maker, with different priorities for the students' and supervisors' objectives. This allows decision makers to trade-off between

different levels of quality for each objective, as well as choosing the solutions that best matches his/her preferences.

- The data is provided to the tool in a friendly Excel spreadsheet format, so that users can prepare their data in comfortable and familiar way. The output of the tool is also an Excel spreadsheet that allows easy interpretation of the results. Examples of these files, with synthetic data, can be observed in Figure 2.

Student ID	Student Name	Keyword 1	Keyword 2	Keyword 3	Keyword 4	Keyword 5
5640317	Deanna Pittard	Network algorithms	Network protocols	Models of computation	Hardware validation	Computer systems org
8285751	Camie Junious	Information theory	Dependable and fault-tolerant systems and networks	Concurrent computing methodologies	Computer graphics	Parallel computing me
6932519	Twila Starkes	Computing / technology policy	General and reference	Cryptography	Network components	Software notations an
5230783	Wilhelmina Dismukes	Security in hardware	Operations research	Network security	Mathematical analysis	Concurrent computing
1139444	Ellamae Allgood	Hardware	Electronic design automation	Document management and text processing	Professional topics	Randomness, geomet
8478490	Nelly Buell	Logic	Network performance evaluation	Arts and humanities	Hardware test	Computational compl
8201201	Shante Brass	Continuous mathematics	Security services	Information retrieval	Network protocols	Emerging technologie
2625849	Kate Franceschi	Printed circuit boards	Network architectures	Randomness, geometry and discrete structures	Information storage systems	Software creation and
3881668	Tiffani Ebert	Software and application security	Continuous mathematics	Intrusion/anomaly detection and malware mitigation	Emerging technologies	Computational compl
3730005	Donte Luebbers	Design and analysis of algorithms	Machine learning	Continuous mathematics	Document management and text processing	Architectures
5708348	Derick Stuber	Arts and humanities	Mathematics of computing	Architectures	Electronic design automation	Software and applicat
3556004	Venita Marksberry	Enterprise computing	Randomness, geometry and discrete structures	Network properties	Network security	Logic
4981375	Loan Chevere	Hardware	Robustness	Models of computation	Arts and humanities	Semantics and reason
6453992	Teresa Worthen	Software and its engineering	Human-centered computing	Education	Systems security	Document manageme
4135264	Tien Pomerleau	Information retrieval	Theory and algorithms for application domains	Visualization	Human computer interaction (HCI)	Emerging technologie
9282954	Cherilyn Frigo	Collaborative and social computing	Information theory	Integrated circuits	Mathematical analysis	Document manageme
4724075	Matthew Beery	Theory of computation	Information retrieval	Integrated circuits	Robustness	Security services
7701687	Rosa Manner	Models of computation	Professional topics	Network types	Network protocols	Database and storage
2348495	Glady Hoos	Human computer interaction (HCI)	Computer systems organization	Data management systems	Security and privacy	Systems security
4523338	Ernestine Paulsen	Theory and algorithms for application domain:	Applied computing	Education	Probability and statistics	Software and its engir
4750845	Boyd Harm	Information theory	Hardware test	Computer forensics	Social and professional topics	Logic
9862378	Ernie Sloat	Information theory	Formal languages and automata theory	Security services	ACM Computing Classification System	Computing methodol
2913396	Marion Lacey	Randomness, geometry and discrete structure	Ubiquitous and mobile computing	Network algorithms	Machine learning	Hardware validation
3452245	Ernie Sloat	Software and application security	Network architectures	Life and medical sciences	Concurrent computing methodologies	Network services
5567213	Tiffani Ebert	Visualization	Security in hardware	Network protocols	Accessibility	Interaction design

Supervisor ID	Supervisor Name	Quota	Keyword 1	Keyword 2	Keyword 3	Keyword 4
aab9775	Boris Lovell	4	Computing methodologies	Computational complexity and cryptography	Human and societal aspects of security and privacy	Modeling and simulation
aab5394	Boyd Harm	8	Distributed computing methodologies	Computer graphics	Law, social and behavioral sciences	Accessibility
aab6941	Lila Speno	7	Design and analysis of algorithms	Physical sciences and engineering	Cross-computing tools and techniques	Computing methodologies
aab7707	Camie Junious	7	Interaction design	Software and application security	Intrusion/anomaly detection and malware mitigatior	Law, social and behavioral sciences
aab4056	Nelly Buell	9	Communication hardware, interfaces and storage	Software notations and tools	Concurrent computing methodologies	Education
aab3234	Ngoc Licari	6	Software notations and tools	Machine learning	Document types	Distributed computing methodologie
aab3532	Venita Marksberry	9	Security in hardware	Software creation and management	Human computer interaction (HCI)	Printed circuit boards
aab9476	Andreas Breitenbach	10	Computer systems organization	Security services	Computers in other domains	Modeling and simulation
aab7527	Latisha Mongillo	10	ACM Computing Classification System	Network algorithms	Software notations and tools	Software organization and properties
aab4593	Lajuana Bilodeau	7	ACM Computing Classification System	Security services	Theory of computation	Electronic design automation
aab3189	Boris Lovell	9	Computing methodologies	Security in hardware	Physical sciences and engineering	Probability and statistics
aab4893	Deanna Pittard	10	Human-centered computing	Law, social and behavioral sciences	Arts and humanities	Software and application security
aab3424	Tien Pomerleau	5	Machine learning	Cryptography	Software organization and properties	Information retrieval
aab4354	Akiko Kellen	4	Computing methodologies	Network algorithms	Life and medical sciences	Information retrieval
aab6922	Teresa Worthen	7	Communication hardware, interfaces and storage	Applied computing	Theory and algorithms for application domains	Randomness, geometry and discrete
aab6377	Shana Burghardt	5	Information theory	Law, social and behavioral sciences	Parallel computing methodologies	Power and energy
aab9331	Akiko Kellen	9	Cryptography	Logic	Accessibility	Concurrent computing methodologie
aab6769	Noel Langer	5	Computer forensics	Communication hardware, interfaces and storage	Printed circuit boards	Very large scale integration design
aab5703	Nichole Pedretti	10	Formal languages and automata theory	Architectures	Information theory	Computational complexity and crypto
aab6390	Anastasia Turk	4	Security services	Human and societal aspects of security and privacy	Software creation and management	Integrated circuits
aab2726	Moises Brumer	4	Network architectures	Symbolic and algebraic manipulation	Visualization	Parallel computing methodologies
aab5149	Ngoc Licari	6	Robustness	ACM Computing Classification System	Software notations and tools	Randomness, geometry and discrete
aab9850	Nery Stribling	9	Dependable and fault-tolerant systems and networks	Software notations and tools	Network algorithms	Ubiquitous and mobile computing
aab5516	Loan Chevere	10	Information systems applications	Printed circuit boards	Network performance evaluation	Life and medical sciences
aab8341	Kera Mohler	4	Information theory	Hardware test	Information storage systems	ACM Computing Classification System

Figure 2. Examples of the spreadsheets provided to the tool for students (top) and supervisors (bottom)

The tool provides a simple graphical interface that aims to be intuitive and easy to use by everyone. In the main window, one can select an input spreadsheet file for the students' data, an input spreadsheet file for supervisors' data, and the filename where the results should be output. The artificial intelligence algorithm has been configured with optimal parameters, so that the user can just focus on creating his/her allocations. Figure 3 shows the main window of the application.

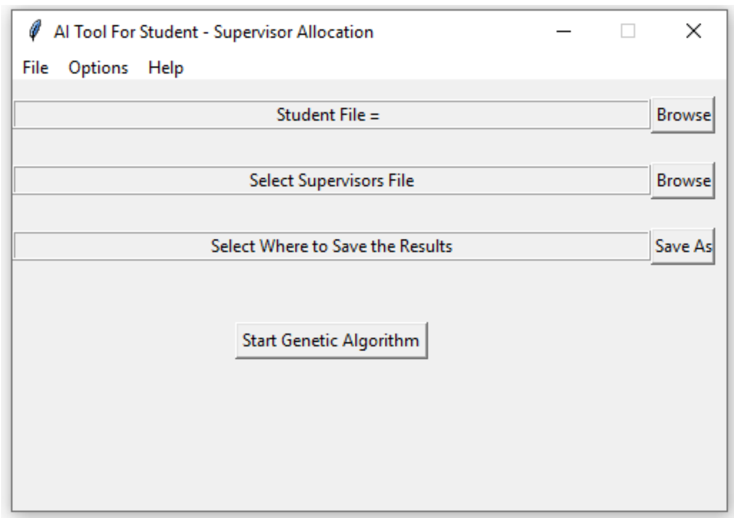

Figure 3. The main window of the tool

The tool has been employed with success at Coventry University (United Kingdom) to allocate a cohort of more than 300 students to approximately 50 supervisors. The feedback gathered from the academic in charge of the allocation was positive, highlighting its ability to significantly reduce the time employed to allocate students to supervisors. In addition to this, the tool is currently being tested at other UK and Spanish institutions.

The Artificial Intelligence algorithm has proved to provide near optimal results in minutes even for large and complex decision problems [16]. In fact, the tool can obtain solutions that have 88.63% of the best quality achievable for the best solution for students, and 93.83% of the best quality achievable for the best solution for supervisors. These results have been obtained on an average of 247 seconds (4 minutes).

4 CONCLUSIONS AND FUTURE WORK

In this paper, we have presented a new tool, powered by Artificial Intelligence, that supports academics in the process of allocating students to supervisors for their respective undergraduate, master, and PhD dissertations. This tool falls into the category of a centralized decision-making tool that is guided by the preferences of students and supervisors on research topics rather than specific projects or students/supervisors. In addition, the tool takes into consideration multiple objectives in the form of the students' preferences with regards to research topics, and the supervisors' preference with regards to research topics. The tool also includes several tools to produce realistic allocations such as the maximum supervision quota of each supervisor, or the need to produce a balanced allocation that avoids some supervisors being overutilized and other supervisors being underutilized.

As mentioned, there is limited research with regards to student supervision compared to other teaching activities. This means that there is still plenty of room to improve and produce tools that help in the process of allocating and supervising students. For instance, in this article we have employed research topics extracted from a hierarchy of topics to represent the preferences of students and supervisors. This preference elicitation process could be enriched with information from the latest articles published by the supervisors, descriptions from the subjects taught by supervisors, or abstracts from previously supervised dissertations. In addition to this, another potential area of exploration is studying the impact of psychological profiles in the evolving relationship between students and supervisors.

REFERENCES

[1] R. R. Schmeck, Learning strategies and learning styles, Springer Science, 2013.

[2] J. Biggs and C. Tang, Teaching for Quality Learning at University, McGraw Hill, 2011.

[3] M. Woolhouse, "Supervising dissertation projects: Expectations of supervisors and students," *Innovations in Education and Teaching Internationa,* vol. 39, no. 2, pp. 137-144, 2002.

[4] S. Potter, "Critical reasoning: AI for emergency response," *Applied Intelligence,* vol. 37, no. 3, pp. 337-256, 2012.

[5] F. Y. Wang, "A big-data perspective on AI: Newton, Mertin, and analytics intelligence," *IEEE Intelligent Systems,* vol. 27, no. 5, pp. 2-4, 2012.

[6] J. Huang, N. R. Jennings and J. Fox, "Agent-based approach to health care management," *Applied Artificial Intelligence,* vol. 9, no. 4, pp. 401-420, 1995.

[7] J. M. Alberola and A. Garcia-Fornes, "Using a case-based reasoning approach for trading in sports betting markets," *Applied Intelligence,* vol. 38, no. 3, pp. 465-477, 2012.

[8] J. M. Alberola, E. Del Val, V. Sanchez-Anguix, A. Palomares and M. D. Teruel, "An artificial intelligence tool for heterogeneous team formation in the classroom," *Knowledge-Based Systems,* vol. 101, pp. 1-14, 2016.

[9] E. Del Val, J. M. Alberola, V. Sanchez-Anguix, A. Palomares and M. D. Teruel, "A team formation tool for educational environments," in *Trends in Practical Applications of Heterogeneous Multi-Agent Systems. The PAAMS Collection*, Springer, 2014, pp. 173-181.

[10] R. Luckin, W. Holmes, M. Griffiths and L. B. Forcier, Intelligence Unleashed: An argument for AI in Education, Pearson, 2016.

[11] D. J. Abraham, R. W. Irving and D. F. Manlove, "Two algorithms for the student-project allocation problem," *Journal of Discrete Algorithms,* vol. 5, no. 1, pp. 73-90, 2007.

[12] D. Srinivasan and L. Rachmawati, "Efficient fuzzy evolutionary algorithm-based approach for solving the student project allocation problem," *IEEE Transactions on Education,* vol. 51, no. 4, pp. 439-447, 2008.

[13] L. Pan, S. Chu, G. Han and J. Z. Huang, "Multi-criteria student project allocation: A case study of goal programming formulation with dss implementation," in *8th International Symposium on Operations Research and Its Applications*, Zhangjiajie, 2009.

[14] A. A. Anwar and A. Bahaj, "Student project allocaton using integer programming," *IEEE Transactions on Education,* vol. 46, no. 3, pp. 359-367, 2003.

[15] P. R. Harper, V. de Senna, I. T. Vieira and A. K. Shahani, "A genetic algorithm for the project assignment problem," *Computers & Operations Research,* vol. 32, no. 5, pp. 1255-1265, 2005.

[16] V. Sanchez-Anguix, R. Chalumuri, R. Aydoğan and V. Julian, "A near Pareto optimal approach to student–supervisor allocation with two sided preferences and workload balance," *Applied Soft Computing,* vol. 76, pp. 1-15, 2019.