Iterateted Local Search with Random Restarts for the Mentorship and Teamwork Problem (Google Hash Code 2022)

Kadri Sylejmani1, Labeat Arbneshi2, Elvir Misini3 and Uran Lajçi4

1 Faculty of Computer and Software Engineering, Prishtina University, Agim Ramadani, 1000 Prishtinë, Kosovë

kadri.sylejmani@uni-pr.edu

2 Faculty of Computer and Software Engineering, Prishtina University, Agim Ramadani, 1000 Prishtinë, Kosovë

labeat.arbneshi@uni-pr.edu

3 Faculty of Computer and Software Engineering, Prishtina University, Agim Ramadani, 1000 Prishtinë, Kosovë

elvir.misini@student.uni-pr.edu

4 Faculty of Computer and Software Engineering, Prishtina University, Agim Ramadani, 1000 Prishtinë, Kosovë

uran.lajci@student.uni-pr.edu

Keywords: Mentorship and Teamwork *·* Iterated Local Search *·* Google Hash Code 2022

# Introduction and problem description

In the paper "Iterated Local Search with Random Restarts for the Mentorship and Teamwork Problem (Google Hash Code 2022)," the authors focus on a specific problem proposed in the Google Hash Code 2022 competition. The Mentorship and Teamwork Problem (MTP) involves creating optimal teams and assigning mentors, considering various constraints and objectives to maximize the overall performance and satisfaction of participants.This variant of MTP consists of assigning team members and mentors to form teams, while respecting hard constraints, such as the maximum and minimum team size and matching the skill sets of team members with the mentor's expertise. Furthermore, soft constraints, such as minimizing the skill gap within a team and promoting diversity, need to be considered to enhance the overall team performance.To tackle this challenging combinatorial optimization problem, the authors propose an Iterated Local Search (ILS) algorithm with random restarts. The algorithm starts with an initial team formation and iteratively refines the solution by exploring the search space, aiming to optimize the given objective function. The ILS algorithm incorporates perturbations and acceptance criteria to escape local optima and enhance search performance.The random restarts are introduced to further improve the algorithm's exploration capabilities. When a certain stopping criterion is met, the algorithm restarts from a new randomly generated solution. This helps the algorithm to explore different regions of the search space and avoid getting trapped in suboptimal solutions.For the sake of brevity, the full problem description and experimental results are not reported here; instead, we refer to the original paper for a detailed discussion on the proposed approach and its performance on the Mentorship and Teamwork Problem.

# Solution method

Following the spirit of the work by Bellio et al. [2] for the uncapacitated ETT, we developed a multi-neighbourhood Simulated Annealing (SA) algorithm. The choice for SA is motivated by the fact that SA has already proven to be very e ective for this [1, 9] and a number of other timetabling problems (see, e.g., [3, 6]). Our search space is composed by an array of pairs that assigns to each exam a period and a room, and also includes solutions that may violate hard

Table 1. Considered neighbourhoods

Move(e,p,r) Move exam e to period p and room r.

Swap(e1,e2) Swap the period and room assigned to exams e1 and

e2.

Kick(e1,e2,p,r) Move exam e1 to the period and room assigned to e2.

Move exam e2 to period p and room r.

constraints such as con icts or room capacities. These violations are included in the cost function, along with the soft constraints, but with a suitably larger weight.

The portfolio of neighbourhoods that we already implemented is given in Table 1. These neighbourhoods were originally proposed for the uncapacitated version to the ITC-2007 problem by Bellio et al. [2], and were adapted to deal with the assignment of rooms which is not considered in the uncapacitated prob- lem.

# Preliminary experimental results

Idea to improve the c and f solution instances:

Projects: We should sort the projects following this: number of contributors/skills required in project / summary of the required skills.

Contributors: Sort contributors from the lower or the higher skills or find a heuristic for the sort.

Mentor: We should not add a contributor in a project with a mentor it he has another skill in that project.

We should do an implementation of the initial solution that checks for mentors after there are added all the possible contributors that do not require mentors.

Table 2. Preliminary results. Best available solutions are from https://codingcompetitions.withgoogle.com/hashcode/round/00000000008caae7.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Team | a | b | c | d | e | f | Total |
| **The Best Results from all Teams** | 33 | 1,005,020 | 288,508 | 674,945 | 1,650,488 | 1,194,515 | 4,813,476 |
| Make love, not war | 33 | 969,087 | 229,517 | 674,945 | 1,640,454 | 706,200 | 4,220,236 |
| Rethinkers | 33 | 932,759 | 274,679 | 384,328 | 1,599,952 | 904,867 | 4,096,618 |
| Past Glory | 33 | 909,802 | 223,267 | 562,814 | 1,640,172 | 756,170 | 4,092,258 |
| Proof by Submission | 33 | 800,991 | 256,630 | 173,626 | 1,587,033 | 1,194,515 | 4,015,828 |
| code | 33 | 900,799 | 259,165 | 399,105 | 1,640,492 | 765,794 | 3,965,388 |
| **Our Solution** | **33** | **900,270**  **70 iterime** | **20,703**  **5 iterime** | **225,134**  **50 iterime** | **1,308,127**  **10 iterime** | **75,367**  **5 iterime** | **2,516,537** |

Testimet me numer te iterimeve

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Nr. iterimeve | a | b | c | d | e | f |
| 5 |  |  |  |  |  |  |
| 10 |  |  |  |  |  |  |
| 15 |  |  |  |  |  |  |
| 20 |  |  |  |  |  |  |
| 40 |  |  |  |  |  |  |
| 50 |  |  |  |  |  |  |
| 60 |  |  |  |  |  |  |
| 70 |  |  |  |  |  |  |
| 80 |  |  |  |  |  |  |
| 90 |  |  |  |  |  |  |
| 100 |  |  |  |  |  |  |

Testimet me minuta

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Minutat | a | b | c | d | e | f |
| 5 |  |  |  |  |  |  |
| 10 |  |  |  |  |  |  |
| 15 |  |  |  |  |  |  |
| 20 |  |  |  |  |  |  |
| 40 |  |  |  |  |  |  |
| 50 |  |  |  |  |  |  |
| 60 |  |  |  |  |  |  |
| 70 |  |  |  |  |  |  |
| 80 |  |  |  |  |  |  |
| 90 |  |  |  |  |  |  |
| 100 |  |  |  |  |  |  |

# Future work

The present work is an initial step toward a more comprehensive nal goal. First of all, we will develop other, more elaborate, neighbourhood relations, speci cally designed for this problem. Secondly, we plan to investigate SA variants (see [7]) , alternative metaheuristics, and hybrid techniques. Finally, we aim at performing an instance space analysis and a corresponding algorithm selection procedure for this problem.

Regarding the rst point, we are currently developing a Kempe chain neigh- bourhood, such that possible con icts generated by a movement are repaired. In detail, in order to repair any new con ict introduced by moving exam e to period p and room r, the move KempeChain(e,p,r), reassigns all exams in p in con ict with e to the period originally assigned to e and to the cheapest room (greedily determined), and so on until there are no newly introduced con icts (see also [2]).

# References

* 1. Battistutta, M., Schaerf, A., Urli, T.: Feature-based tuning of single-stage sim- ulated annealing for examination timetabling. Annals of Operations Research 252(2), 239 254 (2017)
  2. Bellio, R., Ceschia, S., Di Gaspero, L., Schaerf, A.: Two-stage multi-neighborhood

simulated annealing for uncapacitated examination timetabling. Computers and Operations Research 132, 105300 (2021)

* 1. Bellio, R., Ceschia, S., Di Gaspero, L., Schaerf, A., Urli, T.: Feature-based tuning of

simulated annealing applied to the curriculum-based course timetabling problem. Computers & Operations Research 65, 83 92 (2016)

* 1. Burke, E.K., Bykov, Y.: An adaptive ex-deluge approach to university exam

timetabling. INFORMS Journal on Computing 28(4), 781 794 (2016)

* 1. Burke, E.K., Bykov, Y.: The late acceptance hill-climbing heuristic. European Jour- nal of Operational Research 258, 70 78 (2017)
  2. Ceschia, S., Di Gaspero, L., Schaerf, A.: Design, engineering, and experimen-

tal analysis of a simulated annealing approach to the post-enrolment course timetabling problem. Computers & Operations Research 39, 1615 1624 (2012)

* 1. Franzin, A., St tzle, T.: Revisiting simulated annealing: A component-based anal-

ysis. Computers & Operations Research 104, 191 206 (2019)

* 1. Gogos, C., Goulas, G., Alefragis, P., Kolonias, V., Housos, E.: Distributed scatter search for the examination timetabling problem. In: McCollum, B., Burke, E.K., White, G. (eds.) 8th International Conference on the Practice and Theory of Au- tomated Timetabling (PATAT-2010). pp. 211 223. PATAT, Belfast (2010)
  2. Leite, N., Mel cio, F., Rosa, A.C.: A fast simulated annealing algorithm for the examination timetabling problem. Expert Systems with Applications 122, 137 151 (2019)
  3. McCollum, B., McMullan, P., Burke, E.K., Parkes, A.J., Qu, R.: The sec- ond international timetabling competition: Examination timetabling track. Tech. Rep. QUB/IEEE/Tech/ITC2007/Exam/v4.0/17, Queen's University, Belfast (UK) (September 2007)