Sports timetabling: towards generic algorithms and performance insights

David Van Bulck[1,2][0000−0002−1222−4541] and Dries Goossens[1,2][0000−0003−0224−3412]

[1] Ghent University, Ghent, Belgium
[2] Core lab CVAMO, FlandersMake@UGent, Belgium
{david.vanbulck,dries.goossens}@ugent.be

Keywords: Sports scheduling, Algorithm benchmarking, Instance Space Analysis, ITC2021.

1 Introduction

For decades, sports timetabling has been a case-study driven field, with researchers developing tailor-made algorithms. The value of these case studies notwithstanding, the lack of a framework for algorithm benchmarking made it difficult to compare timetable requirements and algorithm performance across different studies. With the recent introduction of RobinX [4] and the International Timetabling Competition 2021 (ITC2021, [2]), a unified file format and a set of common benchmark instances is now available. Even though ITC2021 has paved the way for more generic algorithms it also indicates that, depending on the specifics of the sports tournament, some types of algorithms may be more suitable than others. This extended abstract is based on Van Bulck et al. [3] and demonstrates how to use techniques from instance space analysis to predict which algorithm is most suited for a given sports timetabling application. Our results are based on large computational experiments involving about 50 years of CPU time on more than 500 newly generated problem instances.

2 The International Timetabling Competition 2021

The task in ITC2021 is to construct a compact timetable for a double round-robin tournament, meaning that each team faces every opponent twice, both at home and away, with exactly one game per round. Constraints are categorized as hard or soft: hard ones are fundamental and cannot be broken, while soft ones are preferences. The objective is to respect as many soft constraints as possible, while adhering to all hard constraints. Nine constraint types are considered, grouped into four classes (see also [4]).

Capacity constraints regulate when teams play at home or away. We consider four types: CA1, CA2, CA3, and CA4.

Break constraints regulate the number of breaks (i.e., consecutive home or consecutive away games) in the timetable. We consider two types: BR1 and BR2.

Game constraints enforce or forbid specific assignments of games to rounds. We consider only one type: GA1.

Fairness and separation constraints are always soft and only two types are considered. FA1 limits the maximal difference in home games played by any two teams at any point in the season, whereas SE1 requests a minimal number of rounds between games with the same opponents.

In addition, some problem instances require a 'phased' timetable, meaning that each team plays against every other team once before any rematch takes place. ITC2021 offers a set of 45 problem instances, which are all expressed in the standard file format of RobinX. The number of teams varies from 16 to 20. Instances and best solutions are available from the competition website at itc2021.ugent.be.

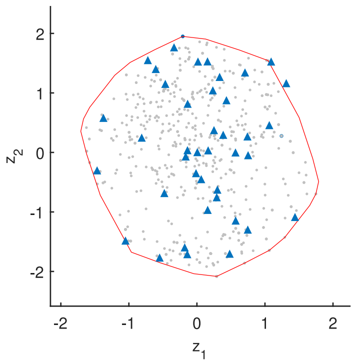
3 A Problem Type Analysis for ITC2021

Since the set of 45 problem instances from ITC2021 is rather limited to apply machine learning tools, we used the generator from [2] resulting in a diverse set of 518 additional instances. This was done in such a way that the instances are well scattered in the so-called two-dimensional (2D) problem type space (see Figure 1a). The axes of this 2D space correspond to linear combinations of the number and type of constraints (i.e., the problem type) present in each instance, combined in such a way that the performance of the algorithms linearly varies over the 2D space.

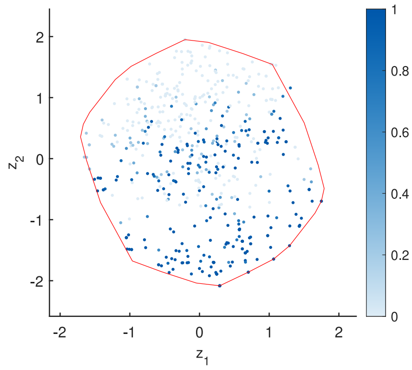
In order to get more insights into the composition of the problem type space, Figure 1 also shows the distribution of some prominent constraint types over the space. Besides, Figure 2 visualizes the regions where some of the algorithms we consider are expected to perform well. Goal and UoS are matheuristics (fix-and-optimize and variable neighborhood descent), FBHS shares similarities with the well-known first-break-then-schedule decomposition method, and Udine is a simulated annealing metaheuristic. For more details, see Van Bulck et al. [3].

Comparing the distribution of the problem type characteristics (cf., Figure 1) with the algorithm footprints (cf., Figure 2), we derive the following insights. First, the footprint of Udine shows that the algorithm not only finds a feasible solution for the majority of the problem instances, but also that the solutions found are of high quality. However, near the top of the instance space where there are many BR2 soft constraints, FBHS seems more promising. Finally, for instances near the middle-left to bottom-right diagonal, Goal and UoS are a suitable choice. As none of the problem type characteristics dominate in this region, these instances could be considered as 'average instances'. On the other hand, all algorithms struggle to find feasible solutions near the bottom of the space where the 'hardest instances' from our dataset are located. This part of the problem type space corresponds to problems that are phased or have SE1 soft constraints, in combination with many BR2 and CA4 hard constraints. On the other hand, near the middle of the space, and especially near the middle left, there are several instances for which multiple or even all algorithms find a good solution. Based on Figure 1, this area is characterized by the lack of BR2 hard and soft constraints.

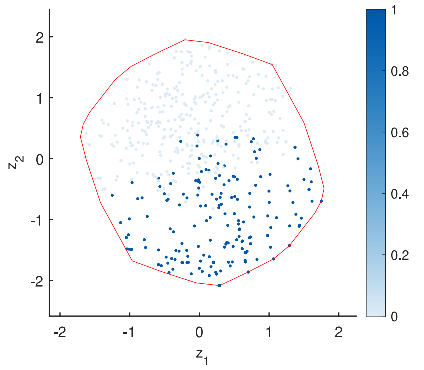
Finally, we note that we can also use the coordinates of the problem instances in the 2D problem-type space as input to a machine learning model to predict which algorithm is expected to perform best. The results of these predictions, made with the



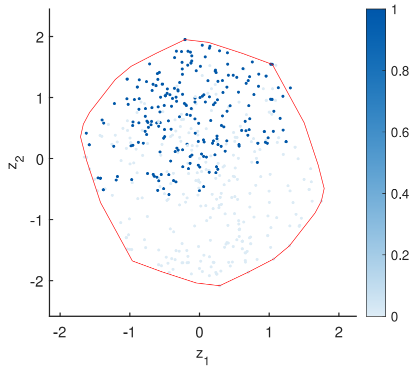
(a) Problem type space



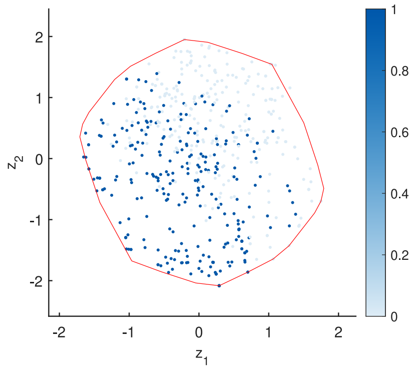
(b) CA4 Hard



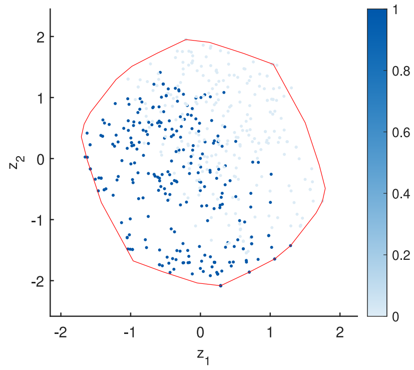
(c) BR2 Hard



(d) BR2 Soft

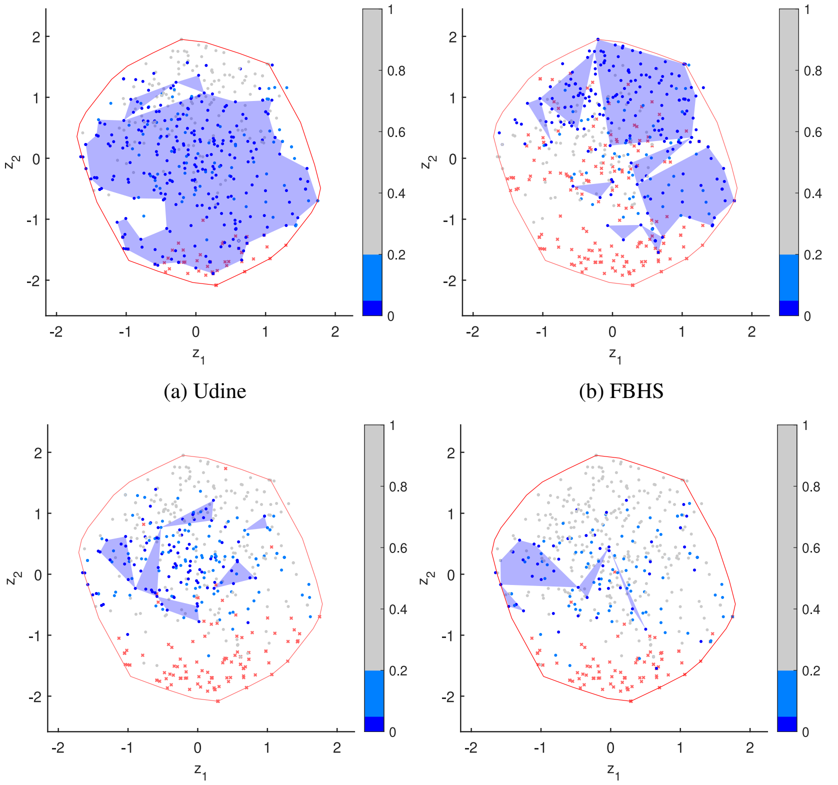


(e) SE1 Soft



(f) Phased

Fig. 1: Distribution of the number of constraints for some prominent constraint types over the so-called problem type space constructed with the ISA toolkit [10]. In (a), blue triangles denote the 45 original ITC2021 instances, whereas grey dots denote the 518 additional instances.

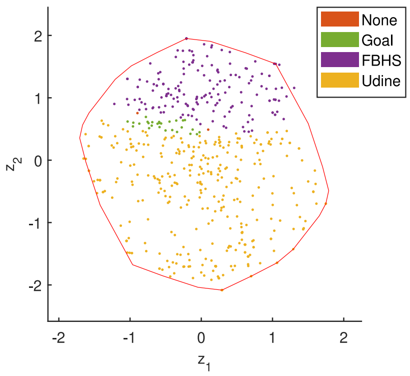


(a) Udine (b) FBHS

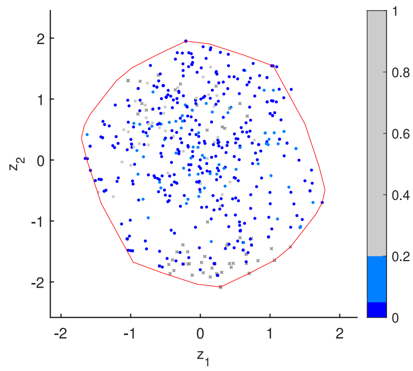
(c) Goal (d) UoS

Fig. 2: Algorithm footprints illustrating regions where the algorithm is expected to excel (blue regions, as identified by the ISA-toolkit [10]). Colours denote the relative gap compared to the best solution found by any of the algorithms, with red x-marks indicating instances for which no feasible solution was found.

ISA toolkit [10], and the quality of the predictions in terms of the average relative gap with regard to the best solution found by any of the algorithms can be found in Figure 3. Using the recommended algorithm for each instance, we are able to reduce the average relative gap for the Udine solver (the single-best algorithm) from 12.8% to just 4.93% (measured over a set of unseen test instances; always predicting the best algorithm results in a gap of 0%). This lets us conclude that we are able to effectively predict which algorithm a practitioner should use, when given the type of constraints typically present in the sports competition under consideration.



(a) Recommendations



(b) Performance

Fig. 3: Algorithm recommendations by the ISA toolkit [1] and performance thereof using as features the 2D coordinates in the problem type space. UoS was never predicted to be best.

References

1. Smith-Miles, K., Muñoz, M.A.: Instance space analysis for algorithm testing: Methodology and software tools. ACM Comput. Surv. **55** (2023)
2. Van Bulck, D., Goossens, D.: The international timetabling competition on sports timetabling (ITC2021). Eur. J. Oper. Res. **308**, 1249–1267 (2023)
3. Van Bulck, D., Goossens, D., Clarner, J., Dimitsas, A., Fonseca, G.H.G., Lamas-Fernandez, C., Lester, M.M., Pedersen, J., Phillips, A.E., Rosati, R.M.: Which algorithm to select in sports timetabling? Eur. J. Oper. Res. (2024), https://doi.org/10.1016/j.ejor.2024.06.005, in press
4. Van Bulck, D., Goossens, D., Schönberger, J., Guajardo, M.: RobinX: A three-field classification and unified data format for round-robin sports timetabling. Eur. J. Oper. Res. **280**, 568 – 580 (2020)