Effective classroom management with a self-generating memetic algorithm for course timetabling

Cevriye Altıntaş · Tuncay Yiğit · Özkan Ünsal

1 Introduction and Problem Description

Although the course timetabling problem is very complex due to many different characteristics of teaching space and constraints, various approaches have been developed to solve the problem. A course timetabling problem consists of the designation of a set of courses to a given set of timeslots subject to various practical constraints. A timetable that must satisfy all the hard constraints of a problem is called a feasible timetable. The hard constraints cannot be violated. Soft constraints represent preferences that can be violated, but in many cases solution approaches attempt to reduce the number of such violations as much as they can to improve the quality of a generated timetable further. More on course timetabling can be found in [6]. Suleyman Demirel University(SDU) as one of the public university in Turkey is among other public higher education institutions (HEIs) has take incentive to increase its number of students from 10000 students in 2009 to almost 30000 in 2015. These drastic increase of students enrolled has required SDU to provide more space for teaching and learning facilities and other related supporting facilities. As a leading public university, SDU has to review and further investigate its teaching and learning facilities to cope with those changes. Therefore we added extra soft constraint which is called Classroom Utilisation Rate. In this study, we focus on the course timetabling problem with effective classroom usage, which is one of the most important and repetitive administrative activities that occur in the educational institutions. This study presents a self-generating multimeme algorithm for solving the course timetabling problem at SDU. Unlike previous multimeme algorithms, each meme in the proposed algorithm encodes a score as a performance indicator of the associated operator. Those scores are then used in the process of choosing operators to create/modify new candidate solutions, self-adaptively. SDU also deals with the course timetabling issue once in a year. This problem is not that different than the course timetabling problems faced by the other educational institutions across the world. Recently, the second International Timetabling Competition (ITC2007) was organised

C. Altıntaş, T. Yiğit and Ö. Ünsal
Department of Computer Engineering, Süleyman Demirel University, Isparta, Turkey
Tel.: +90-246-211 1378
Fax: +90-246-237 0859
E-mail: cevriyealtintas@sdu.edu.tr, tuncayyigit@sdu.edu.tr, ozkanunsal@gmail.com

Proceedings of the 11^{th} International Confenference on Practice and Theory of Automated Timetabling (PATAT-2016) – Udine, Italy, August 23–26, 2016

Table 1 The characteristics of the SDU course timetabling problem instances and best results obtained from SGMA.

Instance	Courses	Lectures	Rooms	DaysxPeriods	Constraint	Cirrucula	SGMA Score	Utilisation R.
SDU 1	76	486	9	5x8	171	33	162	62.66
SDU 2	51	936	11	5x8	10	29	81	57.47
SDU 3	212	1266	20	5x8	626	113	255	35.66
SDU 4	132	807	18	5x8	294	72	160	77.92
SDU 5	141	892	21	5x9	830	86	21	60.14
SDU 6	78	571	31	5x8	414	47	146	53.31
SDU 7	36	236	13	5x8	166	29	47	47.76
SDU 8	60	331	12	5x9	49	46	73	45.42
SDU 9	83	595	14	5x8	324	49	177	45.45
SDU 10	76	464	13	5x8	324	41	120	27.22

with the goal of providing a set of real world problem instances and determining the stateof-the-art for educational timetabling. One of the competition tracks was on curriculum-based course timetabling and the instances used at the competition turned into a benchmark. SDU course timetabling problem is formulated in the same way as in the ITC2007. The SDU problems instances used in this study will be publicly provided extending the ITC2007 benchmark instances. The properties of each SDU instance is summarised in Table 1. The curriculum-based course timetabling problem was introduced for use in ITC2007. The problem involves four hard constraints whose satisfaction is mandatory, and four soft constraints, whose satisfaction is desirable, but not essential. Except the ITC2007 competition constraints, there are several additional constraints which came up during our work on course timetabling problem. Travel Distance: if there are two classes placed one after another so that there is no time slot in between (also called back-to-back classes), distances between buildings need to be considered [1], [4]. If the distance is between 50 and 400 meters counts as 1 violation. The standard measure for walking distance is 80 m for 1 min of walking time. Classroom Utilisation Rate (1): Space utilisation is a measure of whether and how space is being used. The utilisation rate is a function of a frequency rate (2) and occupancy rate (3) [3], [7]. Each classroom which has lower usage that is belove 30 in a curriculum counts as 1 violation.

$$ClassroomUtilisationRate(U) = \frac{FrequencyRatexOccupancyRate}{100} \quad (1)$$

$$FrequencyRate(F) = \frac{Numberofhoursusedduringweek}{Hoursallocatedduringweek}x100 \quad (2)$$

$$OccupancyRate(O) = \frac{Totalstudentnumbersduringweek}{Roomcapacityduringweek}x100 \quad (3)$$

2 Proposed Approach

A generic Memetic Algorithm is an evolutionary algorithm which makes heavy use of hill climbing as introduced by Moscato in [9]. The main components of an MA are mutation, crossover and hill climbing. In this study, we describe a novel Self- Generating Multimeme Algorithm (SGMA) that manages 6 mutation, 2 crossover and 2 hill climbing operators. The main feature of the proposed algorithm is that each meme encodes a score as a performance indicator of the associated operator. SGMA self-adaptively select which heuristic to use for different instances, stage of the search or individuals in the population. Memes can adapt through changes in their parameter set or through changes in the actions they perform. Algorithm 1 presents the pseudocode of our algorithm.

Algorithm 1 Pseudocode of a self-generating multimeme algorithm

```
Create a population of popSize random individuals.
// Apply hill-climbing on each individual
for i = 1 : popSize do
    OP_ID = Random-Choice(Hill-Climbing-Operators)
    Ind(i) ← Apply-Hill-Climbing(OP_ID, Ind(i))
end for
while termination criteria is not satisfied do
    for i = 1 : popSize − 1 do
        Parent1 ← Select-Parent(Population, tour-size)
        Parent2 ← Select-Parent(Population, tour-size)
        S_Meme = SelectMeme(Parent1,Parent2)            ▷ S_Meme represents Selected Meme
        Op_ID = Tournament_Select(S_Meme.Crossover-Operators)
        Offspr ← Apply-Crossover(Op_ID, Parent1, Parent2)
UpdateMeme(S_Meme, Op_ID, newfitness, oldfitness)
        Op_ID = Tournament_Select(S_Meme.Mutation-Operators)
        Offspr ← Apply-Mutation(Op_ID, Offspr)
UpdateMeme(S_Meme, Op_ID, newfitness, oldfitness)
        Op_ID =Tournament_Select(S_Meme.Hill-Climbing-Operators)
        Offspr ← Apply-Hill-Climbing(Op_ID, Offspr)
UpdateMeme(S_Meme, Op_ID, newfitness, oldfitness)
        Add(Offspr, Offspring-Pool)
            ▷ Tournament_Select is based on numberofoperatorbestscore of S_Meme.Operators
    end for
    Replacement: Replace the worst individual by Offspr
end while

function UpdateMeme(S_Meme, Op_ID, newfitness, oldfitness)
    if newfitness¿=oldfitness then
        numberofoperatorworstscore++
    else
        numberofoperatorbestscore++
    end if
end function
```

3 Conclusions

The performance of a self-generating multimeme algorithm for the course timetabling problem is investigated on a subset of ITC2007 and SDU instances. Each experiment is repeated 10 times and a run is terminated after 325 seconds complying with the ITC2007 competition rules. Feasible solutions are obtained for all problem instances used during the experiments. Table 1 provides the best results obtained by SGMA for solving SDU instances. SGMA performs reasonably well and Classroom Utilisation Rate is between 30 and 70. It is not the best approach, but it performs potentially better than other memetic approaches. Hence, we will be implementing different memetic algorithms and test them on all ITC2007 and SDU instances. Table 2 Performance comparison of our approach to the previously proposed approaches based on the best results (scores) obtained in 10 runs over the ITC2007 benchmark instances.

Acknowledgements The authors wish to thank Süleyman Demirel University Scientific Research Projects Management Unit Presidency that supported this Project financially with project number 4538-D1-16.

Table 2 Comparison of results.

Data Set	SGMA	Müller[10]	Lü and Hao[8]	Bellio et al.[2]	Cacchiani[5]	Soria-Alcaraz[11]
comp01	**4**	5	5	5	5	5
comp02	44	51	34	40	16	**15**
comp03	70	84	70	70	52	**30**
comp04	**35**	37	38	35	35	35
comp05	188	330	298	326	166	**68**
comp06	42	48	47	41	**11**	18
comp07	**6**	20	19	17	6	6
comp08	**37**	41	43	40	37	37
comp09	103	109	99	98	92	**35**
comp10	16	16	16	11	**2**	4
comp11	**0**	0	0	0	0	0
comp12	169	333	320	325	100	**90**
comp13	65	66	65	64	57	**55**
comp14	53	59	52	54	**48**	50
comp15	69	84	69	70	52	**30**
comp16	52	34	38	27	**13**	15
comp17	80	83	80	67	**48**	50
comp18	**25**	83	67	69	52	27
comp19	**48**	62	59	61	48	49
comp20	**2**	27	35	33	4	4
comp21	102	103	105	89	**42**	43

References

1. Belien J., Mercy A., Building university course timetables with minimized resulting student flows, In Proceedings of the Multidisciplinary International Conference on Scheduling: Theory and Applications (6), 1-4 (2013).
2. Bellio, R., Ceschia, S., Di Gaspero, L., Schaerf, A., Urli, T, A simulated annealing approach to the curriculum-based course timetabling problem, In Proc. of the 6th Multidisciplinary International Conference on Scheduling: Theory and Applications, (2013)
3. Beyrouthy, C., Burke, E. K., Landa-Silva, J. D., McCollum, B., McMullan, P., Parkes, A. J., Understanding the role of UFOs within space exploitation, In Proceedings of the 6th International Conference on the Practice and Theory of Automated Timetabling, Brno, 359-362,(2006).
4. Burke E. K., Drake J. H., McCollum B., Özcan E., Comments on: An overview of curriculum-based course timetabling. TOP: An Offcial Journal of the Spanish Society of Statistics and Operations Research, 23(2), 355-358 (2015).
5. Cacchiani, V., Caprara,A., Roberti R., Toth,P., A new lower bound forcurriculum-based course timetabling, Comput. Oper. Res. 40 (10) 24662477 (2013).
6. DiGaspero L.,McCollum B.,Schaerf A, The second international timetabling competition (ITC-2007):curriculum-based course timetabling (track 3), Technical Report QUB/IEEE2007/08/01,University of Udine DIEGM. Udine, Italy (2007).
7. Kasim R., Nor H. M., Masirin M., Idrus M., Assessing Space Utilisation for Teaching and Learning Facilities at the Higher Education Institution: A Case Study of G3 Building, Universiti Tun Hussein Onn Malaysia. OIDA International Journal of Sustainable Development, 4(5), 125-134 (2012).
8. Lü, Z., Hao ,J.-K. . Adaptive tabu search for course timetabling. European Journal of Operational Research, 200(1), 235, 244(2009).
9. Moscato, P., On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms, Caltech Concurrent Computation Program Report 826, California Institute of Technology (1989).
10. Muller, T., Itc2007 solver description: A hybrid approach, Annals of Operations Research 172(1), 429-446 (2009).
11. Soria-Alcaraz, J. A., zcan, E., Swan, J., Kendall, G., Carpio, M., Iterated local search using an add and delete hyper-heuristic for university course timetabling, Applied Soft Computing, 40, 581-593, (2016).