Advances in Artificial Intelligence Research (AAIR)
ISSN 2757-7422
https://dergipark.org.tr/tr/pub/aair

Vol. 1 (No. 2), pp. 43-51, 2021

Published online: Aug 15, 2021

Analysing the effects of classroom utilisation with a self-generating multimeme memetic algorithm for the exam timetabling problem

Cevriye Altıntaş [1,*], Tuncay Yiğit [2]

[1] Isparta Universityof Applied Sciences, Dept. of Computer Engineering, Isparta, Turkey
0000-0001-5928-3402

[2] Süleyman Demirel University, Dept. of Computer Engineering, Isparta, Turkey
0000-0001-7397-7224

Abstract

Universities have three main missions which are education, research and community service. The activities related to these missions gain vitality through the elements of "human" and "space." The Human element consisting of academic staff and administrative staff working in universities represents the university's human capital. The buildings titled as faculty, vocational school, hospital, laboratory, research center on the other hand, constitute the elements of space. These spaces, housing the activities of academic and administrative staff, also represent the physical capital of universities. Universities not only need to be continue their education processes successfully, but they should also be manage to the use of resources in the most effective way. The classroom is the center of the school activities. Classroom management is very different from planning and evaluating other space needs. Without an effective classroom management, the heavy investment in the school system could produce loss rather than gain. It relates to the effective classroom planning, management techniques and classroom utilisation in determining accurately how many students the facilities will adequately support. Classroom capacity utilization is an economics concept which refers to the extent to which a higher education institutes or a nation actually uses its available classroom capacity. Therefore, the relationship between whether used and how classroom is being used is very important. Classroom utilisation rate is a percentage-based ratio based on an occupancy rate and frequency rate. The frequency rate evaluates how many times that classroom is used compared to its availability, and the occupancy rate evaluates how many users can actually use the space at one time the classroom is compared to its actual capacity. The main purpose of this study is to analysis the classroom utilisation effects on exam timetabling problem with a self-generating memetic algorithm. The results from the analysis show that classroom utilisation rates for exam timetabling problem should be addressed fully by top management. This study also offers that in order to improve the classroom utilisation rate, higher education institutes should think about the occupancy rate as it is the determining agent affecting the utilisation rate.

Keywords: *Classroom Management, Exam Timetabling, Memetic Algorithm*.

1. Introduction

The exam timetabling(ETP) is a multi-dimensional optimisation problem because it cannot be solved in polynomial time. Exam scheduling is one of the most critical administrative actions that occurs in all higher education institutions [1]. The aim in the ETP, time and the availability of classrooms is maximization of the overall process for a certain period of time constraints in a series. There are many researchers who have suggested models and methods for solving ETP problem. However, the problem of assigning a number of exams into a restricted number of classrooms and the classroom utilisation rate have not been widely analyzed in exam timetabling studies. The assignment of exams to periods and the assignment of exams to classrooms should not be considered separately from each other [2]. A large number of exams and the various classroom capacities with difference in exam periods make the assignment between classrooms and exams complex. As well, the capacity of classrooms may play a significant role in producing the solution to the ETP. As it has been already defined in many publications, constraints are usually divided into two categories. First, those constraints are designated hard constraints. All of them must be satisfied under any circumstances, and only a timetable without violations of hard constraints can be considered as feasible. Second, those constraints are represented soft constraints. Some or all of them are not essential but should be fulfilled as much as possible. But, it is desired that all of the soft constraints should have minimum values. Constraints, improving the quality of education, raising the performance of students and teaching staff, teaching staff are based on a desire to respond to administrative needs. We may not obtain optimal solution, even though the whole capacity of classrooms exceeds the entire student population to take all exams for a given period.

*Corresponding author
E-mail address: cevriyealtintas@isparta.edu.tr

Received: 13/04/2021; Received in revised form: 18/04/2021; Accepted: 15/06/2021.

According to these theories, using the precise actual total capacity of classrooms as an input to ETP may not produce an appropriate solution to the classroom-assignment problem [3]. Space inventory management process plays an important role in any institution of higher education and requires comprehensive planning to be done in systematic way [4]. Higher education institutes not only need to be accomplished in teaching, in doing research and in expanding participation but they should also be efficient in order to perform in a manner which makes the best manage their properties [5]. Classroom management is more than the evaluation of space needs. It relates to the classroom planning, management process and classroom utilisation in determining accurately how many students the facilities will adequately support. Efficient use of space is as valuable as its effective usage. The latest standards for space use and related work practices should be strike the balance between efficiency and effectiveness on the basis of evidence. Classroom utilisation rate is an indicator of whether and how classroom is being used [6]. The analysis of classroom utilisation rate was conducted with the following our goals:

- Determine the current classroom utilisation rate
- Evaluate potential reduction in the number of classrooms
- Share data and analysis with the stakeholders
- Develop strategies that encourage better use of the available resource base
- Provide added space to departments for research, offices, etc
- Help students avoid scheduling conflicts
- Improve space utilisation on campus
- Use this data on an ongoing basis to assist in planning and investment decisions

The main purpose of this work is to analysis the use of classroom utilisation effects on exam timetabling problem with a self-generating memetic algorithm. The results from the analysis show that classroom utilisation rates for exam timetabling problem should be addressed fully by top management. In addition, this study also suggests that in order to improve the utilisation rate, higher education institutes should think the occupancy rate as it is the critical element affecting the utilisation rate [7]. In the higher education sector, the most important result in terms of public investments of the growth policy on the establishment of new state universities since the middle of the 2000s and the contingent increases made in the existing universities has been the increases of investment demands [8].

129 State Universities, in the Republic of Turkey, have taken incentive to increase its number of students from 3.072.986 students in 2000 to almost 7.131.252 in 2020.These drastic increase of students enrolled has required Turkey to provide more space for teaching and learning facilities and other related supporting facilities. Turkey has to review and further investigate its teaching and learning facilities to cope with those changes. Therefore, we analyzed the effects of classroom utilisation for Space Management. The most important criteria in measuring the effective use of spaces is the space utilisation rate. Each academic semester in higher education institutions in Turkey, the number of students taking the exam in turkey about 21.103.312. However, Coronavirus disease 2019 (COVID-19) has changed the use of classroom. The usage capacity of the classes has been reduced by half and it is more difficult to conduct exams during the pandemic process. In this study, we added a new hard constraint to the exam scheduling problem by analyzing the educational space per student. The hard constraint is the classroom utilization rate that we used in the course scheduling problem we made before regarding this hard constraint.

2. Literature review

There are many investigators who have introduced methods and algorithms for solving ETP problem. In recent years, examination timetabling problems have been studied by using some new hyper-heuristic algorithms that have been shown to be very practical in other related problems. For example, graph coloring, simulated annealing approach, an iterative greedy algorithm, very large neighbourhood search, neural network, tabu search, multistage approaches utilizing case based reasoning, memetic algorithm, fuzzy reasoning, ant colony optimization, hyper-heuristics and hybrid approaches [9]. Evolutionary algorithms are the most frequently used methods for the ETP. Interested researchers can receive more information about examination timetabling study in [10] and [11]. Effective space inventory management relies on information about people, places, and processes. It supplies correct data to administrator by developing consistent space inventory system. The problem of space inventory management in the Turkish Public Universities are from the incorrect data, lack of centralized data administration and uneffective systems implemented. Space management practice requires continuous effort and slight progress has been made in raising awareness of space costs and possible for savings from improved space management. The most important criteria in calculating the space performance is how much the space is used. UK Higher Education Space Management Group (SMG, 2006) aimed to assist other higher education institutions in the definition and implementation of best application in efficient space management in their institutions. [12]. According to SMG the achievement of beneficial space use may provide organizations with a fit-for-purpose physical asset and potentially release funds to supply greater quality of academic learning and student acquisition. Their objective is provided space management applications and good applicable information to institutions and funding committees to allow all to make reliable, rational and informed decisions regarding the property and university mission.

SMG has been studying on space management since 1960 and in year of 1996 published a survey. National Audit Office (NAO) has succeeded in publishing early guideline to present space utilisation study for their higher education institutions [13]. Later, different applications have been reported by SMG. Some higher education institutions use timetabling program for their study and others through examinations. Though there are differences, they attempt to accomplish the same purpose; to advance their space management procedures. There are two different problem of space allocation, a space utilisation and a constraint satisfaction/optimization [14]. In this study, we proposed a new approach to analyze the space utilisation. Table 1 presented the classroom utilisation values for some faculties.

Table 1. *Classroom utilization rate for faculties*

Academic Units	Number of Units	FR %	OR %	UR %	UFA m^2	Number of Students	UFA/ Number of Students m^2
Faculty of Applied Sciences	15	30,16	32,38	10	10.438,49	4.907	2,13
Faculty of Architecture	42	32,93	36,94	13,92	142.785,38	32.364	4,41
Faculty of Communication	53	23,04	27,72	7,54	105.430,93	58.978	1,79
Faculty of Economics and Administrative Sciences	124	39,92	42,89	19,82	600.309,78	325.804	1,84
Faculty of Education	90	34,67	36,46	15,22	656.819,10	216.015	3,04
Faculty of Engineering	107	28,39	30,47	10,73	1.240.641,91	293.134	4,23
Faculty of Fine Arts	61	27,8	30,98	10,38	255.407,87	32.984	7,74
Faculty of Health Sciences	95	30,69	32,58	12,09	214.501,39	96.294	2,23
Faculty of Technology	20	21,1	21,58	5,3	209.479,43	37.221	5,63
Vocational School	610	27,06	27,06	8,96	2.015.401,40	560.889	3,59
Average Scores		**29,576**	**31,906**	**11,396**			**3,663**

3. Modeling of Exam timetabling Problem

Educational timetabling problems are among the most studied scheduling problems including high school scheduling, university examination and course timetabling. In recent years, a huge number of papers published and many approaches have been developed to solve the timetabling problems. These problems draw much research effort due to its difficulty. Many benchmark problems in the exam literature still have not been solved to optimality. In this work, we concentrate on the ETP, which is one of the most significant administrative process that take place every educational organizations. The ETP comprises of the assignment of a set of exams to a certain list of periods subject to some constraints [15]. The number of examinations to be scheduled change greatly based on the institution. The created exam timetable which is called suitable timetable must satisfy all hard constraints of the problem, whilst violation of the soft constraints should be minimised. Soft constraints may not be satisfied, but in many circumstances, solution approaches try to minimize the number of violations as much as they can to raise the quality of a produced timetable further [16]. Tables 2 and 3 show the constraints of the exam timetabling problem.

Table 2. *Hard Constraints*

No	Hard Constraints
HC1	Assigning multiple exams to any student in the same time period should be avoided.
HC2	The capacity of a classroom can not be exceeded at a given period.
HC3	The exam durations should not violate the period length.
HC4	An exam may need to be scheduled before/after another
HC5	Schedule exam in a particular classroom (classroom-related hard constraints).

Table 3. *Soft Constraints*

No	Soft Constraints
SC1	Students should not take exams in successive periods and preferably not in the same day
SC2	The number of events in which a student takes two exams on the same day. If the exams assign back to back, SC1 violation is considered to prevent duplication.
SC3	Each group of student examinations should be distributed as equally as possible throught the exam sessions.
SC4	Minimize the number of exams have a dissimilar length that is assigned in the same classroom.
SC5	Minimize the number of the largest exams are assigned close to the end of the exam session.
SC6	Minimize the number of exams scheduled in the period which have a related penalty.
SC7	Minimize the number of exams assigned in a classroom which have a related penalty.
SC8	Exam must be assigned to a classroom with lower utilization rate.

3.1. Problem Formulation

The variables used in the definition and formulations of the examination timetabling problem are given in Table 4:

Table 4. *Definition of ETP*

N	the number of exams
E_i	defines a collection of N examinations where $i \in \{1,\dots,N\}$
e_i	the number of students enrolled in exam E_i where $i \in \{1,\dots,N\}$
B	the set of all N exams, $B=\{E_1,E_2,\dots,E_N\}$
D	the number of days
P	the number of periods
M	the number of students
R	the number of available classrooms
L_f	the capacity of classroom f where $f \in \{1,\dots,R\}$
U_f	the utilisation rate of classroom f where $f \in \{1,\dots,R\}$
r_i	defines the assigned classroom for exam, where $r_i \in \{1,\dots,R\}$, $i \in \{1,\dots,N\}$
t_i	defines the assigned period for exam, where $t_i \in \{1,\dots,P\}$, $i \in \{1,\dots,N\}$
d_i	defines the assigned day for exam, where $d_i \in \{1,\dots,D\}$, $i \in \{1,\dots,N\}$
$C=(c_{ij})_{N\times N}$	defines a matrix in which each record indicated by c_{ij}, $(i, j \in \{1,\dots,N\})$ represents the number of students sitting exams E_i and E_j
$\Delta_t = \lvert t_i - t_j \rvert$	the period different between exam E_i and E_j
$\Delta_d = \lvert d_i - d_j \rvert$	the day different between exam E_i and E_j

The constraints of our dataset are described Eq.(1-10) [17]:

1) All exams must be assigned and each exam has to be assigned only once.

$$\sum_{s=1}^{T} \lambda_{is} = 1 \text{ for all i } \in \{1,\dots,\text{N}\}, where\ \lambda_{is} = \begin{cases} 1 \text{ if exam i is assigned to} \\ 0 \text{ otherwise} \end{cases} \tag{1}$$

2) A student sitting in two exams concurrently (student-conflict) must not occur. If examination i and j are planned in slot *s*, the number of students taking both examination *i* and *j* must be equal to zero, i.e. $c_{ij} = 0$.

$$\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} c_{ij}.x(t_i,t_j) = 0 \text{ ,} where\ x(t_i,t_j) = \begin{cases} 1 \text{ if } t_i = t_j \\ 0 \text{ otherwise} \end{cases} \quad (2)$$

3) For each period t, the number of students enrolled exams (*Students$_t$*) cannot exceed the maximum capacity for each individual classroom (*Seats*) .

$$Students_t \leq Seats\ for\ \in \{1 \dots P\} \quad (3)$$

4) Student who takes the exams on the same day consecutively should be assigned to the same classroom, i.e. both exams are assigned to the same classroom.

$$\text{If } t_i = x; t_j = x+1; d_i = d_j \text{ and } c_{ij} \neq 0 \text{ then } r_i = r_j \text{ for all } i, j \in \{1,\dots,N\} \quad (4)$$

5) Particular examination, $E_i \in S$ where $S \subset B$ should be seperated from other exams, i.e. the particular exam cannot share classroom with different exam at the same period.

$$\sum_{i=1}^{N} \alpha_{ir} \leq 1 \text{ for all r} \in \{1,\dots,\text{R}\}\text{ ,} where\ \alpha_{ir} = \begin{cases} 1 \text{ if exam } E_i \in S \text{ is assigned to classroom r} \\ 0 \text{ otherwise} \end{cases} \quad (5)$$

6) No student can seat maximum number of exam consecutive exams in a day.

$$If\ c_{ij} \neq 0; c_{ik} \neq 0;\ t_i = x;\ [t_j = x+1\ or\ t_j = x-1] and\ d_i = d_j\ then\ d_k \neq d_i\ for\ all\ i,j \in \{1,\dots,N\} (6)$$

7) If it is possible, each examination must be assigned to one classroom.

$$\sum_{f=1}^{R} \beta_{if} = 1\ for\ all\ i\ \epsilon, where\ \beta_{if} = \begin{cases} 1 \text{ if exam } i \text{ is assigned to classroom } f \\ 0 \text{ otherwise} \end{cases} \quad (7)$$

8) Exam must be scheduled to a classroom without exceed the classroom capacity.

$$\sum_{i=1}^{N} e_i.\beta_{if} \leq L_f\ \ for\ all\ f\ \epsilon\ S \quad (8)$$

9) Exam must be assigned to a classroom with lower utilization rate.

$$\sum_{i=1}^{N} e_i.\beta_{if} \leq U_f\ \ for\ all\ f\ \epsilon\ S \quad (9)$$

3.2. Fitness Function

The fitness function is based on Cost Penalty [18,19].

$$Minimize\ F = \frac{\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} c_{ij}.Penalty(t_i,t_j)}{M} \quad (10)$$

3.3. Classroom Utilisation Rate

Classroom Utilisation Rate Eq. (11): Classroom utilisation is a measure of whether and how space is being used. The utilisation rate is a function which comes from a frequency rate Eq. (12) and occupancy rate Eq.(13) [12]. Each classroom which has lower usage that is below 30 in a curriculum counts as 1 violation and below 75 in a cirriculum counts as 2 violation.

$$\text{Classroom Utilisation Rate (UR)} = \frac{\text{Frequency Rate xOccupancy Rate}}{100} \quad (11)$$

$$\text{Frequency Rate (FR)} = \frac{(\text{Number of hours used during exam periods}) * 100}{(\text{Hours allocated during exam periods})} \quad (12)$$

$$\text{Occupancy Rate (OR)} = \frac{(\text{Total student numbers during exam periods}) * 100}{(\text{Classroom capacity during exam periods})} \quad (13)$$

3.4. Parameters

There are various parameters for our self-generating memetic algorithm and the local improvement step of the algorithm. The self-generating memetic algorithm can be controlled via the following parameters:

Population size *p*: A small population size can lead to a less diverse population and not enough memetic material. However, with a large population size, it will take a long time to get the solution.

Construction strategy *c*: Sets the strategy to construct the initial population. Our strategy yields completely random solutions.

K-tournament competitors *k*: The number of competitors for the k-tournament in the selection phase of the algorithm. Must be in the interval [1, p]. Depending on the scores encoded in the meme of Parent1, crossover low-level heuristics, mutation low-level heuristics, local search low-level heuristics, and mutation density and depth of search values are chosen (using tournament selection). In any case, the operator or parameter setting with the best value among the tournament size of options, which indicates higher historical performance, is chosen. In the case of equal scores, the random option is selected [18, 19].

Depth of Search *dp*: This parameter controls the number of individuals to be improved in the local search phase of the algorithm. It lies in the interval [0, 1]. To get the number of individuals which are improved, it is multiplied with the population size p.

Intensity of Mutation *im*: The share of individuals to be mutated each generation is specified by this parameter. Must be in the interval [0, 1].

Our parameter configuration is shown in Table 5.

Table 5: *Parameter configuration SGMMA.*

Parameter	**Value sequence**	**Default value**
Population size *p*	[10, 20, 50, 100]	50
K tournament competitors *k**	[0.0, 0.15, 0.35, 0.5]	0.15
Depth of Search *dp*	[0.2, 0.4, 0.6, 0.8, 1.0]	0.2
Intensity of Mutation *mf*	[0.2, 0.4, 0.6, 0.8, 1.0]	0.2

4. Self-Generating Multimeme Memetic Algorithm

A generic Memetic Algorithm (MA) as introduced by Moscato in [20], is an evolutionary algorithm which uses the feature of GA with some search heuristic algorithm like hill climbing, an iterative greedy algorithm, simulated annealing, tabu search etc. The main components of MA are crossover, mutation and local search. Members of the first population are produced using a weighted roulette wheel technique to select the period in which to place each exam in order to generate a high quality timetable. Mutation operators are divided into two, the light operator moving a number of exams to new available periods, the heavy operator disturbing whole periods. Hill climbing is then applied, taking the exams in each period in turn and checking all other periods to move an exam to the period of lowest penalty. The evaluation function penalises unassigned exams heavily as well as calculating the number of conflicts in the timetable between two periods on the same day. Roulette wheel selection is again applied during the selection phase in order to control a defined population size. In this work, we represent a useful Self-Generating Multimeme Memetic Algorithm (SGMMA) that manages to 6 mutational, 2 crossover and 2 local seach operators. The first population is made up of using multiple constructive low-level heuristics with the aim of producing suitable initial solutions. The main characteristic of the suggested algorithm is that each meme encodes a score as a performance pointer of the as sociated operator. Finding efficient parameters for an algorithm is very challenging if a large number of parameters are available [21]. This is because the resulting combinatorial space of parameter settings is extremely large. Parameter tuning, if performed manually by hand, is tedious work. And if the number of parameters increase, the combinatorial space of parameter settings is too large to be managed manually. Self-adaptation is very important, if there is more than one operator and there are a number of values that can be selected for from the parameter settings. Evolutionary algorithm can be categorized according to the adaptation type as steady, trangenerational, adaptive and self-adaptive. A steady-state algorithm is not taking into acount any feedback during the search phase to modify the selection mechanism. On the other part, an adaptive algorithm uses online feedback to manage the selection of a local optimization algorithms. In this study, we propose a Self-Adaptive Multimeme Memetic Algorithm which employs a novel encoding for the self-adaptation of memetic operators and their parameter values. During the evolutionary phases, when it is time to use an operator of determined type, e.g., mutation, one of the operators is chosen and applied randomly using roulette wheel selection depend on the scores of operators of that type. The pseudocode for SGMMA is given in Algorithm 1.

Algorithm 1 Pseudocode of Self-Generating Multimeme Memetic Algorithm (SGMMA) fort he ETP

Generate initial population of *Population_Size* random individuals
for *i* =1 : *Population_Size* **do**
  OpID =Random-Select(Hill-Climbing-Operators)
  Ind(*i*) Apply-Hill-Climbing(OpID,Ind(i))
end for
while termination criteria is not satisfied **do**
for *i* =1 : *Population_Size*−1 **do**
  Parent$_1$← Select-Parent(Population, tour-size)
  Parent$_2$ ← Select-Parent(Population, tour-size)
  S_Meme=*SELECTMEME*(Parent$_1$, Parent$_2$)//S_Meme represents Selected Meme
  OpID =Tournament Select(S_Meme.Crossover-Operators)
  Offspring_Apply-Crossover(OpID, Parent$_1$, Parent$_2$)
  UPDATEMEME(S_Meme, OpID, newfitness, oldfitness)
  OpID =Tournament Select(S_Meme.Mutation-Operators)
  Offspring_Apply-Mutation(OpID, Offspr)
  UPDATEMEME(S Meme, OpID, newfitness, oldfitness)
  OpID =TournamentSelect(S_Meme.Hill-Climbing-Operators)
  Offspring_Apply-Hill-Climbing(OpID, Offspr)
  UPDATEMEME(S_Meme,OpID,newfitness, oldfitness)
  Add(Offspring,Offspring-Pool)
  Tournament Select is based on number of operator best score of S_Meme.Operators
end for
  Replacement: Offspring replaces the worst individual in the population
end while
function *SELECTMEME*(Parent$_1$, Parent$_2$)
  if both parents carried the same meme **then**
   MemeID =Meme.Crossover-Operator
  elseif Parent$_1$.fitness== Parent$_2$.Fitness **then**
   SelectedMeme=Random-Choice(Parent$_1$.Meme, Parent$_2$.Meme)
   MemeID =SelectedMeme.Crossover-Operator
  else
   SelectedMeme=BestMeme(Parent$_1$.Meme, Parent$_2$.Meme)
   MemeID= SelectedMeme.Crossover-Operator
  end if
end function
function *UPDATEMEME*(S Meme,OpID, newfitness, oldfitness)
  if newfitness:=oldfitness **then**
   numberofoperatorworstscore++
  else
   numberofoperatorbestscore++
  end if
end function

5. Numerical Results

The best and the average penalty from 10 independents runs are presented in the following Table 6. The performance of a self-generating multimeme memetic algorithm for the exam timetabling problem was analyzed on some instances from two universities. Each experiment is run for a notional duration of 325 seconds and repeated 10 times. Feasible timetables are gained from all instances used during the experiments. Table 6 provides that the best results gained by SGMMA for solving instances and Figure 1 shows that comparision of the UR. SGMMA performs pretty well and Classroom Utilisation Rate is between 30 and 70. It is not good algorithm, but it performs potentially grater than other evolutionary algorithms. It is further that other local search techniques may also be mixed in different steps of MA and can be tested on a number of different benchmark problems.

Table 6. *Performance of SGMMA*

Instance	Approach	Best penalty	Average Penalty	Utilisation Rate(%)
1	SGMMA	4	5	62.66
2	SGMMA	60	62	57.47
3	SGMMA	293	295	35.66
4	SGMMA	27	29	77.92
5	SGMMA	15	16	60.14
6	SGMMA	4	4	53.31
7	SGMMA	800	2000	47.76
8	SGMMA	52	55	45.42
9	SGMMA	11	11	45.45
10	SGMMA	5	6	27.22

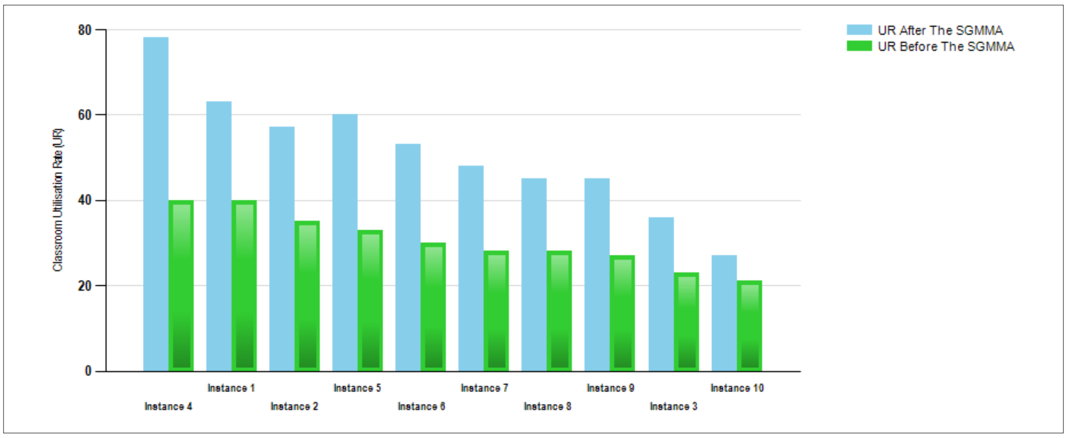

Figure 1.*Comparision of Classroom Utilisation Rates After Applied SGMMA*

6. Conclusions and Future Work

Evolutionary algorithm can be categorized according to the adaptation type as steady, trangenerational, adaptive and self-adaptive. An adaptive algorithm uses online feedback to manage the selection of a local optimization algorithms. Parameter setting and control are very important in many evolutionary algorithms. Finding efficient parameters for an algorithm is very challenging if a large number of parameters are available. This is because the resulting combinatorial space of parameter settings is extremely large. Parameter tuning, if performed manually by hand, is tedious work. And if the number of parameters increase, the combinatorial space of parameter settings is too large to be managed manually. Self-adaptation is very important, if there is more than one operator and there are a number of values that can be selected for from the parameter settings. In this study, we propose a Self-Adaptive Multimeme Memetic Algorithm which employs a novel encoding for the self-adaptation of memetic operators and their parameter values.

We have successfully integrated MA's into the examination timetabling problem. We measured the space utilisation rate in two universities. We created new formula for new/additional hard constraint. In future we plan to: use different algorithm and a more intelligent selection mechanism for choosing exam timetabling problem. Managing restricted spaces in the best way. Coronavirus disease 2019 (COVID-19) has changed the use of classroom, so it will become very important to increase classroom utilization rates using less capacity to adapt to this new situation. The usage capacity of the classes has been reduced by half or less and it is more difficult to conduct exams during the pandemic process. Furthermore, the effectiveness of expenditures based on space, one of the most important items of operating expenses of the universities, can only be achieved by rational space planning. We encourage to develop an effective space management system during the pandemic process.

Acknowledgements

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

References

[1] B. McCollum, P. McMullan, A. Parkes, E. Burke, R. Qu, A new model for automated examination timetabling, Annals of Operations Research 194(1), 291315 (2012).

[2] Qu, R., Burke, E.K., McCollum, B., Merlot, L., Lee, S.: A survey of search methodologies and automated system development for examination timetabling. Journal of Scheduling 12(1), 55-89 (2009).

[3] Muller, T.: Itc2007 solver description: A hybrid approach. Annals of Operations Research 172(1), 429-446 (2009).

[4] Amaratunga, D., Baldry, D. (2000) Assessment of Facilities Management Performance in Higher Education Properties Facilities Management, 18(7/8), 293-301.

[5] Pursglove, J., & Simpson, M. (2007). Benchmarking the performance of English universities. Benchmarking: An International Journal, 14, 102-122.

[6] Rogers, C. (2002). Space Management in Higher Education: Report of the Findings of the Newcastle University Space Management Project, Jointly Funded by the HEFCE Good Management Practice Programme and the University (pp. 1-88). Newcastle: University of Newcastle.

[7] HEFCW. (2002). Space Management: A Good Practice Guide. Swansea: University of Wales.

[8] SCHEV. (2004). Space Utilization and Comparison Report. Richmond: State Council of Higher Education for Virginia.

[9] Tan, J. S., Goh, S. L., Kendall, G., & Sabar, N. R., A survey of the state-of-the-art of optimisation methodologies in school timetabling problems. Expert Systems with Applications, 165, 113943, (2021).

[10] C. Gogos, P. Alefragis, E. Housos, A multi-staged algorithmic process for the solution of the examination timetabling problem, in: Practice and Theory of Automated Timetabling (PATAT 2008), Montreal, 19-22, August (2008).

[11] M. Atsuta, K. Nonobe, T. Ibaraki, Itc2007 track 1: An approach using general csp solver, www.cs.qub.ac.uk/itc2007.

[12] Space Management Group. (2006). Space Utilisation: Practice, Performace and Guidelines. London.

[13] NAO. (1996). Space Management in Higher Education: A Good Practice Guide. London: National Audit Office.

[14] Aziz A. A., Hashim A. E., Baharum Z. A. (2013). Space Inventory Management in the Malaysian Public Universities Social and Behavioral Science

[15] G. De Smet, Itc2007 - examination track, in: Practice and Theory of Automated Timetabling (PATAT 2008), Montreal, 19-22, August (2008).

[16] N. Pillay, A developmental approach to the examination timetabling problem., in: Practice and Theory of Automated Timetabling (PATAT 2008), Montreal, 19-22, August (2008).

[17] B. McCollum, P. McMullan, A. J. Parkes, E. K. Burke, S. Abdullah, An extended great deluge approach to the examination timetabling problem, in: The 4th Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA09), Dublin, (2009).

[18] E. Özcan, et al., A self-adaptive multimeme memetic algorithm co-evolving utility scores to control genetic operators and their parameter settings, Applied Soft Computing 49,81-93, (2016).

[19] S. Abdul-Rahman, A. Bargiela, E. K. Burke, E. Ozcan, B. McCollum, Linear Combination of Heuristic Orderings in Constructing Examination Timetables, European Journal of Operational Research, 232(2), 287-297 (2014).

[20] P. Moscato, On evolution, search, optimization, genetic algorithms and martial arts:Towards memetic algorithms, Caltech concurrent computation program, C3P Report, vol.826, p. 1989, (1989).

[21] Huang, C., Li, Y., & Yao, X., A survey of automatic parameter tuning methods for metaheuristics. IEEE transactions on evolutionary computation, 24(2), 201-216, (2019).