A New Intense Stochastic Search Method Based On Hypercube Evaluation for Examination Timetabling Problems

Mustafa Tunay
Department of Computer Engineering
Istanbul Gelisim University
Istanbul, Turkey
mtunay@gelisim.edu.tr

Abstract—**This study of article presents a novel optimization search algorithm to solve the examination timetabling problems. These are discrete, multi-objective and combinatorial optimization problem which is generally solved using stochastic search techniques as evolutionary algorithms (EAs) and heuristic technique. In this study, the purposed optimization search algorithm is to give a brief introduction to examination timetabling problems that have been solved using Hypercube Optimization Search (HOS) algorithm which is a new intense stochastic search technique that is based on a hypercube evolution. The proposed method is applied to solve a timetabling problem which is hard to solve for many institutions of higher education and as a result this approach has shown good performance when the proposed method is compared with other performances of existing approaches.**

Keywords—evolutionary algorithms, heuristic method, stochastic search, hypercube evolution, timetabling problem

I. INTRODUCTION

Timetabling problems are a huge subclass of plan for carrying out a process or procedure as schedule problems since the 1960s exam timetable has been one of the most studied subjects in timetable problem research in [1-4]. A timetable is explained as, a scheduling of exams under set of constraints related to keeping away the overlapping of exams with students in common, satisfying room and time constraints, teacher availability, invigilator availability. These constraints can be generated as hard constraints; that is, it can't be violated. Next one is soft constraints that must minimize as much as possible in studied as [5-6].

The examination timetabling is a difficult and time consuming educational problem which has to be tackled annually or quarterly that overview different aspect of educational timetabling research in [7-8].

There are many new methods and improvements for known evolutionary algorithms used in optimization. The early heuristics approaches are based on graph coloring that includes in [2-3]. Hybrid techniques using constraints have been applied to timetable problems [9-10]. The significant number of researches is related the application of meta-heuristics to the examination timetable. They are tabu search [11], simulated annealing [12-13] and evolutionary algorithms [14]. Other algorithms that were used for solving the schedule examination problems that are a hybrid variable neighborhood search with a genetic algorithm [15], neighborhood search approach hybridized with local search methods [16], heuristic search within great deluge [17-18], ant algorithms [19-20], bees algorithm [21] hybrid swarm-based algorithm [22], hybrid discrete particle swarm optimization [29-30] for solving timetabling problems. As shown number of different many approaches are used to find better solutions for the examination timetabling problem. In this study, Hypercube Optimization Search (HOS) algorithm [24,31] which references the best elements from an evaluation function to solve applied examination scheduling problems is designed. This method can be expressed as a hypercube evolution and it is a driven by convergence.

This study of article is presented as follows: section 2 describes a novel optimization search algorithm for solving timetabling problems. The descriptions of the processes used in HOS algorithm are presented. Section 3 describes examination timetabling problems. Section 4 describes dataset specification. Section 5 presents the experimental results for dataset from universities and comparative results with hybrid combination of approaches. Finally, section 6 gives the conclusion.

II. A NOVEL OPTIMIZATION SEARCH ALGORITHM

This study of article is devoted to the design of a novel evolutionary search algorithm to solve optimization of multivariate systems. This optimization search algorithm is a new intense stochastic search method that is based on a hypercube evolution driven by convergence. The proposed algorithm is inspired from the behaviors of a pigeon discovering new areas for seeds (food) in natural world. The behavior of pigeon moves according to specific rules. In fact, the aim of pigeon discovers new places (areas) for food like as in natural life. In such behavior, a flying pigeon searches for new locations of seeds. The pigeon moves 360^0 swivel in a unique way for marked areas and seeds. The pigeon moves around again and the behavior of the pigeon moves towards the area where there is accumulation of location by eating the seeds to in its surrounding one by one and changes its shrink by the sizes of the search area for a higher accumulation of location of seeds. The pigeon may be found in undiscovered branches in the marked area. The pigeon stops moving and keeps in mind the area which has seeds.

978-1-7281-7116-6/20/$31.00 ©2020 IEEE

The pigeon maintains jumping or moving to another area to find a new one. The pigeon does not move to another area when it gets to an area that has the most seeds. For this investigation, some experiments have been done with the pigeons in real life. The behavior of a pigeon is simulated to searching seeds in Figure 1 in real life.

The Behavior of a Pigeon in Real Life

Fig. 1. The behavior of a pigeon is simulated to searching seeds in real life

The structure of the HOS algorithm used is main processes. The flowcharts of HOS algorithm mainly presents *Fly process*, *Displacement-Shrink process* and *Searching Areas process* are described in Figure 2. The functions of each block are described and mathematical formulas and basic operations used in each processes. The HOS algorithm is inspired by the behaviour of pigeon to discover new areas for seeds (food) in natural life. This proposed algorithm is modified timetabling method for solving exam timetabling problems.

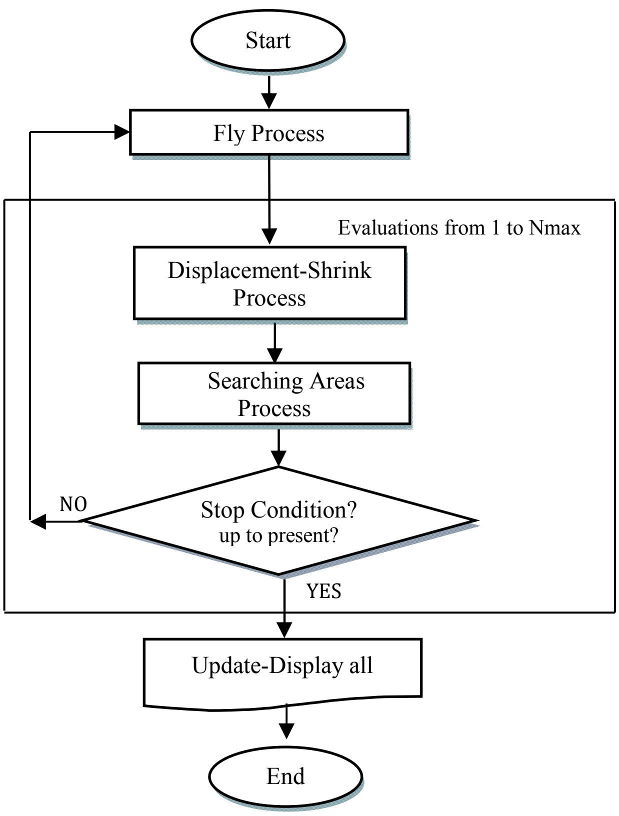

Fig. 2. The structure of the HOS algorithm

III. Examination Timetabling Problems

In this section presents the real-life application of the HOS algorithm is modified into a new method for solving exam timetabling problem. As mentioned the timetabling problem can be considered; the exams of students should be scheduled as a set of constraints within certain time interval. The input elements for this kind of problems are a set of exams to be scheduled timeslots and enrollment of student exams. They have considered the timetabling problem by using hard and soft constraints. Hard constraints are related to the "conflict" constraints and it cannot be violated under any circumstances. The common accepted hard constraints is studied as [3, 5-6, 23] for examination time-tabling problems are:

- A student cannot sit two examinations at once (3 periods in a day),
- Seating capacity (chairs-tables) for exam classes and number of students in an each period,
- The order for exams (for instance: the exam A should be before exam B),
- In a specific room (exam classes) for exams.

On the other hand, soft constraints are more related to the distribution of exams over timetable. The soft constraints are: the student should not be scheduled to have consecutive exams; the student's exam should be equally allocated over exam period in [15,27-28] note that the soft constraints for each university are usually different. In this paper "the conflicting of exams having students in common" is taken as hard constraint and "the students' exams should be spread as much as possible during the timetable". This is taken as a soft constraint.

There have been many studies on finding new methods and improvements for known evolutionary algorithms used in optimization. In this study of article, a novel Hypercube Optimization Search (HOS) method which references the best elements from an evaluation function to solve applied examination scheduling problems for optimization is designed. This method can be expressed as a hypercube evolution driven by convergence. The actual processes used by the flowchart of HOS method and is shown in Figure 3.

The examination timetable problems consist of allocating a number of exams to a limited number of periods with the specific (hard and soft) constraints. For each period, these constraints mainly form the basis for a feasible scheduling of the exams timetabling problem. For this purpose, the datasets of the examinations of the students of some universities were used (Table I). A parameter c_{ij} was defined in Table I to present the density of conflicts. If exam i conflict with exam j then c_{ij} = 1, otherwise c_{ij} = 0. The conflict density describes the ratio between the numbers of elements of value "1" to the total number of elements in the conflict matrix.

The model can be formulated as the minimization of the following objective function in [7].

$$F = \frac{\sum_{i=1}^{N-1} F_1(i) + \sum_{i=1}^{N-1} F_2(i)}{M} \tag{1}$$

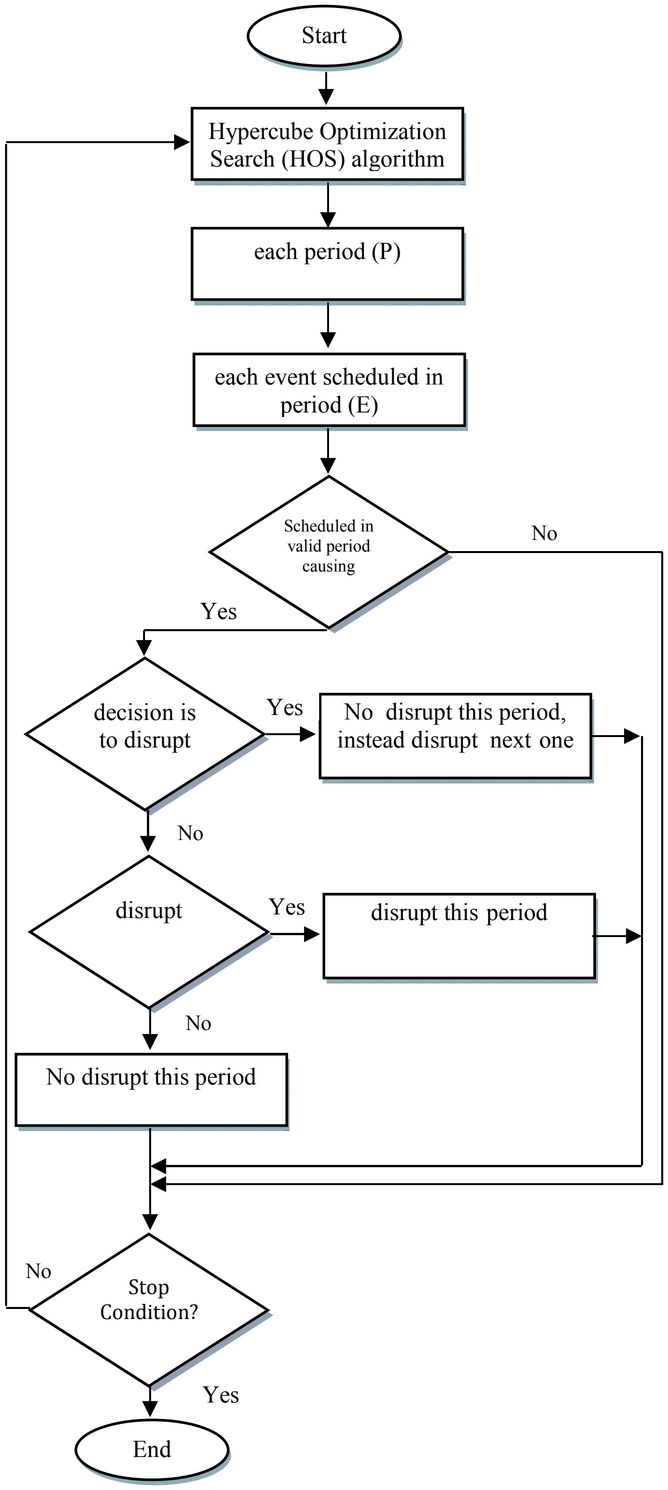

Fig. 3. The flowchart of HOS method

$$F_1(i) = \sum_{j=i+1}^{N} c_{ij} \cdot clash(t_i, t_j) \tag{2}$$

$$clash(t_i, t_j) = 1000 \quad if\ t_i = t_j \qquad 0 \quad otherwise \tag{3}$$

$$F_2(i) = \sum_{j=i+1}^{N} c_{ij} \cdot prox(t_i, t_j) \tag{4}$$

$$prox(t_i, t_j) = 32 / 2^{|t_i - t_j|} if\ 1 \leq |t_i = t_j| \leq 5 \qquad 0 \quad otherwise \tag{5}$$

The input variables for timetabling problem are stated as follows:

- exams (N),
- timeslots (P),
- students (M),
- t_i, $i \in \{1,...,N\}$ is the timeslot of exam i.
- c_{ij}, $i,j \in \{1,...,N\}$ is the number of students taking both exams i and j where $c_{ij} = 0$ if $i = j$.

Here, $\boldsymbol{F}$ is cost (or penalty) function describing the mean value of violation for each student. The function F_1 presents the cost related to hard constraint violation for exam i. F_1 is estimated by *clash* penalty equal to 10^3 multiplied by the number of students having a conflict. Here if the student will have clash then the value *clash* penalty will be 10^3, otherwise 0. The function of F_2 presents the cost related to soft constraint violation for exam i. F_2 is estimated by *proximity* penalty multiplied by the number of students having consecutive exams i and j. The value of *proximity* penalty is measured by $| t_i - t_j |$. For example, if a student will have two consecutive exams, then a proximity value of $(2^5 / 2^1) = 2^4$ will be assigned. If a student will have two exams with a free timeslot in between then a value of $(2^5 / 2^2) = 2^3$ will be assigned, etc. Depends on a number of timeslots (from 0 to 5) between the exams, the proximity penalty is decreased logarithmically from 16 to 1.

IV. Data Specification

There are many studies for searches and journals on finding new methods and improvements about known EAs used in optimization. However, the (HOS) method which references the best elements or solutions from an evaluation functions to solve applied examination scheduling problems that is designed. The experiments are carried out on an actual datasets (exam-data) from University of Nottingham, semester 1 and "toronto.tar" file of archive of datasets for various institutions. This file is taken from *ftp://ftp.cs.nott.ac.uk/ttp/Data/* for modeling of timetabling problem. The following information about the name of universities-institutions like as below and the information of datasets represented in Table I and is used in [5, 19-20, 23, 25-28].

University of Carleton (1991) **(carf91)** in Ottawa
University of Carleton (1992) (**carf92**) in Ottawa
Earl Haig Collegiate Institute (**ear83**) in Toronto
Montreal School of Hautes Etudes Commercials (**hec92**)
University of King Fahd (1993) (**kfu93**) in Dhahran
London School of Economics (**lse91**)
University of Ryerson (**rye93**) in Toronto
High School of St. Andrew's Junior (**sta83**) in Toronto
University of Trent (**tre92**) in Peterborough
University of Toronto, Faculty of Arts and Science (**ute92**)
York Mills Collegiate Institute (**yor83**) in Toronto

TABLE I. DATASETS FROM UNIVERSITIES

Dataset	Periods	Exams	Students	Density
carf91	35	682	16 925	0.13
carf92	32	543	18 419	0.14
ear83	24	190	1 125	0.27
hec92	18	81	2 823	0.42
kfu93	20	461	5 349	0.06
lse91	18	381	2 726	0.06
rye93	23	486	11 483	0.08
sta83	13	139	611	0.14
tre92	23	261	4 360	0.18
ute92	35	622	21 267	0.13
yor83	21	181	941	0.29

V. EXPERIMENTAL RESULTS

The performance of the proposed algorithm was implemented by using Matlab-Simulink (R2012a) and the specifications of used hardware and software tools are stated below;

1. Processor (CPU): Intel (R) Core (TM) i3-3110 M
2. Speed: 2.40 GHz – 64 Bit of System Type.
3. Memory: 4.00 GB (usable 3.89)
4. Operating System: Microsoft Windows 8.1

The performance of the HOS method tested on all above datasets with sizes subset of 50. We have summarized the best penalty, the worst penalty and the mean of the penalty results obtained by the HOS method in results of evaluations of the data sets over successful five independent runs. The maximum numbers of function evaluations (FEs) are equal to 10^5. The details of the experimental results of the datasets are shown in Table II for the proposed method.

TABLE II. EXPERIMENTAL RESULTS

Datasets	Experimental Results For HOS Method		
	Best	**Worst**	**Mean**
carf91	4.24	4.52	4.39
carf92	3.68	4.15	3.92
ear83	30.27	35.25	32.56
hec92	9.69	11.24	10.16
kfu93	12.81	13.29	12.99
lse91	6.64	8.95	7.60
rye93	5.13	7.56	5.79
sta83	156.9	160.8	157.97
tre92	6.68	8.59	7.29
ute92	22.19	24.25	23.00
yor83	34.66	40.96	36.29

The performance of experiments conducted in this section was made to distribute student exams during timeslots (exam period). Termination is initially set at 10^5 maximum number of function evaluations (FEs). The performance of methods compared using only one criterion: the minimum value of the cost function obtained in all tests (best). The details regarding of the comparative results of the datasets are shown in Table III. In order to compare the dataset under the same initial conditions, we defined the number of function evaluations (FE). All the proposed methods were run five times and with a maximum number of predefined function evaluations, considered as a single stopping criterion. The performance of the HOS method compared with other performance of methods; used a hybrid combination of approaches and particularly used a hybrid combination of approaches. The performance of the proposed method compared with the following methods that are given in references.

Method 1 (M1)—Merlot et al. [9],
Method 2 (M2)—Cote, Wong and Sabouri [14],
Method 3 (M3)—Burke et al. [15].
Method 4 (M4)—Abdullah et al. [16],
Method 5 (M5)—Tunay [31].

TABLE III. COMPARISON DATASETS WITH THE PROPOSED METHOD

Datasets	M1	M2	M3	M4	M5
carf91	5.1	5.2	4.6	5.2	**4.39**
carf92	4.3	4.2	**3.9**	4.4	3.92
ear83	35.1	34.2	32.8	34.9	**32.56**
hec92	10.6	10.2	**10.0**	10.3	10.16
kfu93	13.5	14.2	13.0	13.5	**12.99**
lse91	10.5	11.2	10.0	10.2	**7.60**
rye93	-	8.8	-	8.7	**5.79**
sta83	157.3	**157.2**	156.9	159.2	157.97
tre92	8.4	8.2	7.9	8.4	**7.29**
ute92	25.1	25.2	24.8	26.0	**23.00**
yor83	37.4	36.2	**34.9**	36.2	36.29

In Table III, we can see that our method algorithm (**M5**) compared with seven best results for the set of datasets applying universities namely; carf91, ear83, kfu93, lse91, rye93, tre92 and ute92 (be marked in boldface) as 4.39, 32.56, 12.99, 7.60, 5.79, 7.29 and 23.00 correspondingly. **M2** is best result for sta83 dataset as 157.2 and **M3** produces three best results for the set of datasets namely; carf92, hec92 and yor83 as 3.9, 10.00, 34.9 correspondingly. The focus of the comparative results have shown when HOS method has been started on already much better solutions than others for solving a set of optimal results. The finding of a near optimal solution of timetabling is important. Many timetabling processes are a little more complicated when room allocation (maximum sitting students per periods) is a part of the problem.

VI. CONCLUSION

The HOS method tested on all datasets from University of Nottingham, semester 1 and "toronto.tar" file of archive of datasets for various institutions.

The performance of the HOS method has demonstrated much better solution with convergence experimental results for datasets. Moreover, the performance of the HOS method compared with the performances of other methods by using examination timetable problems (datasets). The proposed method is much better in 7 out of 11 datasets; namely, carf91, ear83, kfu93, lse91, rye93, tre92 and ute92. Thus, the performance of the proposed method has shown a good performance than the performance of other methods.

In the future, the proposed algorithm will greatly contribute to the optimization of analysis results in many areas as computer science, economy, industry and many fields.

ACKNOWLEDGMENT

The author would like to thank the Faculty of Engineering and Architecture, the Department of Computer Engineering, *Istanbul Gelisim University*, Turkey.

REFERENCES

[1] Davison, V.E. & Sullivan, E. G. (1963). Mourning doves' selection of foods. J. Wildl. Manage. 27:373-383.

[2] Balakrishnan, N. (1991). Examination scheduling: a computerized application. The International Journal of Management Science (Omega) 1991;19(1):37–41.

[3] Carter, M. W., Laporte, G. & Lee, S. Y. (1996). Examination timetabling: algorithmic strategies and applications, Journal of the Operational Research Society 47 (1996) 373–383.

[4] Qu, R., Burke, E. K. & McCollum, B. (2009). Adaptive automated construction of hybrid heuristics for exam timetabling and graph coloring problems. European Journal of Operational Research 2009;198(2):392–404.

[5] Burke, E.K., Elliman, D. G., Ford, P. H. & Weare, R. F. (1996). Examination timetabling in British universities: a survey, in: E.K. Burke, P. Ross (Eds.), Practice and Theory of Automated Timetabling: Selected Papers from the First International Conference. Vol. 1153 of Lecture Notes in Computer Science, Springer-Verlag, Berlin, Heidelberg, pp. 76–90.

[6] Qu, R., Burke, E. K., McCollum, B. & Merlot, L. T. G. (2009). A survey of search methodologies and automated system development for examination timetabling, Journal of Scheduling, 12 (1) 55–89.

[7] Ahandani, M.A., Baghmisheh, M. T. V., Zadeh, M.A.B. & Ghaemi, S. (2012). Hybrid particle swarm optimization transplanted into a hyper-heuristic structure for solving examination timetabling problem. Swarm and Evolutionary Computation 7, 21–34.

[8] Schaerf A. (1999). A survey of automated timetabling, Artificial Intelligence Review 13 (2), 87–127.

[9] Merlot, L. T. G., Boland, N., Hughes, B. D. & Stuckey, P. J. (2003). A hybrid algorithm for the examination timetabling problem. In: Burke, E. K., De Causmaecker P, editors. Lecture notes in computer science: vol. 2740. Practice and theory of automated timetabling, IV: selected papers from the 4th international conference Berlin: Springer; p. 207–31.

[10] Muller, T. (2009). ITC2007 solver description: a hybrid approach. Annals of Operation Research; 172(1):429–46.

[11] Di Gaspero, L. and Schaerf, A. (2001). Tabu search techniques for examination timetabling. In: Burke E. K, Erben W, editors. Lecture notes in computerscience: vol.2079. Practice and theory of automated timetabling, III: selected papers from the 3rd internationalconference Berlin:Springer. p.104–17.

[12] Duong, T. A. and Lam, K. H. (2004). Combining constraint programming and simulated annealing on university exam timetabling. In: Proceedings of the 2nd international conference in computer sciences, research, innovation & vision for the future (RIVF2004), Hanoi, Vietnam, 2–5 February 2004. p. 205–10.

[13] Thompson, J.M. & Dowsland, K.A. (1998). A robust simulated annealing based examina- tion timetabling system. Comput Oper Res; 25(7–8):637–48.

[14] Cote P., Wong, T. and Sabouri, R. (2004). Application of a hybrid multi objective evolutionary algorithm to the uncapacitatedex amproximity problem. In: Burke EK, Trick M, editors. Lecturenotesin computer science: vol.3616. Practice and theory of automated timetabling,V:selected papers from the 5th international conference Berlin: Springer. p.151–68.

[15] Burke, E. K., Eckersley, A. J., McCollum, B., Petrovic, S. and Qu, R. (2010). Hybrid variable neighbourhood approaches to university exam timetabling. European Journal of Operation Research. 206(1):46–53.

[16] Abdullah, S., Ahmadi, S., Burke, E. K. & Dror, M. (2007). Investigating Ahuja-Orlin's large neighbourhood search approach for examination timetabling. OR Spectrum 2007;29(2):351–72.

[17] Mc Collum, B., McMullan P.J., Parkes, A.J., Burke, E.K. & Abdullah, S. (2009). An extended great deluge approach to the examination timetabling problem. In: MISTA09, The 4th multidisciplinary international conference on scheduling: theory and applications. Dublin, Ireland, p. 424–34.

[18] Burke E. K., Bykov, Y., Newall, J.P., Petrovic, S. (2004). A time predefined local search approach to exam timetabling problems. IIE Transactions 2004;36(6):509–28.

[19] Dowsland, K. A. & Thompson, J. (2005). Ant colony optimization for the examination scheduling problem. Journal of Operational Research Society; 56:426–38.

[20] Eley, M. (2007). Ant algorithms for the exam timetabling problem, in: Burke, E. K., H. Rudova (Eds.), PATAT 2007. Lecture Notes in Computer Science, vol. 3867, Springer, Heidelberg, pp. 364–382.

[21] Sabar, N. R., Ayob, M. & Kendall, G. (2009). Solving examination timetabling problems using honey-bee mating optimization (ETP-HBMO). In: Proceedings of the 4th Multidisciplinary International Scheduling Conference: Theory andAp- plications (MISTA 2009), Dublin, Ireland, pp. 399–408.

[22] Fong, C. W., Asmuni, H., McCollum, B., McMullan, P. and Omatu, S. (2014). A new hybrid imperialist swarm-based optimization algorithm for university timetabling problems. Information Sciences 283, 1–21.

[23] Alzaqebah, M. & Salwani, A. (2011). Comparison on the selection strategies in the artificial bee colony algorithm for examination timetabling problems, Inter- national Journal of Soft Computing and Engineering 1, 158–163.

[24] Abiyev, R. & Tunay, M. (2015). Optimization of High Dimensional Functions through Hypercube Evaluation. Computational Intelligence and Neuroscience, Volume 2015, 2015.

[25] Burke, E. K. & Newall, J.P. (2003). Enhancing timetable solutions with local search methods. In: Burke E. K., De Causmaecker P., editors. Lecture notes in computer science: vol. 2740. Practice and theory of automated timetabling, IV: selected papers from the 4th international conference Berlin: Springer; 2003. p. 195–206.

[26] Abdullah, S., & Turabieh, H. (2012). On the use of multi neighbourhood structures within a Tabu-based memetic approach to university timetabling problems. Elsevier 2012; 191,146–168.

[27] Rajah, C. and Pillay, N. (2019). "A Structure-Based Partial Solution Search for the Examination Timetabling Problem," *2019 IEEE Congress on Evolutionary Computation (CEC)*, Wellington, New Zealand, 2019, pp. 81-86.

[28] Mauritsius, T., Legowo N. and Gunawan, F. E. (2018). "Reducing the Timeslot Used in Examination Timetable Problem," *2018 International Conference on Information Management and Technology (ICIMTech)*, Jakarta, 2018, pp. 211-216.

[29] Psarra, E. and Apostolou, D. (2019). "Timetable Scheduling Using a Hybrid Particle Swarm Optimization with Local Search Approach," *2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA)*, PATRAS, Greece, 2019, pp. 1-8.

[30] Unprasertporn, T. and Lohpetch, D. (2020). "An Outperforming Hybrid Discrete Particle Swarm Optimization for Solving the Timetabling Problem," *2020 12th International Conference on Knowledge and Smart Technology (KST)*, Pattaya, Chonburi, Thailand, 2020, pp. 18-23.

[31] Tunay, M. (2020). Evolutionary Search Algorithm Based on Hypercube Optimization For High-Dimensional Functions. International Journal of Computational and Experimental Science and Engineering (IJCESEN), 6 (1), pp. 42-62.