UCTP based on Hybrid PSO with Tabu Search Algorithm using Mosul University Dataset

Aseel Ismael Ali
Computer Science Department, Faculty of Computer Science and Mathematics,
Mosul University, Iraq.

Ruba Talal
Computer Science Department, Faculty of Computer Science and Mathematics,
Mosul University, Iraq.

ABSTRACT
One of the most well-known constraint problem is a university course timetabling problem (UCTP), which become more difficult and more complex specially when we have a course with more than one teacher or a teacher with more than one course. This constraint problem take a lot of times to construct. In this paper we solve hard and soft constraint and take less time than usual by used one of artificial techniques, branch of swarm intelligent (SI), called particle swarm optimization (PSO). After a lot of researches we find that UTP can be solved with less time and more efficient based on PSO than other artificial intelligent (AI), or SI. In this paper, some improvements has been added to the algorithm to fit with the existing parameters in a dataset Mosul University College of Computer Science and Mathematics.

General Terms
UCTP, PSO, Tabu Search Algorithm.

Keywords
Timetabling, hard constraints, PSO, NP-complete.

1. INTRODUCTION
University course timetabling problem (UCTP) is usually defined as assigning a set of events (subjects) to a number of rooms (resources) and timeslots (periods) such that a number of constraints are satisfy. The course timetable problem is to organize a set of subject into a number of room and timeslot in order to satisfy a set of constraints [1]. Generally, it means "is the problem of allocation the resources within a specific time to accomplish a range of tasks" [2]. Many researches confirmed the difficulty in resolving scheduling problems by using the development traditional techniques [3] which has been studied extensively by different researchers because of their use in many areas, such as school, examination, or course timetabling, employee timetabling ,transport timetabling , and nurse rostering etc.

The efficiency of University course timetabling problem (UCTP) is important for teachers and students, it is also necessary to exploit the human and material resources in better form. In the past, resolve the scheduling (UCTP) manually by using trial and error, and it was difficult to resolve the optimal free access of errors, even if any of this solution, it is not the best, this has been resorting to scientific methods to resolve this problem. The scheduling in schools is easier than in the universities, because the room system (grades) in school is constant for each group of students while it is variable and will be chosen at random for each subject in the university. Increases the complexity of the problem at the university Whenever increased the complexity of the rooms system, so is the traditional way to resolve the problem is very difficult, particularly if there are several objectives to be achieved at the same time [5]. The problem is being studied for last more than four decades, but a general solution technique for it is yet to be formulated. The problem was first studied by Gotlieb ,who formulated a class-teacher timetabling problem by considering that each lecture contained one group of students ,one teacher, and any number of times which could be chosen freely[5].

So the problem is defined as multiple goals, there must be more than one goal determines the problem, the definition of objectives varies from one university to another. It is noteworthy that most researchers had to address the problem as a question and a single goal, where the only goal is to reduce all constraints (Abramson and Abela 1992; Blum et al. 2002; Lima et al. 2001; Piola 1994), but there are many options to define goals such as access to a solid and tight schedule, and the number of successive lectures for teacher that makes scheduling multiple goals.

There are a small number of researchers who have defined the problem as a matter of multiple goals. Carrasco and Pato (2001) used the dual-objective model scheduling for schools to reduce the soft constrains for both symmetrical sides the teacher and class, Filho and Lorena (2001) also define the problem as a dual goal. Dicev (2004) was known as a dual problem, but the definition of two different goals such as reduce spare time, and the expiry date of the daily quota, so it is given the priority to early stakes in one of the goals, and is given priority to quotas late goal in the other goal [6]. In 2001, Geem et al. developed a new metaheuristic algorithm called (Harmony search) algorithm, which mimics the musical improvisation process in which a group of musicians play the pitches of their musical instruments together seeking a pleasing harmony as determined by an audio-aesthetic standard. It is considered a population-based algorithm with local search based aspects . The idea of the algorithm lies in its ability to combine the major components of the community to achieve a simple optimization model. Harmony Search algorithm was applied to achieve the idea of university course scheduling. The results showed that this algorithm is able to achieve successful solutions compared to previous work. [7]

Other research has been used graph theory, which is characterized by smooth and more understanding of the researchers, especially non-specialists in operations research, particularly working in the field of management who are interested in the problem of timetables, as if this method is characterized by the presence of the Legends of the components of the matter which is the charts that make the researcher feels that he is able to control components matter more than any other method. [8]

2. CONSTRAINTS

In the final solution takes into account a set of parameters in order to be acceptable solution for both the student and the teacher, and is divided into the difficult constraints which value are (infinity) and soft Constraints which valued according to their importance. For hard constraints are as follows:

- Do not have a teacher lectures in the same time period.
- Do not have a student lectures in the same time period.
- Do not have a teacher lecture at times that not present in the university.
- Do not be in the hall two lectures in the same period of time and be resized fit the number of students.

The soft constraints are as follows:

- Do not students have many consecutive lectures without time intervals of rest.
- Do not have a teacher or student free time exceeds three hours.
- Do not have a teacher lectures a day late.
- Do not have any group of students consecutive lectures in distant places.

3. PARTICLE SWARM OPTIMIZATION (PSO)

PSO, intelligent swarms, taken from the behavior of flocks in the search for food in open space [11]. Each bird begins to search for food in the random areas and then change his position by following the best value obtained in the each cycle search (local best) [12]. In the first cycle of research, the first position is best value that obtained But in the second cycle, PSO start updating the positions and compare the rest of the bird positions with each other and then choose the best position among them (global best) [13]. So the update divided into two category [12]:

1. Local Update.
2. Global Update.

Each bird is considered a potential solution to the problem. Birds' velocity depends on his experience and the experience of its neighbors. PSO finds a better solution in less evolution than the rest of the algorithms. Each bird in the swarm has the following characteristics:

- xi: is the current position of the bird.
- vi: the current velocity of the bird.
- i: best position taken by a bird.

Suppose that (f) represents the fitness function that measure how close the solution to the optimal solution and (t) is the current time, the best position taken by the bird is updated as follows[16]:

$$Pbest_i\,(t+1) = \begin{cases} Pbest_i\,(t) & \text{if } f\big(x_i\,(t+1)\big) \geq f\big(Pbest_i\,(t)\big) \\ x_i(t+1) & \text{if } f\big(x_i\,\,(t+1)\big) < f\big(Pbest_i(t)\big) \end{cases} \quad (1)$$

The best position in the swarm at time t is calculated as in the equation(2)[16]:

$$Gbest(\mathrm{t}) \in \{Pbest_o(t), \ldots, Pbest_{n_s}(t)\} | f(Gbest(\mathrm{t})) = min\left\{f\big(Pbest_o(t)\big), \ldots, f\left(Pbest_{n_s}(t)\right)\right\} \quad (2)$$

Ns: represents the number of elements in the swarm.

The velocity of the element is calculated as in the equation(3)[16]:

$$V_{ij}\,\,(t+1) = W\,V_{ij}\,\,(t) + c1\,r1\,(t)\big[Pbest_{ij}\,\,(t) - X_{ij}\,\,(t)\big] + c2\,r2_j\,\,(t)\big[\,Gbest_j(t) - X_{ij}\,\,(t)\big] \quad (3)$$

Vij (t): represents the velocity of the element i in dimension j where (j = 1... nx) at time t.

i: represents the position of bird {i = 1,2, ... n}, where n represents the number of birds in the swarm.

j: {j = 1,2, ..., m} where m represents the dimensions of the problem.

w: represents the weight of inertia value of {0,1}

x_{ij} (t):represents the position of the item i in dimension j at time t.

c1,c2: represent constants.

r_{1j} ,r_{2j} : represent random values in the range (0,1) are taken from the uniform distribution, and these values add random to the algorithm.

P_{bestij}(t):Represent the best local position visited element since i first time.

finally the position of bird(i) is updated (x_i)by the following equation:

$$\mathrm{X}_i(\mathrm{t}+1) = \mathrm{X}_i\,(t) + \mathrm{V}_i\,(\,\mathrm{t}+1\,) \quad (4)$$

The previous steps are repeated until the maximum of the specific iterations to be realized within the algorithm or conditions stop-and-so is getting the best solution.

4. TABU SEARCH ALGORITHM (TS)

Its iterative optimization algorithm that similarity to the PSO algorithm. Its presented for the first time by Glover in 1986 [16]. Differ from PSO algorithm that does not fall into the wrong local because of its use of memory, which keeps the current value of the the target function and some information about the search for the optimal solutions. as well as the this algorithm used neighborhood method for the selection of scale solutions in each iteration, which will be examined If it's contained ineffective solution, this will be placed in the Target Function, If the solution is effective will be the value of the Fitness Function which is updated continuously to bear effective solution. As for the neighborhood method would be neighborhood of center of the best solution in the iteration subsequent and so the process will continue until they reach the measure stop.

The following are steps of Tabu Search Algorithm TS[18,19] (See figure1):

- Input: Tabu List size
- Output: Sbest
- Construct Initial Solution and put in Sbest
- Empty the Tabu List
- While (Stop Condition())
- Empty Candidate List
- For (Candidate swarm є Sbest neighborhood)
- If any of candidate swarm is not in the Tabu List
- Put Candidate swarm in the Candidate List
- Endif
- Endwhile
- Generate Candidate List
- Find best answer in the swarm and get it as new Candi-date
- If value of Candidate is better than the best answer so far
- put Candidate as Sbest
- Put the Candidate List in the Tabu List
- While Tabu List is greater than Tabu list size
- Delete extra feature in the Tabu List
- End while
- Endif
- End
- Return Sbest

Fig 1: show the steps of Tabu Search Algorithm (TS)

5. MODIFIED VELOCITY OF PSO ALGORITHM AND COMBINED WITH THE TABU SEARCH ALGORITHM (TS)

The PSO algorithm has some of the factors that cause variation and difference in the solution. One of the factors is the velocity, If the value is too large could lead to a move away solutions for optimal, while the other factors affecting the final solution of the algorithm are constant weight (w) and constants coefficients accelerate (c1, c2). When choosing an appropriate values for these constants will lead to increase the speed of convergence of the optimal solution and the solution becomes more accurate.

Therefore, in this paper has been updated velocities in the PSO algorithm and combined with the Tabu search algorithm to resolve the problem of the scheduling of the university course.

Initially we'll adjust the values of velocities in the PSO algorithm:

Because of generating random values of constants accelerate (r1&r2) by random generating function(ran), the values of the constants will be either too large or too small. Experiments has shown in the case that the large values, will lead to excessive values (Pbest& Gbest) and possibly lead to a move away bird (solution) for the ideal solution. But in the case of the fact that the values of the constants is too small will lead to inadequate adjustment velocities and the lack of access the convergence of small values. So was the definition of the two values for the high-speed (Vmax) and the low-lying velocities (Vmin) and then to update velocities by (3) equations and another two equations [20]:

$$V_{ij}\ (t+1) = \frac{V_{ij}\ (t+1).V_{i\,max}}{/V_{ij}\ (t+1)/},\ if\left|V_{ij}(t+1)\right| > V_{i\,max} \quad (5)$$

$$V_{ij}\ (t+1) = \frac{V_{ij}\ (t+1).V_{i\,min}}{/V_{ij}\ (t+1)/}, if\ |V_{ij}(t+1)\,| < V_{i\,min} \quad (6)$$

After updating the velocity of PSO algorithm ,we will insert TS algorithm by choosing constants values for coefficients accelerate (r1 ,r2) , determine the values of previous velocities and determine the distance between Pbest and Gbest of the new velocity, as we proceed towards the values of fitness function, Whenever we need to reduce or increase the effect of the distance between the Pbest and Gbest, so TS algorithm was used as a stage in the PSO algorithm. Here are the steps of PSO algorithm with TS algorithm(See figure2):

- Initialize the particles
- Initialize the w, and c1 and c2
- Initialize the Tabu List
- For every iteration
- Calculate target function for every particles
- If the recent value of the particle is better than previous
- one update the and set it by this value
- Choose the best particle of swarm that has best fitness value as
- Origin the neighborhood of w , and
- For every solution of neighborhood
- If the solution is not Tabu calculate the fitness value
- Choose the best value
- Update the Tabu List
- Calculate the velocities by the equation
- Calculate the particles position by the equation
- End

Fig 2: show the steps of PSO algorithm with TS algorithm

6. SOLUTION TABLES BASED ON THE PSO ALGORITHM:

There are four main factors in timetable: the teaching staff, lectures, students, classrooms. These factors represent the bird position and all the birds represent a group solution. The goal of the algorithm is to find the best position for the bird (best table) [15]. Conditions set for the College of Computer Science and Mathematics of the University of Mosul are as follows: the search space is a matrix of one-dimensional 15-time unit (5 days a week every day 6 hours per two-hour lecture). Each lecture has a laboratory in the week, not only placed laboratories or lecture only day there must be diversification between lectures and labs per day. Also taken into account the number of days that teacher presence at the department.

7. THE MATHEMATICAL MODEL

For pages other than the first page, start at the top of the page, and continue in double-column format. The two columns on the last page should be as close to equal length as possible.

7.1 Initialization

Do not include headers, footers or page numbers in your submission. These will be added when the publications are assembled.

- Initialize variables:
  - Number of teachers = T, the number of rooms = C, the number of days = D, the number of hours per day = H, the number of lessons per class = L, the number of birds = P.
  - The unit of time ti = the number of days D * the number of hours H
  - The number of teaching hours per college teacher = total_hour [T]
  - The total number of teaching hours for each row = total_hour [C].
- Initialize fitness function:
  - Local fitness function for each individual Local_fitness [p] = largest number Default.
  - Number of days of non-existence of teacher = day_off [T].
  - Distribution the quotas of teachers on the actual days = Day_dispersion [T].
  - Timetable = X [P, C, Ti]
  - General fitness function:
  - Global_fitness = largest number Default
- Initialize the movement of birds:

  Determine the starting point (destination) for each bird through and give it a random class and the unity of time and then each bird put a teacher randomly in a room and a unit of time.

7.2 Calculate The Fitness Function:

- calculate the local fitness function by the following equation:

  Local_fitness = cost1+ cost2 + cost3 + cost4.

- But general fitness function are calculated by taking the highest value of the local fitness function after the end of each generation of birds.
- Hard Constraints:
  - Cost1:Teacher availability at that unit of time which Influenced by the following:
  - Parallel teaching: giving teacher two lectures at the same time.
  - unavailablity in that day Because of Day_off or index to another college.
  - Cost2: a (class busy) the room is not available at that moment in time.
- Soft Constraints:
  - Cost3: distribution lectures of a teacher along the actual days of the week.
  - Cost4: diversification between lectures and laboratories in the day to that stage.

7.3 The Movement of Birds and Generate the Generations:

After initialize the movement of birds in the previous step, each bird fill its table by giving time unit for a particular room, teacher chosen at random, and then calculates the fitness function for that table. After the completion of the various tables for all birds, adjust the function of general fitness. Each bird has a local fitness less than the minimum is not inherited qualities of the second generation (canceled) made up the next generation has the highest fitness and lower the number of the previous generation that we get to the stop condition (a certain number of birds and High Fitness General) (See figure3).

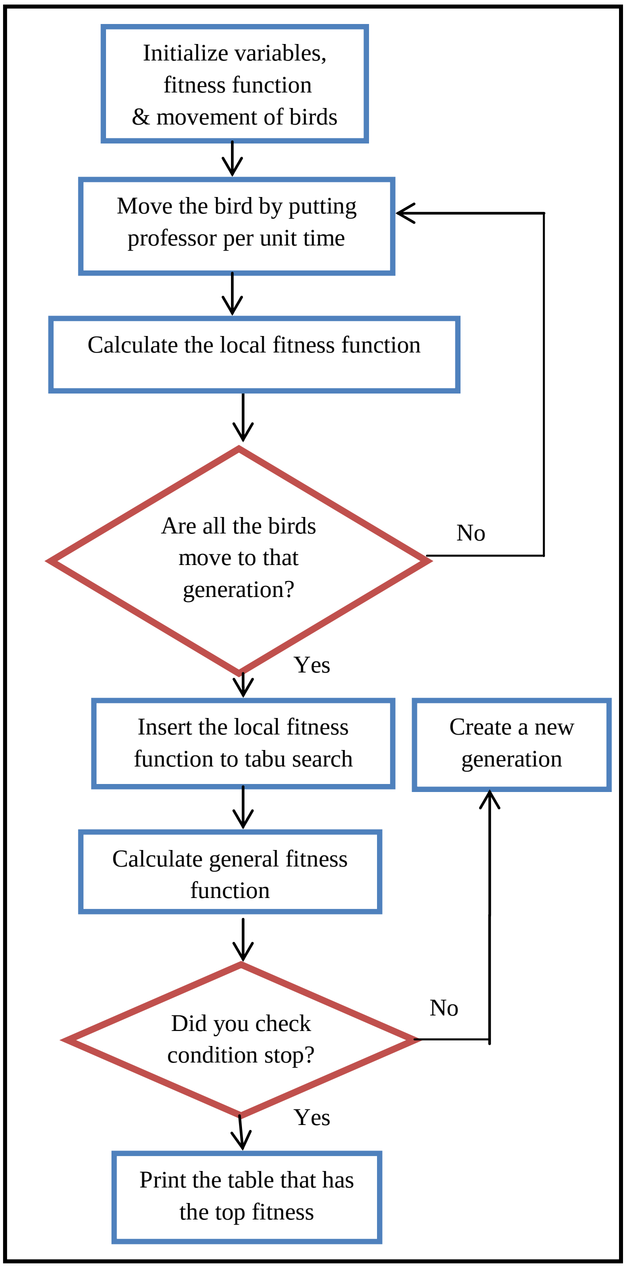


Fig 3: illustrates the algorithm of PSO based on UCTP

8. DATASET

In this paper we using dataset Department of Computer Science in the College of Computer Science and Mathematics at the University of Mosul, where it contains one teaching in professor degree (spotter 6 hours) and six teaching in Assistant Professor degree(spotter each one 8 hours) and twenty-four teaching in teacher degree (spotter each one 10 hours) and fourteen teaching in Assistant Lecturer degree (spotter each one 12 hours) and the total number of hours for each stage is the first stage : the first course and the second 15

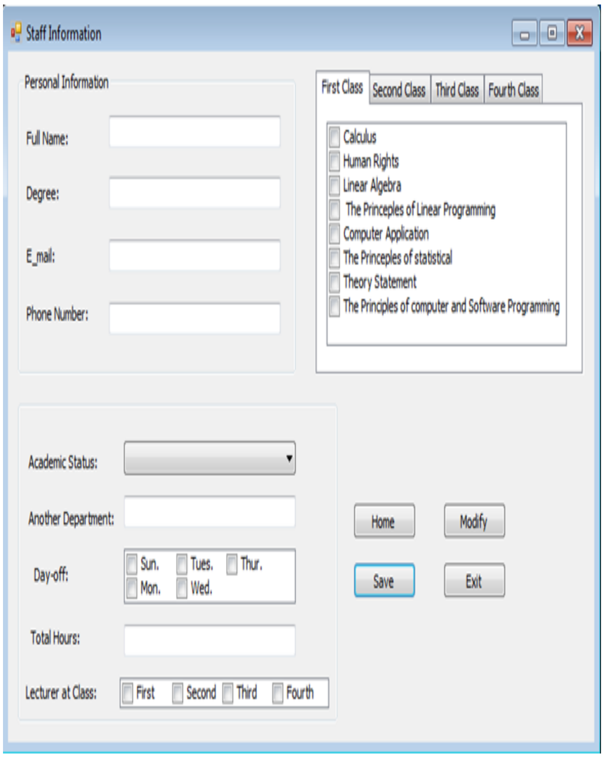


Figure 4 illustrates the process of data entry

hours of theoretical and 6 hours of a practical per week and second course 13hour theoretical and 8 hours of a practical per week. But the third stage - first course 16-hour theoretical and 10 hours of a practical per week and the second course 15 hours of theoretical and 12 hours a practical per week. But fourth stage - the first 11-hour course a theoretical and a practical 12 hours per week and 10 hours the second course theoretical and a practical 12-hour per week. A dataset programmed using the SQL server and Microsoft access database 2007.(See above Figure 4) illustrates the process of data entry.

9. THE STEPS OF PSO ALGORITHM BASED ON UCTP

The work steps university courses timetabling combined with PSO Algorithm are given in general, without giving details, so this paper give details about how to arrange the data to fit the PSO algorithm with the stated changes that have been proposed to improve the previous algorithms.

The algorithm begins the process of definition of initialization variables and give the values of the matrixes through a set of input data and initialize variables PSO algorithm and as shown in the following points:

9.1 Initialization

1. Get information from dataset and put it in several arrays as follow:
   a) For each class:
      I. Class_prof[c] = T
      II. Class_total_hour [c]= h
      III. Class_lesson [c] = L

b) For each prof:

I. Total_hour[T] = H

II. Day_off [T] = d

III. Lesson [T] = L

2. Initialize PSO:

a) Initialize fitness_function

I. For each particle

I.1 Local_fitness [p] = max_no

II. Global_fitness = max_no

b) Initialize particle (move particle)

For each particle:

I. P[c][ti] = T

II. Class_total_hour [c] = Class_total_hour [c] - 1

III. Total_hour[T] = Total_hour[T] – 2

9.2 Calculate Disruption of each prof

1. Disruption D = Lectures \Teaching_days
2. If D is not integer

a) X = A (integer of D) * Teaching_days

b) I = Lectures - X

c) For_loop from j to I + 1

i. For each day add 1

ii. j += Teaching_days

iii. end for

3. end if
4. end function

9.3 Main algorithm

Local_fitness = cost1 + cost2 + cost3 + cost4

1. for each prof assign to table

a) if Day_off [T] = 1 or assign to another class at the same time then cost1 = cost1 + 1;

2. for each unavailable class assign to table cost2++
3. for each not ideal disruption for prof assign to table cost3++.
4. for each not ideal disruption for class assign to table cost4++.

9.4 Calculate the optimal distribution of the stage through the function proposed the following:

1. maximum NO of lecture (Lec) to that day and class = 2 .
2. maximum NO of lab (Lab) to that day and class = 2
3. if timeslot not full.

a) if lec not egual 2

i. add lec .

b) else add lab

4. end if

9.5 The proposed function of the stop condition:

1. tolerance = ∑local_fitness[p] / p
2. eliminate all partical that have local_fitness < tolerance
3. if partical < 4 and global_fitness is the same for last three loop then stop condition is true.

10. EXPERIMENTS RESULTS

Applied the PSO algorithm based on UCTP on dataset professors in the Faculty of Computer Science and Mathematics at the University of Mosul, the implementation of the algorithm were based on (Visual Studio. Net 2008) through the language Visual C #.

The following figure shows the results of the implementation of the program (written symbols instead of the name of a professor because of the confidentiality of the data:

For each positive number, it is represent a lecture with a specific teacher. While the negative number represent the laboratory for that lesson.

Table 1 reveals information about the detail of dataset have been used. The number of teachers, classes, lectures and labs. The small dataset is the actual data used in Department of Computer Science in the College of Computer Science and Mathematics at the University of Mosul.

Table 1. Dataset has been used

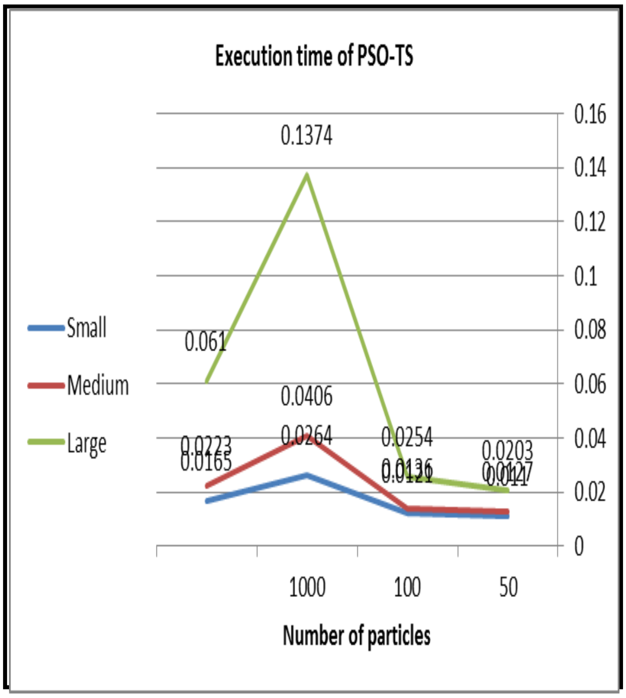
Category	Small	Medium	Large
No. of teachers	45	100	400
No. of classes	4	6	10
No. of Lectures	32	48	80
No. of Labs	24	36	60

Table 2 give information about execution time for proposed algorithm (PSO-TS) with several dataset and several number of particles. It can clearly be seen that average time for all dataset have below 1 second. With this result, it can give a new approach of using UCTP based on PSO.

Table 2. Execution time (sec) of proposed algorithm for each group of dataset

No. of particle	Small	Medium	Large
50	0.0110	0.0127	0.0203
100	0.0121	0.0136	0.0254
1000	0.0264	0.0406	0.1374
Average	0.0165	0.0223	0.0610

Graph 1. Execution time (sec) of proposed algorithm for each group of dataset



The algorithm will compare with two basic algorithms in this field which are standard PSO and PSO with local search (PSO-LS). Table 3 shows that PSO-TS algorithm has the best result in execution time with several dataset which the value bellow 1 millisecond while two algorithm have 18 sec with minimum execution time.

Table 3. Comparison studies PSO-TS with PSO-LS in execution time (sec)

Category	PSO-TS	PSO-LS	Standard PSO
Small	0.0165	18.00	18.40
Medium	0.0223	2.00	2.00
Large	0.0610	28.00	28.20

Graph 2. Comparison studies PSO-TS with PSO-LS in execution time (sec)

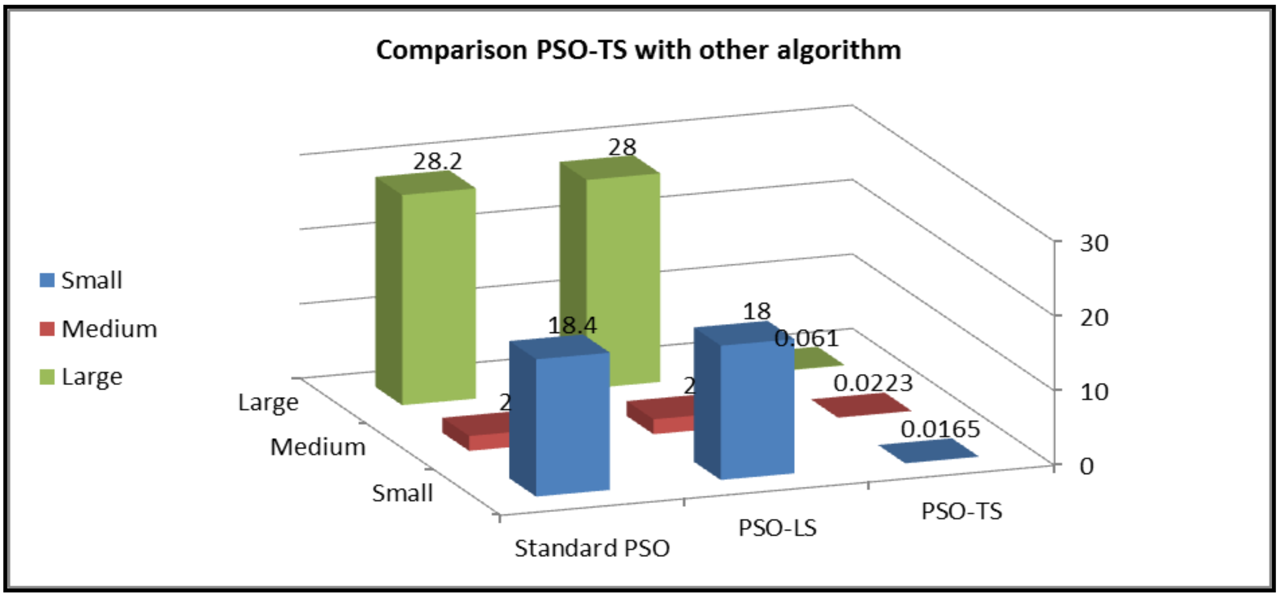


Table 4. Timetable of class A.

Hours/ Days	Sunday	Monday	Tuesday	Wednesday	Thursday
8:30-10:30	1	-4	3	6	-1
10:30-12:30	-8	7	-2	4	-3
12:30-2:30	2	5	8	-7	

11. CONCLUSION

This paper applied particle swarm optimization algorithm on a university course timetabling problem with tabu search added to it. the time taken in setting the table of quotas compared with existing algorithms have been reduced and the algorithm has also been improved to prevent falling into the end of the local optimum condition during the stop and through the integration of the algorithm with an algorithm Tabu search. Also a proposed functions meet the requirements of scheduling course university Department of Computer Science, including the function of diversification between lectures, laboratories, and a function to distribute lectures professor along the actual days, as taking into account whether a professor index to other college than under the terms of setting the table.

It is highly recommendable that future work should be directed:

1. To add a timetabling for each teacher.
2. To integrate PSO with other algorithm that determined the first position for each particle instead of random position.
3. To apply PSO for examination timetabling problem.
4. To convert dataset to database of Mosul university.

12. REFERENCES

[1] Safaai, D., Sigeru, O., Hiroshi, O., Puteh, S.: Incorporating Constraint Propagation in Genetic Algorithm for University Timetable Planning. Engineering Applications of Artificial Intelligence 12, pp. 231-253. (1999)

[2] Ho Sheau Fen @ Irene, Deris, Safaai, MohdHashim, SitiZaiton:Solving University Course Timetable Problem Using Hybrid Particle Swarm Optimization.Faculty of Comp. Sc. & Info. UniversitiTeknologiMalysia.81310. Johor, Malaysia.(2013)

[3] Elizabeth Montero, Mar´ıa-Cristina Riff, LeopoldoAltamirano:A PSO algorithm to solve a Real Course+ExamTimetabling Problem.Universit´e de Nice2000 route des Lucioles, 06903 Sophia AntipolisC´edex, France.(2011)

[4] Burke EK, McCollum B, Meisels A, Petrovic S, Qu R : Agraph-based hyper heuristic for educational timetabling problems.Eur J Oper Res 176:177–192 .(2007)

[5] D. Datta, K. Deb, and C. M. Fonseca, “Solving class timetabling problem of IIT kanpur using multi-objective evolutionary algorithm”, (2006).

[6] P. Pongcharoen, W. Promtetn, P. Yenradee, and C. Hicks, “Stochastic optimization timetabling tool for university course scheduling”, (2007).

[7] Mohammed Azmi Al-Betar , AhamadTajudinKhader:A harmony search algorithm for university course timetabling.Springer Science.(2012)

[8] Jensen , B.J. &Gutin , Gregory : Digraphs Theory ,Algorithms and Applications .(2008)

[9] J. Digalakis , K. Margaritis: Parallel evolutionary algorithms on mpi. University of Macedonia ,Department of Applied Informatics.(2003)

[10] Ruey-Maw Chen,Hsiao-Fang Shih:Solving University Course Timetabling Problems Using Constriction Particle Swarm Optimization with Local Search. Department of Computer Science and Information Engineering, National Chinyi University of Technology, Taichung, Taiwan.(2013).

[11] S. Kitayama, K. Yamazaki, M. Arakawa " adaptive range particle swarm optimization". Springer Science + Business Media, Journal: Optimization and Engineering ISSN: 13894420 Year: 2009 Volume: 10 Issue: 4 pages:575-597 DOI:10.1007/s11081-009-9081-7.

[12] D. Bratton, J. Kennedy "defining a standard for particle swarm optimization", IEEE, Journal: 2007 IEEE Swarm Intelligence Symposium ISBN: 1424407087 Year: 2007 Pages: 120-127 Provider: IEEE Publisher: IEEE DOI: 10.1109/SIS.2007.368035

[13] D. P. Rini, S. M. Shamsuddin, S. S. yuhaniz "Particle Swarm Optimization: Technique, System and Challenges" International Journal of Computer Applications (0975 – 8887) Volume 14– No.1, January 2011.

[14] D. Souravilas, K. parsopoulos "Particle Swarm Optimization with Budget Allocation through Neighborhood Ranking", GECCO'13, July 6–10, 2013, Amsterdam, The Netherlands. 2013 ACM 978-1-4503-1963-8/13/07.

[15] Shiau, D.F., "A hybrid particle swarm optimization for a university course scheduling problem with flexible preferences " Journal: Expert Systems With Applications ISSN: 09574174 Year: 2011 Volume: 38 Issue: 1 Pages: 235-248.

[16] Engelbrecht A. P., (2007):"Computational Intelligence An Introduction", Second Edition, John Wiley & Sons Ltd, West Sussex, England.

[17] F. Glover and C. McMillan . The general employee scheduling prob-lem: an integration of MS and AI. Computers and Operations Re-search,1986.

[18] F.Glover, M.Laguna, Tabu Search, Univ. Boulder, 1997.

[19] A.hertz, E.Taillard, D.De werra, A Tutorial on Tabu Search, OR Spektrum 11, 1989, 131-141.

[20] Z.Zainuddin, O.Pauline. Function Approximation Using Artificial Neural Network. WSEAS transaction on Mathematics, issue 6, Vol 7, June 2008.