PERFORMANCE EVALUATION OF PSO, PSOCA AND MPSOCA FOR SOLVING UNIVERSITY TIMETABLING PROBLEM

Oluwaseun M. Alade, Christopher A. Oyeleye, Oluyinka T. Adedeji, Elijah O. Omidiora, Stephen O. Olabiyisi

Department of Computer Science and Engineering, Ladoke Akintola University of Technology, Ogbomoso, Nigeria

Corresponding Author: O. T. Adedeji, otadedeji@lautech.edu.ng

***ABSTRACT*:** In this paper, performance evaluation of Particle Swarm Optimization algorithm (PSO), Particle Swarm Optimization based Cultural Algorithm (PSOCA) and Modified Particle Swarm Optimization based Cultural Algorithm (MPSOCA) was carried out using simulation time, fitness value and number of unallocated courses as performance metrics. The evaluation results of PSO, PSOCA and MPSOCA yielded average simulation times of 35.29, 37.68 and 17.42 seconds, respectively. Also, fitness values of 85, 89 and 90% were recorded for PSO, PSOCA and MPSOCA, respectively. PSO have a total average number of 60 subjects unallocated compare to PSOCA and MPSOCA that successfully allocated all the subjects.
KEYWORDS: Particle Swarm Optimization Algorithm, Cultural Algorithm, Timetabling.

1. INTRODUCTION

Timetabling is combinatorial optimization problems, which consist of scheduling a set of courses within a given number of rooms and time periods [AT08]. It takes longer time to solve a real life timetabling problem. The timetable generated using manual approaches requires several days which may also produce an inadequate results because of its constraints violation. Timetabling is the allocation of given resources to objects being placed in space time, subject to constraints, in such a way as to satisfy a set of necessary objectives as virtually as possible [BN02, AT08, O+12, A+16).
Cultural Algorithm (CA) is a technique that incorporates domain knowledge obtained during the evolutionary process so as to make the search procedure more effective [RZ01]. The goal is to increase the learning or convergence rates of the algorithm so as to provide a better response to a large number of problems [BM00].
Particle swarm optimization (PSO) is one of the evolutionary computational techniques and population-based search algorithms [Yuh04]. The characteristics of PSO method makes it very prevalent, it has memory which is vital to the algorithm. Also it is simple to implement, it has ability to swiftly converge to a good solution, as compared with other optimization methods; it is faster, cheaper and more effective. Also, there are a small number of factors to be adjusted in PSO. Compared with genetic algorithms (GAs), the information sharing mechanism in PSO is significantly different [ZJC04, Qin10].
Different authors had worked on timetable using different methods to tackle the problems associating with timetabling. [A+14] used Genetic Algorithm, [CK10] used hybrid of genetic algorithms and fuzzy logic, [SS11] used Hybrid of Genetic algorithms with Guided and Local Search Strategies, [O+12] used hybrid of genetic algorithms and simulated annealing, hybrid of genetic algorithms and fuzzy logic with randomized iterative local search, simulated annealing and tabu search was used by[MM12, IDM09] solved the problem using hybrid particle swarm optimization- constraint-based reasoning approach, [A+16] employed the use of modified Particle Swarm Optimization based Cultural Algorithm among others.

2. RESEARCH METHODOLOGY

In this paper, PSO, PSOCA AND MPSOCA were used for solving the timetabling problem. The cultural algorithm comprises of population space and belief space. In the population space of the cultural algorithm, Modified Particle swarm optimization (MPSO) was used in the MPSOCA while PSO was used in PSOCA.

Modification of Particle Swarm Optimization Algorithm

There is problem of slow convergence speed, falling into local extreme point and high software and computational complexity in standard particle swarm optimization (SPSO) which makes it difficult to achieve a good result. The modification of standard PSO is in terms of update formula of the standard PSO (equation 2.1). Update formula was adjusted in order to track the historical best particle of cultural algorithm stored in the belief space. To do this acceleration component $q_3 r_3 (R_{cd}^i - X_{id}^i)$ was incorporated into the cognitive and social component of the update formula of SPSO. Also, Influence factors (μ_1 and μ_2) were introduced into the equation to represent how population space and cultural framework guides particle's flight. Acceleration component measures the performance of the particles relative to global best position of the cultural algorithm, thus improving the convergence speed. How population space and cultural framework guides particle's flight were represented using influence factors. A strong local search capacity was maintained thus making the algorithm to have a fast convergence speed as well as improving the software and computational complexity of the system. In standard PSO, the update formulas of particle i are as follows:

$$V_{id}^{i+1} = \gamma \left(V_{id}^i + q_1 r_1 (R_{id}^i - X_{id}^i) + \right.$$
q2r2Rgdi−Xidi (2.1)

$$X_{id}^{i+1} = X_{id}^i + V_{id}^{i+1} \quad (2.2)$$

In Modified PSO, the update formula now becomes:

$$V_{id}^{i+1} = \gamma \left(V_{id}^i + q_1 r_1 (R_{id}^i - X_{id}^i) + \mu_1 q_2 r_2 (R_{gd}^i - \right.$$
Xidi+μ2q3r3Rcdi−Xidi (2.3)

$$\gamma = \frac{2}{\left|2-\varphi-\sqrt{\varphi^2-4\varphi}\right|} \quad (2.4)$$

$$\varphi = \varphi_1 + \varphi_2 + \varphi_3 \; \varphi_1 = c_1 r_1, \varphi_2 = c_2 r_2, \; \varphi_3 = c_3 r_3,$$

Where:
V_{id} is the velocity component of the ith particle in the dth dimension
X_{id} is the position component of the ith particle in the dth dimension
χ is constriction factor
φ is sum of learning factors
q_1, q_2 *and* q_3 are learning factors; (cognitive, social and acceleration)
r_1, $r_2$and r_3 are random numbers in [0, 1].
μ_1 and μ_2 are influence factor which represent respectively how population space and cultural framework guides particle's flight.
R_{id} is the individual historical best position of particle i in the dth dimension
R_{gd} is the historical best position component of the G_{best} in the dth dimension
R_{cd} is the historical best population of cultural framework

$q_1 r_1 (P_{id}^i - X_{id}^i)$ is a cognitive component which measures the performance of the particles i relative to past performance
$q_2 r_2 (P_{gd}^i - X_{id}^i)$ is a social component which measures the performance of the particles relative to a group of particles or neighbors?
$q_3 r_3 (P_{cd}^i - X_{id}^i)$ is an acceleration component which measure the performance of the particles relative to global best position of cultural algorithm stored in a belief space.
q_1, q_2 and q_3 together with r_1, r_2 and r_3 maintain the stochastic influence of cognitive, social and acceleration components of the particles velocity respectively.

Modified PSO Algorithm (MPSO)

The MPSO algorithm can be described as follows:
Step 1: Identify the number of particles that will be used to solve the problem.

$$f(P_k^i) \le f(P_k^{i-1}) \le \cdots \ldots \le f(P_k^1) \quad (2.5)$$

Step 2: Estimate the fitness value of each particle
Step 3: Set P_{best} to the current position if the fitness value of each particle's current position is better than its previous P_{best}
Step 4: Fitness value of the particle is compared with that of the G_{best}. If it is better, the $G_{best}$is updated
Step 5: Update the velocity and position of each particles using

$$V_{id}^{i+1} = \gamma \left(V_{id}^i + q_1 r_1 (R_{id}^i - X_{id}^i) + q_2 r_2 (R_{gd}^i - \right.$$
Xidi (2.6)

$$X_{id}^{i+1} = X_{id}^i + V_{id}^{i+1} \quad (2.7)$$

Step 6: Repeat the process from step 2 until the termination condition is satisfied.

Formulation of a Modified Particle Swarm Optimization based Cultural Algorithm (MPSOCA)

In formulating an MPSOCA, the modified particle swarm optimization algorithm (MPSO) was substituted into the population space of the cultural algorithm framework.
The formulated modified particle swarm optimization based cultural algorithm has the following steps
Step 1: Initialize the algorithm parameters of MPSOCA

Step 2: Initialize the particle in both population space and belief space

Step 3: Renew population space with MPSO algorithm, compute the fitness of each particle, update and store the individual best particle of the population space

Step 4: If accept condition is satisfied, carry on accept operation, send some better particles to belief space

Step 5: If belief space satisfy reset condition, reset the best particle of belief space

Step6: Renew belief space with update formula, compute fitness of each particle, update and store the best particle of the belief space.

Step 7: If influence condition is satisfied, carry on influence operation, and substitute some better particles of belief space for some worst particle of population space.

Step 8: Check whether the stop condition is satisfy. If the stopping condition is not satisfied then go to step 3. Otherwise stop and obtain the best solution from the global best position

Formulation of a Particle Swarm Optimization based Cultural Algorithm (PSOCA)

In formulating a PSOCA, particle swarm optimization algorithm (PSO) was substituted into the population space of the cultural algorithm framework.
The formulated particle swarm optimization based cultural algorithm has the following steps:

Step 1: Initialize the algorithm parameters of PSOCA

Step 2: Initialize the particle in both population space and belief space

Step 3: Renew population space with PSO algorithm, compute the fitness of each particle, update and store the individual best particle of the population space

Step 4: If accept condition is satisfied, carry on accept operation, send some better particles to belief space

Step 5: If belief space satisfy reset condition, reset the best particle of belief space

Step 6: Renew belief space with update formula, compute fitness of each particle, update and store the best particle of the belief space.

Step 7: If influence condition is satisfied, carry on influence operation, and substitute some better particles of belief space for some worst particle of population space.

Step 8: Check whether the stop condition is satisfy. If the stopping condition is not satisfied then go to step 3. Otherwise stop and obtain the best solution from the global best position

PSO Algorithm (PSO)

The PSO algorithm can be described as follows:

Step 1: Identify the number of particles that will be used to solve the problem.

$$f(P_k^i) \leq f(P_k^{i-1}) \leq \cdots \leq f(P_k^1) \qquad (2.8)$$

Step 2: Estimate the fitness value of each particle

Step 3: Set P_{best} to the current position if the fitness value of each particle's current position is better than its previous P_{best}

Step 4: Fitness value of the particle is compared with that of the G_{best}. If it is better, the G_{best} is updated

Step 5: Update the velocity and position of each particles using

$$V_{id}^{i+1} = \gamma\left(V_{id}^i + q_1 r_1 (R_{id}^i - X_{id}^i) + q_2 r_2 (R_{gd}^i - Xidi\right. \qquad (2.9)$$

$$X_{id}^{i+1} = X_{id}^i + V_{id}^{i+1} \qquad (2.10)$$

Step 6: Repeat the process from step 2 until the termination condition is satisfied.

Mathematical Representation of the Problem

The following important parameters are defined as follows:
$E = \{1..e\}$ of events, each of which contains certain students and needs certain features
$R = \{1..r\}$ of rooms, each of which has a seat capacity and its own features.
$S = \{1..s\}$ of students, each of whom enrolls in some events
$F = \{1..f\}$ of features, such as overhead projectors or special whiteboards
$P = \{1..p\}$ of timeslots where p = 40 (5 days with 8 periods on each day)
$D = \{D_1..D_5\}$ of days where each day has 8 periods
Ordered subsets P^d of P corresponding to a period in a day d where
$P^1 = \{p_1, p_2 \ldots..p_8\}, P^2 = \{p_9, p_{10} \ldots..., p_{16}\}\ldots.$
An ordered subset $L^d = \{p_8, p_{16}, p_{24}, p_{32}, p_{40}\}$ that contains the last periods of each day.
$L^d \varepsilon P, d \varepsilon D^d$

e, r, s, f, p are the number of events, rooms, students, features and timeslots respectively

S_r^R = the size of room r, e ε E

S_e^E = the number of students enrolled in event e, e ε E

$$w_{f,e} = \begin{cases} 1 & \textit{if event e requires feature f} \\ 0 & \textit{otherwise } e\varepsilon E \textit{ and } f\varepsilon F \end{cases}$$

$$y_{f,r} = \begin{cases} 1 & \textit{if room r contains feature f} \\ 0 & \textit{otherwise } r\varepsilon R \textit{ and } f\varepsilon F \end{cases}$$

$$t_{s,e} = \begin{cases} 1 & \textit{if student s is enrolled in event e} \\ 0 & \textit{otherwise } s\varepsilon S \textit{ and } e\varepsilon E \end{cases}$$

Decision variables

x are binary decision variables indexed by events, rooms, and timeslots.

$$x_{e,r,p} = \begin{cases} 1, f & \textit{event e occured in room r, and time period p} \\ 0, & \textit{otherwise } e\varepsilon E, r\varepsilon R \textit{ and } p\varepsilon P \end{cases}$$

C_s^{ldp} (last period of day): Its value representing the number of violations of soft constraint S_1 by student s.

C_s^{3R} (More than three events in a row): Its value representing the number of violations of soft constraint S_2 by student s.

C_s^{id} (single class in a day): Its value representing the number of violations of soft constraint S_3 by student s.

C_s^{sr} (student and room ratio): Its value representing the number of violations of soft constraint S_4 by student s.

$z_{s,d}$ are binary decision variables indexed by student and day; their value indicates that student s has a single class in a day d. s ε S and d ε D^d

The objective function is given as follows:

Minimize

$$\sum_{s\varepsilon E}\left(C_s^{ldp} + C_s^{3R} + C_s^{id} + C_s^{sr}\right) \quad (2.11)$$

C_s^{ldp}, C_s^{3R}, C_s^{id} and C_s^{sr} consecutively describe the violations of the soft constraints S1, S2, S3 and S4 made against the will of each student. When each violation occurs in the solution, it will be penalized by 1. Soft constraints are described by Equations (2.12) to (2.17).

Subject to

$$\forall e\varepsilon E \quad \sum_{r\varepsilon R}\sum_{p\varepsilon P} x_{e,r,p} = 1 \quad (2.12)$$

$$\forall s\varepsilon S \quad C_s^{ldp} = \sum_{e\varepsilon E}\sum_{r\varepsilon R}\sum_{q\varepsilon L} t_{s,e}x_{e,r,q} \quad (2.13)$$

$$\forall s\varepsilon S \quad C_s^{3R} = \sum_{\substack{i,j,k\varepsilon E \\ i\neq j\neq k}}\sum_{r\varepsilon R}\sum_{\substack{p,q,m\varepsilon P \\ q=p+1 \\ m=q+1}} t_{s,i}t_{s,j}t_{s,k}x_{i,r,p}x_{j,r,p}x_{k,r,m} \quad (2.14)$$

$$\forall_{s\varepsilon S}\forall_{d\varepsilon D^d} \quad z_{s,d} = \begin{cases} 1 & \sum_{e\varepsilon E}\sum_{r\varepsilon R}\sum_{i\varepsilon P} t_{s,e}x_{e,r,i} = 1 \\ 0 & otherwise \end{cases} \quad (2.15)$$

$$\forall_{s\varepsilon S} \quad C_s^{ld} = \sum_{q\varepsilon D^d} z_{s,q} \quad (2.16)$$

$$\forall_{r\varepsilon R}\forall_{p\varepsilon P} \quad C_s^{sr} = \sum_{e\varepsilon E} S_e^E x_{e,r,p} \leq S_r^R \quad (2.17)$$

Equation (2.12) describes the implicit constraint which means that timetable solution must be complete and each event must be presented once. Equation (2.13) for (S1), equation (2.14) for (S2), equation (2.16) for (S3), equation (2.15) is necessary for describing (S3) which penalizes students who have only attended a single event in a day by 1, while equation (2.16) calculates all violations of any students for all days. Equation (2.17) for (S4). Also in equation (2.17) True represent 1 and false represent 0.

Hard and Soft Constraints

Hard constraints are the constraints that must be fulfilled, while soft constraints are the one to be fulfilled as much as possible. A feasible timetable is one in which all hard constraint are satisfied and nearly all soft constraint are satisfied too, while a non-feasible timetable is the one in which part of the hard constraint is not fulfilled even though all soft constraints are satisfied. In this research, the hard constraints under consideration are as follows:

H1: Lectures having students in common cannot take place at the same time

H2: Each classroom can only be used for one course in the same timeslot

H3: Lecturer cannot teach more than one course at a time

H4: No courses are to be conducted in the 13-14 hours and 15-17 hours each Friday and Wednesday as that slot are allotted for Muslim prayers and Sport respectively in LAUTECH

Concurrently, the following soft constraints were used:

S1: A student shall not have a class in the last slot of the day.

S2: A student shall not have more than three classes in a row

S3: A student shall not have a single class on one day

S4: The number of students that attend the course for each lecture, must be less than or equal to the number of seats of all the rooms that host its lectures.

3. EXPERIMENTAL RESULTS

The algorithm was tested with 858 courses and 135 venues; particle size is 20, with c_1 as 1.0, c_2 as 1.0, and c_3 = 2.4. The result for MPSO algorithm is shown in table 1 while table 2 showed the result for MPSOCA. The two algorithms were tested under 4 separated runs.

Table 1: Result after four runs of PSO Algorithm

Run	Time (sec)	Fitness value	Hard constraint violation	Soft constraint violation	Number of subjects unallocated
1	35.59	0.83	0	2	58
2	32.33	0.85	0	2	60
3	33.31	0.84	0	2	60
4	40.33	0.89	0	2	62
Average	35.39	0.85	0	2	60

Table 2: Result after 4 runs for MPSO

Run	Time (sec)	Fitness value	Hard constraint violation	Soft constraint violation	Number of subjects unallocated
1	22.04	0.88	0	0	0
2	21.12	0.88	0	0	0
3	23.12	0.86	0	0	0
4	20.03	0.90	0	0	0
Average	21.57	0.88	0	0	0

Table 3: Results after 4 runs for PSOCA

Run	Time (sec)	Fitness value	Hard constraint violation	Soft constraint violation	Number of subjects unallocated
1	35.57	0.90	0	0	0
2	37.02	0.87	0	0	0
3	39.23	0.92	0	0	0
4	38.90	0.89	0	0	0
Average	37.68	0.89	0	0	0

Table 4: Result after 4 runs for MPSOCA

Run	Time (sec)	Fitness value	Hard constraint violation	Soft constraint violation	Number of subjects unallocated
1	17.78	0.89	0	0	0
2	18.84	0.87	0	0	0
3	15.67	0.94	0	0	0
4	17.40	0.91	0	0	0
Average	17.42	0.90	0	0	0

From table 1, the standard PSO have a total average number of 60 subjects unallocated compare to MPSO that successfully allocated all the subjects. The main reason is that the standard PSO doesn't have the capability to handle constraint like MPSO. From table 2, MPSO successfully allocate all subjects, there is no violation of both hard and soft constraints. MPSO generated higher best fit values with a lower simulation time as compared to the standard particle swarm optimization algorithm. From this result, it can be deduced that MPSO improved the performance of the standard particle swarm optimization in terms of generating more optimized timetable. Acceleration component in MPSO helps in improving the convergence speed which in turns reduced the average time taken to generate a timetable. MPSO have a fitness value of 0.88 while PSO have a fitness value of 0.85. From table 3, there is no violation of both hard and soft constraint, no unallocated subjects was recorded. An average fitness value of 0.89 at an average time of 37.68seconds was recorded.

From table 4, there is no violation of both hard and soft constraint. MPSOCA successfully allocate all the subjects as a result of their capability to handle constraint. As a result of belief space of cultural algorithm, MPSOCA generated an optimal timetable within an average simulation time of 17.42 seconds with an average fitness value of 90% as compared with PSO, MPSO and PSOCA which had an average simulation time of 35.39, 21.57 and 37.68 seconds respectively with a fitness value of 85%, 88% and 89% respectively. From this result, it can be deduced that the modified particle swarm optimization based cultural algorithm improves the performance of the timetable generated both in terms of simulation time and fitness value.

4. CONCLUSION AND FUTURE WORK

In this paper, we have been able to evaluate the performance of PSO, PSOCA and MPSOCA for solving lecture timetabling problems in universities or other related higher institutions. MPSOCA developed consisting of both influence factors and acceleration component which are highly probable in minimizing high convergence speed associated with the standard Particle Swarm Optimization algorithm (PSO) and PSOCA.

REFERENCES

[AT08] **S. Abdullah, H. Turabieh** - *Generating university course timetable using Genetic Algorithms and Local Search*. The Third International Conference on Convergence and Hybrid Information Technology ICCIT, 1:254-260, 2008.

[AOO14] **O. M. Alade, E. O. Omidiora, S. O. Olabiyisi** - *Development of a University Lecture Timetable using Modified Genetic Algorithms Approach*, International Journal of Advanced Research in Computer Science and Software Engineering, 4(9):163-168, 214.

[A+16] **O. M. Alade, C. A. Oyeleye, O. T. Adedeji, E. O. Omidiora, S. O. Olabiyisi** - *Development of a modified Particle Swarm Optimization based Cultural Algorithm for solving University Course timetabling Problem*, LAUTECH Journal of Engineering and Technology 10 (1) : 1-5, 2016.

[BM00] **F. Benjamin, B. Marcel** - *Cultural algorithms: Concepts and experiments.* In Proceedings of the 2000 Congress on Evolutionary Computation, Piscataway, NJ, 2000. IEEE Service Center.:1245–1251, 2000.

[BN02] **E. K. Burke, J. P. Newall** - *A New Adaptive Heuristic Framework for Examination Timetabling Problems*, Working Group on Automated Timetabling, University of Nottingham. 8: 23-43, 2002.

[CK10] **A. Chaudhuri, D. Kajal** - *Fuzzy Genetic Heuristic for University Course Timetable Problem*. IJASCA, 2(1): 1-13, 2010.

[IDM09] **S. Irene, S. Deris, S. Mohad Hashim** - *A combination of PSO and Local Search in University Course Timetabling Problem*. International Conference on Computer engineering and Technology. 492-495, 2009.

[MM12] **S. K. Meysam, S. A. Mohammad** - *Hybrid Genetic Algorithms for University Course Timetabling*. International Journal of Computer Science Issues, 9(2)(2): 446-455, 2012.

[O+12] **C. A. Oyeleye, S. O. Olabiyisi, E. O. Omidiora, J. B. Oladosu** - *Performance Evaluation of Simulated Annealing and Genetic Algorithm in solving examination timetabling problem*, Scientific Research and Essays 7(17): 1727-1733, 2012.

[Qin10] **B. Qinghai** - *Analysis of Particle Swarm Optimization Algorithm*. Computer and Information Science. 3.(1):180-184, 2010.

[RZ01] **R. G. Reynolds, S. Zhun** - *Knowledge-based function optimization using fuzzy cultural algorithms with evolutionary programming*. IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics. 31(1): 1-18, 2001.

[SS11] **Y. Shengxiang, N. Sadaf** - *Genetic Algorithms with guided and local search strategies for university course timetabling*. IEEE Transactions on systems, MAN and Cybernetics- Part C: Applications and reviews 41(1):93-106, 2011.

[Yuh04] **S. Yuhui** - *Particle Swarm Optimization Electronic Data Systems*, Inc. IEEE Neural Networks Society: 8-13, 2004.

[ZJZ04] **J. Zheng, J. Jing, C. Zhihua** - *Particle swarm optimization algorithm [M]*. Science publishing Company of Beijing, 2004.