LAUTECH Journal of Engineering and Technology 10 (1) 2015: 1-5

DEVELOPMENT OF A MODIFIED PARTICLE SWARM OPTIMIZATION BASED CULTURAL ALGORITHM FOR SOLVING UNIVERSITY TIMETABLING PROBLEM

Alade[1] O. M.[,*], Oyeleye[2] C. A.[,*], Adedeji[3,*] O. T., Omidiora[4] E. O. and Olabiyisi[5] S. O.

Department of Computer Science and Engineering, Ladoke Akintola University of Technology, Ogbomoso, Nigeria.
E-mail: [1]oalade75@lautecg.edu.ng; [2]caoyeleye@lautech.edu.ng; [3]otadedeji@lautech.edu.ng; [4]eoomidiora@lautech.edu.ng; [5]soolabiyisi@lautech.edu.ng; *Corresponding Authors

ABSTRACT

Timetabling problems are search problems in which courses must be arranged around a set of timeslots so that some constraints are satisfied. However, slow convergence speed and high computational complexity are one of drawbacks limiting the efficiency of the existing timetabling algorithms. In this paper, a Modified Particle Swarm Optimization based Cultural Algorithm which is characterized with low computational complexity and high convergence speed was developed for solving university lecture timetabling problems. The standard Particle Swarm Optimization (PSO) algorithm was modified by introducing influence factors and acceleration component in order to improve the converge speed of the algorithm. Cultural algorithm was formulated by incorporating the Modified Particle Swarm Optimization (MPSO) into its population space. Thus, the developed Modified Particle Swarm Optimization based Cultural Algorithm could be implemented and employed for solving lecture timetabling problems in higher institutions.

Keywords: Particle Swarm Optimization Algorithm, Cultural Algorithm, Timetabling,

Introduction

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 (Wren, 1995; Thanh, 2006; Burke and Newall, 2002; Abdullah and Turabieh, 2008; Oyeleye, Olabiyisi, Omidiora and Oladosu,2012).Timetabling problems can be divided into course and examination timetabling. The course timetabling problem basically involves the allocation of courses, rooms, and students to a stable time period, normally a working week, while satisfying a given number of constraints which are hard and soft constraints (Alade, Omidiora and Olabiyisi, 2014). Hard constraints are constraints that must be fulfilled, while soft constraints are to be fulfilled as much as possible (Burke and Newall 2003; Brailsford, Potts and Smith 1999).

Cultural Algorithm (CA) is a technique that incorporates domain knowledge obtained during the evolutionary process so as to make the search procedure more effective (Reynolds and Zhun, 2001). 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 (Benjamin and Marcel, 2000).

Particle swarm optimization (PSO) is one of the evolutionary computational techniques and population-based search algorithms (Yuhui, 2004). 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 (Zheng, Jie and Cui, 2004; Qinghai, 2010).

Different methods had been applied to tackle the problems associating with timetabling. These include sequential methods that treat timetable problems as graph problems, cluster methods in which the problem is divided into a number of event sets, constraint based methods and meta heuristics methods such as genetic algorithms, simulated annealing, ant colony algorithm, cultural algorithm and other heuristic approaches (Carlos, 2000; Carlos, David and Gary, 2002; Chiarandini, 2006; Abdullah and Turabieh, 2008; Wilke, Grobner and Oster, 2002; Shengxiang and Sadaf, 2011; Alade, Omidiora and Olabiyisi, 2014). Hybrid of more than one Meta heuristics methods had also been proposed in literature. Examples are hybrid of genetic algorithms and fuzzy logic (Chaudhuri and Kajal, 2010), Hybrid of Genetic algorithms with Guided and Local Search

Strategies (Shengxiang and Sadaf, 2011), hybrid of genetic algorithms and simulated annealing (Oyeleye *et al*., 2012), hybrid of genetic algorithms and fuzzy logic with randomized iterative local search, simulated annealing and tabu search (Meysam and Mohammad, 2012), hybrid particle swarm optimization- constraint-based reasoning (Irene, Deris and Mohad, 2009), among others.

Research Methodology

In this paper, CA and MPSO were proposed 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.

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 3.1). 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 $q_3 r_3 \left(R_{cd}^i - X_{id}^i\right)$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:

$$V_{id}^{i+1} = \chi\left(V_{id}^i + q_1 r_1\left(R_{id}^i - X_{id}^i\right) + q_2 r_2\left(R_{gd}^i - X_{id}^i\right)\right) \quad 2.1$$

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

In Modified PSO, the update formula now becomes

$$V_{id}^{i+1} = \chi\left(V_{id}^i + q_1 r_1\left(R_{id}^i - X_{id}^i\right) + \mu_1 q_2 r_2\left(R_{gd}^i - X_{id}^i\right) + \mu_2 q_3 r_3\left(R_{cd}^i - X_{id}^i\right)\right) \quad 2.3$$

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

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

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

$q_1 r_1 \left(P_{id}^i - X_{id}^i\right)$ is a cognitive component which measures the performance of the particles i relative to past performance

$q_2 r_2 \left(P_{gd}^i - X_{id}^i\right)$ is a social component which measures the performance of the particles relative to a group of particles or neighbors?

$q_3 r_3 \left(P_{cd}^i - X_{id}^i\right)$ 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.

$$f\left(P_k^i\right) \le f\left(P_k^{i-1}\right) \le .... \le f\left(P_k^1\right) \quad 2.5$$

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

$$V_{id}^{i+1} = \chi\left(V_{id}^i + q_1 r_1\left(R_{id}^i - X_{id}^i\right) + \mu_1 q_2 r_2\left(R_{gd}^i - X_{id}^i\right) + \mu_2 q_3 r_3\left(R_{cd}^i - X_{id}^i\right)\right) \quad 2.6$$

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

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

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

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

$P^1 = \{p_1, p_2 \ldots\ldots p_8\}$, $P^2 = \{p_9, p_{10}, \ldots\ldots, p_{16}\}$ …

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 ε 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

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

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

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

Decision variables

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

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

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^{ld} (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 $$\sum_{s \varepsilon E} \left(C_s^{ldp} + C_s^{3R} + C_s^{id} + C_s^{sr} \right) \quad 2.8$$

C_s^{ldp}, C_s^{3R}, C_s^{ld} 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 (2.9) to (2.14)

Subject to

$$\forall e \varepsilon E \qquad \sum_{r \varepsilon R} \sum_{p \varepsilon P} x_{e,r,p} = 1 \qquad 2.9$$

$$\forall s \varepsilon S \qquad C_s^{ldp} = \sum_{e \varepsilon E} \sum_{r \varepsilon R} \sum_{q \varepsilon L} t_{s,e} x_{e,r,q} \qquad 2.10$$

$$\forall s \varepsilon S \qquad C_s^{3R} = \sum_{\substack{i,j,k,\varepsilon E \\ i \neq j \neq k}} \sum_{r \varepsilon R} \sum_{\substack{p,q,m \varepsilon P \\ q = p+1 \\ m = q+1}} t_{s,i} t_{s,j} t_{s,k} x_{i,r,p} x_{j,r,q} x_{k,r,m} \qquad 2.11$$

$$\forall s \varepsilon S \forall d \varepsilon D^d \qquad z_{s,d} = \begin{cases} 1 & \sum_{e \varepsilon E} \sum_{r \varepsilon R} \sum_{i \varepsilon P} t_{s,e} x_{e,r,i} = 1 \\ 0 & otherwise \end{cases} \qquad 2.12$$

$$\forall s \varepsilon S \qquad C_s^{1d} = \sum_{q \varepsilon D^d} z_{s,q} \qquad 2.13$$

$$\forall r \varepsilon R \forall p \varepsilon P \qquad C_s^{sr} = \sum_{e \varepsilon E} s_e^E x_{e,r,p} \le s_r^R \qquad 2.14$$

Equation (3.9) describes the implicit constraint which means that timetable solution must be complete and each event must be presented once. Equation (2.10) for (S1), equation (2.11) for (S2), equation (2.13) for (S3), equation (2.12) is necessary for describing (S3) which penalizes students who have only attended a single event in a day by 1, while equation (2.13) calculates all violations of any students for all days. Equation (2.14) for (S4). Also in equation (2.14) 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.

Conclusion and Future Work

In this paper, we have been able to develop a Modified Particle Swarm Optimization based Cultural algorithm for solving lecture timetabling problems in universities or other related higher institutions. The algorithm 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. It is recommended that future research may be geared towards implementing and analyzing the performance of the developed algorithm.

References

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

Alade O.M, Omidiora E.O and Olabiyisi S.O (2014): 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

Benjamin Franklin and Marcel Bergerman.(2000): Cultural algorithms: Concepts and experiments. In *Proceedings of the 2000 Congress on Evolutionary Computation*, Piscataway, NJ, 2000. IEEE Service Center.:1245–1251.

Brailsford, S.C., Potts C.N. and Smith, B.M. (1999): "Constraint Satisfaction Problems: Algorithms and Applications," *European Journal of Operational Research*, 119:557-581.

Burke, E. K. and Newall, J. P. (2002): "A New Adaptive Heuristic Framework for Examination Timetabling Problems," Working Group on Automated Timetabling, University of Nottingham. 8: 23-43.

Burke, E.K. and Newall J.P. (2003): "Enhancing Timetable Solutions with Local Search Methods", The Practice and Theory of Automated Timetabling, Lecture Notes in Computer Science. Springer. 2740:195-206.

Carlos A.C, David A. V, and Gary B. L. (2002): *Evolutionary Algorithms for Solving Multi-*

Objective Problems. Kluwer Academic Publishers, New York.

Chaudhuri A, and De Kajal (2010): "Fuzzy Genetic Heuristic for University Course Timetable Problem". IJASCA, 2(1): 1-13

Chiarandini, M., Birattari, M., Socha, K and Rossi-Doria, O. (2006): An effective Hybrid Algorithm for University Course Timetabling. *Journal of Scheduling*. 9(5):403-432.

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

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

Oyeleye, C. Akinwale, Olabiyisi, S. Olatunde, Omidiora, E. Olusayo and Oladosu, J. Babalola: Performance Evaluation of Simulated Annealing and Genetic Algorithm in solving examination timetabling problem, Scientific Research and Essays 7(17): 1727-1733.

Qinghai Bai. (2010): Analysis of Particle Swarm Optimization Algorithm. Computer and Information Science. 3.(1):180-184.

Reynolds R G, and Zhun S. (2001): 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

Reynolds R. G.(1994): An Introduction to Cultural Algorithms. In A. V. Sebald and L. J. Fogel, editors, *Proceedings of the Third Annual Conference on Evolutionary Programming* World Scientific, River Edge, New Jersey, :131– 139.

Shengxiang Y. and Sadaf N. (2011): 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

Thanh, N. D. (2006):" Solving timetabling problem using genetic and heuristics algorithms", Journal of Scheduling, 9(5) :403-432
Towards Memetic Algorithms. Caltech Concurrent Computation Program, Report 790,

Wilke, P., Grobner, M., Oster, N. (2002): A Hybrid Genetic Algorithm for School Timetabling. In: McKay, B., Slaney, J. (eds.) AI 2002: Advances in Artificial Intelligence, Springer. Verlag, New York :455–464.

Wren, A.(1995): "Scheduling, timetabling and rostering – A special relationship?" Proceeding of the 1st International Conference on the Practice and Theory of Automated Timetabling. : 474-495.

Yuhui Shi (2004): Particle Swarm Optimization Electronic Data Systems, Inc. IEEE Neural Networks Society: 8-13.

Zheng, J., Jie Jing and Cui Zhihua (2004): Particle swarm optimization algorithm [M]. Science publishing Company of Beijing.