ACSIJ Advances in Computer Science: an International Journal, Vol. 3, Issue 5, No.11 , September 2014
ISSN : 2322-5157
www.ACSIJ.org

A Review of Distributed Multi-Agent Systems Approach to Solve University Course Timetabling Problem

Hamed Babaei[1*], Amin Hadidi[2]

[1*] Department of Computer Engineering, Ahar Branch, Islamic Azad University, Ahar, Iran
hamedbabaei63@gmail.com

[2] Department of Mechanical Engineering, Ahar Branch, Islamic Azad University, Ahar, Iran
amin.hadidi@yahoo.com

Abstract

Scheduling is one the problems on which so many researches have been done over years and university course timetabling problem which is a NP-hard problem is a type of scheduling problem. On the other hand, timetabling process must be done for each semester frequently which is an exhausting and time consuming task. University course timetabling problem performs the allocation of whole events to timeslots and rooms by considering the list of hard and soft constraints presented in one semester, so that no conflict is created in such allocations. In this problem, the hard constraints should not be violation under any condition, but soft constraints should not be violation as much as possible. The aim of this paper is to analyze a new approach to solve university course timetabling problem called an approach based on multi agent systems (Cooperative Search) in addition to briefly study approaches based on operational researches, meta-heuristic methods and intelligent novel methods.

Keywords: *University Course TimeTabling Problem (UCTTP), Approach Based on Multi-Agent Systems (Cooperative Search), Operational Researches Methods, and Meta-heuristic methods, Intelligence novel methods.*

1. Introduction

The goal of the university course timetabling problem (UCTTP) is to find a method to allocate whole events to fix predefined timeslots and rooms, where all constraints within the problem must be satisfied. Events include students, teachers and courses where resources encompass the facilities and equipment's of classrooms such as theoretical and practical rooms. Also timeslots include two main components, namely daily and weekly timeslots which it varies from one institution to another. However, each classroom also has its own components including audio-visual equipment's (video projector), number of chairs necessary for courses allocated to those classrooms (the capacity of theory and practical rooms), number of blackboards and whiteboards related to each theory and practice classroom and etc….

1.1 Description of the problem

UCTTP is a hybrid optimization problem in the class of NP-hard problems occur at the beginning of each semester of universities and includes the allocation of events (courses, teachers and students) to a number of fixed timeslots and rooms. This problem must satisfy both hard and soft constraints during allocation of events to resources, so that the possible timetables are obtained after full satisfaction of whole hard constraints and also soft constraints to increase and promote the quality of possible generated timetables as necessary [1-3].

There are some problems and complexities in UCTTP process; firstly, the scheduling process is an NP-complete problem, then it could not be solved in the polynomial time classes because of the exponential growth of this problem and the existence of some variations in the fast growth of students' numbers in this problem, so we must seek heuristic approaches. Secondly, the number of constraints (hard and soft) in this problem differs from one institution to another. Therefore, the main aim of all of the mentioned algorithms is to maximize the number of soft constraints satisfied in the final timetables [1] [4].

1.2 The basic definitions of the problem

- Event: a scheduled activity, like: teacher, course, and student.
- Timeslot: a time interval in which each event is scheduled, like: weekly timeslot such as Tuesday and daily timeslot such as 8 a.m. to 9 a.m. and etc.
- Resource: resources are used by events, like: equipment's, rooms, timeslots and etc.
- Constraint: a constraint is a restriction in scheduling of events, categorized into two types of hard and soft constraints, like the capacity of classrooms, given timeslot and etc.
- People: People include lecturers and students and are a part of events.
- Conflict: the confliction of two events with each other, like: scheduling of more than one teacher for one classroom at the same time.

1.3 Different types of constraints in the problem

Constraints in UCTTP problem are classified into two classes of hard and soft constraints. Hard constraints must be satisfied in the problem completely so that the generated solution would be possible and without conflict; no violation is allowed in these constraints. Soft constraints are related to objective function; objective function is to maximize the number of satisfied soft constraints. Unlike hard constraints, soft constraints are not necessarily required to satisfy; but as the number of these satisfied constraints increases, the quality of solutions of objective function increases. In the following, a list of hard and soft constraints presented which are taken from literature [1-6].

1.3.1 Hard constraints

- A teacher could not attend two classes at the same time.
- A course could not be taught in two different classes at the same time.
- A teacher teaches only one course in one room at each timeslot.
- At each daily timeslot in one room only one group of students and one teacher could attend.
- A teacher teaches for only one group of students at each daily timeslot.
- There are some predefined courses which are scheduled in a given timeslots.
- The capacity of the classrooms should be proportional to the number of students of the given course.

1.3.2 Soft constraints

- The teacher can have the choice to suggest priority certain timeslots for her/his courses either public or private times.
- A teacher may request a special classroom for a given course.
- The courses should be scheduled in a way that the empty timeslots of both teacher and student to be minimized.
- Timetabling of the courses should be conducted in a way that the courses not scheduled at evening timeslots, as it is possible; unless an evening timeslot has been requested by a particular teacher.
- The lunch break is either 12 p.m. to 13 p.m. or 13 p.m. to 14 p.m., usually.
- The start time of classes may be 8 a.m. and the ending time may be 20:30 p.m. (evening), usually.
- The maximum teaching hours for teachers in a classroom are 4 hours.
- The maximum learning hours for students is 4 hours.
- Scheduling should be conducted in a way that one or a group of students not attend university for one timeslot in a day.

1.4 Mathematical formulation of the problem

Formal definition of UCTTP problem includes n: the number of events $E=\{e_1, e_2, \dots, e_n\}$, k: the number of timeslots $T=\{t_1, t_2, \dots, t_k\}$, m: the number of rooms $R=\{r_1, r_2, \dots, r_m\}$, L: the number of rooms' features $F=\{f_1, f_2, \dots, f_l\}$ and s: the set of students $S=\{s_1, s_2, \dots, s_s\}$. For example, if the number of daily timeslots is 9 and the number of weekly timeslots is 5, then the total timeslots will be $T= 9 \times 5 = 45$ [1-3] [10].

The input data for each sample problem (data sets) include the size and features of each room, the number of students in an event and information about conflicting events. So, we should know the procedure of measuring violation and non-violation of hard and soft constraints in order to have the ability to replace events within matrixes. At first the penalty function per violation from soft constraint must be calculated for each solution which is corresponding to a timetable, as bellow [1-3] [10]:

$$PF(S)=\sum_{j=1}^{SC} W_j \times (-1) \tag{1}$$

In Eq. (1), S is the solution, W_j is the weight of each soft constraint (value 0 means non-violation, value 1 means violation and -1 shows the cost of each violation per soft constraint) and SC is the number of soft constraints. However, PF represents the penalty function. Value of objective function per solution considering hard constraints can be calculated as:

$$OF(S)=\sum_{i=1}^{HC} W_i \times (-1) + PF(S) \tag{2}$$

In Eq. (2), W_i is the weight of each hard constraint where value 0 means non-violation, value 1 means violation and -1 shows the cost of each violation per hard constraint. Also HC and OF are the number of hard constraints, and the objective function, respectively. Always the value of first term of right hand side of the Eq. (2) is equal to zero ($\sum_{i=1}^{HC} W_i \times (-1)$ =0), this means that the violation of hard constraints is not feasible. So $OF (S) = 0 + PF (S)$, consequently $OF (S) = PF (S)$.

In order to determine the violation of solutions, from hard and soft constraints, results of sample problems are stored in 5 matrixes namely STUDENT-EVENT, EVENT-CONFLICT, ROOM-FEATURES, EVENT-FEATURES and EVENT-ROOM which is introduced in the following. Each event is met by each student which is stored in the matrix STUDENT-EVENT. This matrix called matrix A is a $k \times n$ matrix. If the value of $U_{i,j}$ in the matrix $A_{k,n}$ be 1, then student $i \in S$ must attend event $j \in S$, otherwise, its value will be 0. The matrix size is $k \times n = |S| \times n$. The EVENT-CONFLICT matrix is an $n \times n$ matrix with two arbitrary events which could be scheduled in the same timeslots. This matrix called matrix B is used to quickly identify events which potentially allocated to same timeslots. ROOM-FEATURES matrix is an $m \times l$ matrix which shows the features of each room; this matrix called matrix C. If the value of $C_{i,j}$ be 1, then each $i \in R$ has a feature of $j \in F$, and otherwise its value will be 0. The matrix size is $m \times l = m \times |F|$. The EVENT - FEATURE matrix also called matrix D is an $n \times l$ matrix and represents the features required by each event. Namely, event $i \in E$ requires features of $j \in F$, if and only if $d_{i,j}$=1. The matrix size is an $n \times l = n \times |F|$. Finally the EVENT-ROOM matrix called G matrix is an $n \times m$ matrix which represents the list of possible rooms so that each event could be allocated in those rooms. This matrix represents the quick identification of all rooms in terms of their size and features for each appropriate event. The matrix size is $n \times m$ [1-4] [6].

1.5 The approaches used in the study of UCTTP

The first definition of timetabling has been presented as three sets of: 1) teachers, 2) classrooms and 3) timeslots [5]. Approaches used to solving the UCTTP problem up to now are as follows: 1) Operational Researches (OR) based techniques including graph coloring theory based technique, IP/LP method and Constraint Based Satisfaction(s) technique (CPSs); 2) Meta-heuristic approaches also including Case Base Reasoning method (CBR), population based approaches and single solution based approaches where the population based approaches includes Genetic Algorithms (GAs), Ant Colony Optimization (ACO), Memetic Algorithm (MA), Harmonic Search Algorithm (HAS) and single solution algorithms also includes Tabu Search Algorithm (TS), Variable Neighborhood Search (VNS), Randomized Iterative Improvement with Composite Neighboring algorithm (RIICN), Simulated Annealing (SA) and Great Deluge Algorithm (GD); 3) multi criteria and multi objective approaches; 4) intelligent novel approaches such as hybrid approaches, artificial intelligence based approaches, fuzzy theory based approaches and 5) distributed multi agent systems approach [2- 4] [6] [7].

1.6 Motivation and historical perspective of the problem

Agents are technologies inspired from global environment to develop initial instances of systems. Whenever a distributed multi agent system is considered, it means that there is a network of agents collaborates with each other to solve problems which are out of capability of each single agent [8]. Recently, using distributed multi agent systems based approach to solve UCTTP problem has been applied by [9] where in the this method, a solution is used to deal with UCTTP problem using distributed environment and an interface agent -which is responsible to cooperate different timetabling agents- collaborate with each other to improve the solution of common goal. The initial timetables are generated for multi agent systems by using multiple hybrid meta-heuristics which are a combination of graph coloring meta-heuristics and local search in different methods. The hybrid meta-heuristics provide the capability to generate possible solutions for all samples of both [22] and international competitions timetabling 2002 datasets. However, recently, [10] has used distributed agents to create UCTTP by considering hard (necessary) and soft (desirable) constraints. Also, he presented fairly meeting of distribution in allocating resources in his Ph.D. thesis. There are two types of agents in that model which are year- programmer agent and rooms' agent. However, there are four principles to efficiently organize agents, including: 1) queue and the sequential queue algorithm, 2) queue and interleaved queue algorithm, 3) round robin and sequential round robin algorithm and 4) round robin and interleaved round robin algorithm. The problem formulation and dataset have been adopted from the third section of ICT-2007. The obtained result ensures the consistency of interleaved round robin principle for year-programmer agents in the system and the fairest chance in obtaining the required resources.

1.7 Aim of the Paper

In this paper the aim is to first review approaches in solving UCTTP problem in detail and then introduce distributed multi agent systems based approach to solve UCTTP problem. Also researches on application of distributed multi agent approach to create infrastructure of scheduling common events among multiple departments have been reviewed. In addition, in this paper approaches used in solving UCTTP is classified in four categories as:

- Operational Research (OR) methods,
- Meta-heuristic methods,
- Intelligent novel methods,
- The approach based on distributed multi agent systems (Cooperative Search).

2. Related work

Approaches which are used to study the UCTTP problem up to now surveyed in this section.

2.1 Operational Research (OR) methods

graph coloring approach about the way of modeling a UCTTP problem by using a undirected graph where [11] has used vertices as events, colors as timeslots and edges as constraints in a graph to solve timetabling problem where here no two adjacent vertices should not be co-color; since a mark of conflict has been provided in timetabling. However, [12] has introduced the idea of separating graph vertices to reduce the chromatic number of graph and has applied it to separate students. Here, the separation of one vertex is similar to separation of students in one course. The method of coloring the edges of two part graph also has been done by [13] in order to reduce the number of penalties and build high quality timetables over manual timetables. The scheduling of classrooms has also been performed by using graph coloring method through [14] where vertices and edges represent common courses and students, respectively and the aim is to present a VC* heuristic approach in order to: 1) promote the uniform distribution of courses over colors and 2) balance the number of courses for each timeslot over the existing rooms. Another hybrid approach to solve UCTTP problem by using genetic coloring has been proposed by [15] where this method reduces the cost of finding the least number of required colors to color a graph with this hybrid method.

In [16], IP (integer programming) has been presented for UCTTP problem where the aim is to allocate a set of courses among teachers and groups of students and also a set of weekly and daily time periods pairs. Again, [17] has presented a IP-based two-step relaxation method to generate efficient solutions of timetabling where during step 1, the courses sessions which require consecutive have been done through allocation of courses to days and special times and during step 2, the ensuring on consecutive of those number of courses which require more than one time period for identical student groups has been done. An integer programming (IP 0/1) approach has been presented by [18] to organize courses and teachers according to available timeslots and classrooms where its result is reduction of dissatisfaction of students and teachers during implementation of finite rules by a set of constraints simultaneously. However, ILOC software has been applied to implement CSP approach by [19] performed to build timetabling in university and its result was to reach an objective function due to satisfaction of events' constraints in allocating to resources.

2.2 Meta-heuristic methods

In [20], a genetic algorithm has been used to sort a university timetabling where the crossover rate was 70% and however, no hard constraint has been violated in timetabling and applied constraints mostly were occupied over room and capacity of rooms. However, [21] has proposed a novel GA technique to solve UCTTP problem which uses a learner machine. The results of this technique is to minimize the number of violated soft constraints, high amount of using available rooms and reduction of teachers' workload.

Using MAX-MIN ant system to generate university courses timetabling by [22] has led to build an optimal path where each path could generate a constructive graph to allocate courses to timeslots with influence on amount of pheromone within a range. However, applying ant colony optimization algorithm by [23] for UCTTP problem has been done post submission according to ITC-2007 dataset where ants allocate events to rooms and timeslots based on two types of pheromone T_{ij}^{s} and T_{jk}^{y}. This algorithm has acted well on timetabling and produces better results during longer run time. Applying a hybrid ant colony system has been proposed in [24] to solve UCTTP problem. Here, two type of hybrid ant system including combination of SA with AC and TS with AC has been presented. A number of ants perform the complete allocation of courses to timeslots based on a predefined list. The selection of timeslots probabilities by ants to allocate courses has been done by using heuristic information and information of an indirect coordinator mechanism among agents and activities within an environment.

The memetic algorithm has been performed by using [25] to solve UCTTP problem through a combination of local search method in genetic algorithm. One of the local searches has been run over events and another one over timeslots.

The Tabu search algorithm has been applied by [26] for the first time to allocate students to courses and also balance the number of students in one submitted group where the first phase is to generate a set of solutions for one student, the second one is to combine a set of solutions with applying Tabu search with local strategies and the third phase is to allocate rooms and improve the allocation, of course without modifying the initial allocation of courses to timeslots. In [27], the influence of neighboring structures on Tabu search algorithm to solve UCTTP problem has been presented where the influence of simple and swap transmissions has been tested on Tabu search operations which were based on neighboring structures. Here, 4 new neighboring structures have been applied and compared.

To solve UCTTP problem, the combination of kempe neighboring chain in simulated annealing algorithm has been presented by [28] where here by relaxation one of the hard constraints reformulation is done and then this constraint is created in the form a relaxed soft constraint. However, the relaxation problem is analyzed in two steps: 1) to create a feasible solution a heuristic based graph is used and 2) a simulated annealing algorithm has been used to minimize the violations of soft constraints (in the second phase, a kempe neighboring chain based heuristic has been used). However, the simulated annealing approach by [29] to solve UCTTP problem compares the efficiency of different neighboring search algorithms based on simple search, swap search, simple-swap search and computes the run time cost for each method. The highest satisfaction of timetabling is obtained by combination of three algorithms.

Using two local search and memetic algorithms to solve UCTTP problem has been done by [30] where two improved simulated annealing and memetic algorithms have been combined. However, the modification of crossover operator in memetic algorithm and creating an initial population has been done by a heuristic method. The improvement operator has been done in order to optimize the generated chromosomes and reduce the number of violations in constraints. However, [31] has used local search strategy directed within genetic algorithm to solve UCTTP problem where the directed search strategy uses a data structure to generate children which stores the extracted information of good people of previous generations. The results of local search combined in genetic algorithm are satisfiable. The aim is to maximize allocations and minimize violations of soft constraints.

The Variable Neighborhood Search algorithm (VNS) BY [32] has been presented to solve UCTTP problem which proposed a basic VNS and then to use an exponential Monte Carlo acceptance criterion by each solution, it has given some terms. The main idea is to apply Monte Carlo acceptance criterion, improve the explorations through acceptance of the best solution with the given probability in order to find the number of promised neighbors. Again, [33] has presented a Randomized Iterative Improvement with Composite Neighboring algorithm (RIICN) to improve its previously presented algorithm which is in fact the combination of VNS and local search. The Tabu list has been applied to penalty of inefficient and un-promised neighboring structures after a given number of iteration.

2.3 Intelligent novel methods

A hybrid algorithm by [34] used sequential heuristic and simulated annealing to solve UCTTP problem over ITC-2002 dataset has been presented. This method includes three phases as: phase one; using a sequential heuristic to generate the feasible initial timetables, phase two; applying simulated annealing to minimize the number of violations of soft constraints and phase three also uses simulated annealing algorithm to increase the improvement of generated timetables. A hybrid evolutionary approach has been presented to solve UCTTP problem by [35] where the combination of local search algorithm with evolutionary approaches give better results. The aim of problem is to completely meet of hard constraints and minimize the violations of soft constraints. The obtained results of hybrid evolutionary approach shows the minimization of penalty values from soft constraints.

However, [36] also presented another hybrid approach to solve UCTTP problem implemented by three phases. The first phase is to generate the initial solution by using constructive heuristics, the second phase uses an improved technique applying randomized iterative improvement algorithm and the third phase also uses simulated annealing as an acceptance criterion. However, the hill climbing has been used in the third phase to promote the quality of timetables. Integration of two Tabu search and memetic algorithms to solve UCTTP problem has been done by [37] where the crossover and mutation operators have been used to select a solution from a population and then the random neighboring structures have been reused for each solution whose quality of solutions have not been promoted within Tabu list. The Tabu list is applied to penalty neighboring structures which do not have the capability to generate better solutions. Recently, a multi population hybrid genetic algorithm has been proposed by [38] to solve UCTTP problem based on three genetic algorithms, FGARI, FGASA and FGATS. In this algorithm, fuzzy logic is used to evaluate the number of violations of soft constraints in fitness function to deal with real world data which are ambiguous and non-deterministic and random, local search, simulated annealing and Tabu search methods would also be useful accompanied with fuzzy method to improve inductive search to satisfy the search ability.

To optimal fuzzy classification of students, [39] has used a fuzzy function to solve UCTTP genetic programming problem. The aim was to separate the students of populous classes. This separation has led to reduce the amount of conflict of students' courses in weekly program. Here, at first the fuzzy mean c-clustering algorithm divides students into c classes and then according to the criteria of distance of clusters' centers, density of each cluster, co-entrance of students of each cluster and dimension ratio of clusters by using a fuzzy function, the value of clustering is determined so that by selecting the appropriate features (courses), the best classification of students is obtained. A hybrid fuzzy evolutionary algorithm has been presented by [40] to multi objective resource allocation problem which was a student's project allocation problem. Here, student project allocation must satisfy a number of soft purposes in a sequence of some points. This algorithm uses a fuzzy inductive system to model and collect purposes. Fuzzy system considers some priorities to decide on agreement among different purposes by which the direction of search path toward attractive regions within purpose space is performed. To solve UCTTP problem, [41] has presented a fuzzy multiple heuristic sorting method where the sorting of events has been done through simultaneous considering of three distinct heuristics by using fuzzy methods. The sequential combination of three heuristics is sorted as 1- the highest degree, 2- saturation degree and 3- submission degree and fuzzy weight of an event is also used to represent that event has what problem to be scheduled. The descending sorted events are allocated to the last timeslot with the least penalty cost sequentially while the feasibility is maintained in whole process. A fuzzy solution based on memetic approach has been presented by [42] to solve university timetabling problem where a timetable has been compared with both genetic and memetic algorithms and its results may satisfy the existing constraints simultaneously during a shorter time interval. The aim is to use fuzzy logic as a means for local search in memetic algorithm.

[43] Has presented fuzzy genetic heuristic idea to solve UCTTP problem where the genetic algorithm has been applied by using indirect representation based on integration events features and the fuzzy set model is also to evaluate the violation of soft constraints in objective function according to uncertainties of real world data. Here, a degree of uncertainty within objective function is considered for each soft constraint and this uncertainty is evaluated by formulating violation parameter from soft constraint in objective function using fuzzy membership functions. However, a fuzzy genetic algorithm has been presented by [44] accompanied with local search to solve UCTTP problem where the fuzzy genetic algorithm with local search algorithm uses inductive search to solve the combined problem and applied local search which has the ability of improving efficiency within genetic algorithm. The applied fuzzy logic within this approach is also used to evaluate the violation of soft constraints in objective function due to facing with uncertainty in real world data.

However, recently [45] has used a novel clustering technique based on FP-Tree to solve UCTTP where the given technique is done to classify students based on their selective courses who submitted for the next semester. The aim of this clustering is to solve scheduling of courses where in the previous semesters the submission of students in some courses due to simultaneous scheduling has been prevented, while in this technique no conflict would happen over scheduling of exams since no two exams at the same time would be taken for courses by two identical groups of students.

3. Studying the approach based on distributed multi agent systems in UCTTP

In [8], an agent could observe and receive anything through sensors from its environment and then performs over environment through a driver. Agents are classified into different classes based on their application, including 1) autonomous, 2) intelligent, 3) reaction able, 4) proactive, 5) learner, 6) mobile, 7) cooperative/communicative agents.

3.1 Multi agent systems

Therefore, agents must have a common language and a communication media to cooperate with each other where these two components are essential among two agents. Multi agent systems have a more general concept and for all types of current systems, including multiple autonomous components are applied to the following features and include: 1) each agent has the ability of solving a problem incompletely, 2) in multi agent systems there is no general control system, 3) data are as distributed and 4) computations are asynchronous. However, after stating the autonomous features, we can explain the multi-capacity features in agents as: 1) dividing tasks among a large number of agents which are modular making, flexibility, modifiable and extensible, 2) the knowledge released over different resources (agents) has the capability of being integrated to completion, 3) applications require distributed computations by distributed multi agent systems for better support and 4) the agents technology provides the summary and result of distributed components technology [8] [10].

3.2 Studying the related works to solve UCTTP by distributed multi agent systems based approach

To generate course timetables, [46] have used distributed multi agent architecture. Here, UCTTP problem includes a set of courses in fixed timeslots in a circulating week. UCTTP problem refers to only a set of university departments. Each department has an education program corresponding to particular rules, constraints and purposes based on their resources and resources do not have shared feature, unless the resource exchange among departments would be useful which is done through negotiation. To solve the problem, a multi agent scheduling system based on one market with artificial money has been considered and each department includes three colleague agents: 1) to search a local solution, 2) to negotiate for resources with other departments and 3) to manage the related information. In [47], a distributed timetabling system is in three software layers as: 1) the first layer, scheduling agents, 2) the second layer, using a negotiation protocol to generate a pervasive university scheduling and 3) the third layer, presenting network infrastructures. Protocol implementation is to negotiate among scheduling agents and classroom agent. The main attempt is focused on studying the various places in the protocol of negotiation among scheduling agents.

The studying of distributed timetabling problem based on scheduling agents by [48], almost in real timetabling problems includes organized parts which requires creating timetables for people included in an independent way, while some global constraints are considered. Recently, department timetabling is combined with each other as a result of integration and consistent solution and this combination itself would require negotiation of various agents. Here, only one model including a reduplicated agent called CA (central agent) is studied and the duty of this agent is to cooperate of search process among all SA(s) (scheduling agents). However, this idea is in respect of creating feasible solutions for a network of SA(s). Presenting a multi agent system to create automatic timetabling with shared resources has been done by [49] where an automatic timetabling system has been proposed for a normal state of the universities. Agents (departments) compete for a set of classrooms over a number of given timeslots. Each agent applies its own algorithm and this algorithm may be unknown for other departments. A central system is assumed to determine whether some agents are allowed to allocate resources or not, which is also based on a list of requests received from each agent. The initial priority is evaluated by a number of attendances (expectations) and some of the requirements are valued for particular features over resources. Presenting a distributed technique for UETTP has been done by [50] based on a case study on a real university where the timetabling of distributed exams is performed by beginning of a distributed search for a global solution through computation of agents and this is as the best local scheduling where after that the negotiation among agents is done to find a global solution. In the presented method, there are two solutions to model multi agent systems in timetabling of exams where 1) applying SA(s) system (scheduling agents) and 2) applying Tabu search would be as a hyper heuristic, where the focus is on negotiation of procedures which would be able to detect and avoid collisions among SA(s) with common exams. The negotiation protocol consists of three steps where the first phase is on search for a local solution in timetabling problem at each agent, the second phase is a global timetabling without collision and the third phase is also the negotiation of SA(s) with each other to improve the local solution a creating timetabling.

To solve UCTTP problem, a global solution model based on agent has been presented by [51] where in this model each agent within the system is a constraint in UCTTP problem. The agent based solution includes generality, flexibility, dynamism and scalability which act better than other modes. Each constraint agent is an independent module in the proposed model and new constraint agents have the ability to join and leave simply the timetabling system which would be the dynamism result of constraint in resolving through system technique and design. To evaluate a system with mobile agents in UCTTP problem one test has been presented by [52] which have been done as a mechanism to solve UCTTP problem. Most focus was on the application of new technology of mobile agents to implement the solution of a general UCTTP problem. In modeling an agent, each mobile agent represents a course called course agent (CA) and course agents perform negotiation with other agents with a mechanism defined as a Signboard agent. A Signboard agent is considered for per day of the week and each platform represents the usual days of the week. The mobile agents' technology has been established based on TEEMA (TR Labs Execution Environment for Mobile) platform. The study of UCTTP problem as distributed DisTTP (Distributed TimeTabling Problem) has been done by [53] where previously this problem has been introduced as a distributed constraint satisfaction problem (DisCSP). To solve DisTTP problem based on multiple segmented constraint networks (MSCNs), an alternative method has been presented here. The network topology is based on sparse matrix and unlike DisTTP algorithm, a central agent is required proposed to the presented solution. The proposed method maintains a part of timetabling from all particular agents. The distributed timetabling avoids translation and interpretation of concentrated local constraints and their communication where this process keeps the shorter scheduling and privatization of departments. The obtained solution of a multi agent system consists of an SA (Scheduling Agent) at each department and a CA (Central Agent). SA(s) has been proposed for department timetabling to meet local constraints and CA collects department timetables and performs studies in respect of global constraints to modify and revisit department timetables. The implementation of class timetabling has also been done by [54] based on multi agent systems where the implementation process has been presented through applying hill climbing algorithm with the sharpest upward slope (until reaching the ancestor). CombinationGenerator and MinFinders agents are applied to generate maximum input combinations and create a combination with the minimum evaluation function to consecutive exams, respectively. Using this proposed method would continue the initial random solution until reaching the given optimal solution. A multi agent system has been proposed by [55] for UCTTP problem scheduling where two basic features have been defined by the distribution and the dynamism of environment. An efficient solution to solve this problem could be provided by an agent based approach. The focus is on architecture of multi agent systems which is presented for UCTTP timetabling called MAS-UP-UCT. The advantage of this method includes a large number of communications, collaboration and negotiation among agents.

Presenting a system model to UCTTP problem has been proposed by [56] using mobile agents where a multi agent system has been applied to generate the solutions of UCTTP problem. Four types of agents collaborate sequentially with each other to perform courses scheduling process as: 1) course (mobile), 2) signboard, 3) publisher and 4) mediator. The powerful key of this approach is to use the independency feature of agents and this independency is embedded clearly in the performance of course agent. Each course agent in the system is responsible to negotiate with other course's agents to find the satisfactory resource class (timeslots and rooms) in order to present and represent courses. Solving UCTTP problems by using multi agent systems requires the development of an intelligent decision making system proposed by [57] UCTTP problem is a dynamic distribution problem which requires a system of decision making, where agent are independent and a flexible communication methodology has been used to create the backbone of decision making system. The course agents represent each course in the problem and the negotiation of course agents with each other has also been applied by a signboard agent to find the collision prevented acceptable timetabling. The signboard agent uses a mechanism to identify course agents where each requires a negotiation with other course agents and in other words it performs resolving collisions. However, that mechanism is also built through completion of each timetable available for user. Here also the powerful feature of autonomous has been used for an agent to present all aspects of basic units in problem (course). In [58], has presented a new architecture for multi agent systems in solving UCTTP problem where agents allocate the required technical and human resources through negotiation as agencies from each course. The negotiation and description protocol of independency decisions of agents has been defined in the given framework format. The advantages of the multi agent approach are to resolve collisions directly, variability of strategic negotiation and present some real problems and events.

4. Discussion

After reviewing approaches which solved UCTTP problem, now we could reach an objective classification of these methods. Approaches based on operational research methods do not have good efficiency in solving such problem, but rather they have easier implementation, since they are analyzed by software integrated with efficient and heuristic algorithms. While in turn, the exploration of the search space of solutions performs more efficiently by applying meta-heuristic methods and novel intelligent techniques in analyzing this type of problems. However, we could not call a meta-heuristic approach as the best method to solve UCTTP problem, since the used datasets are diverse and the way of applying this type of methods is different as separately or in combination with other methods. While approaches to solve UCTTP problem on hybrid techniques (the combination of meta-heuristic methods) and multi agent based methods as a distributed architecture of UCTTP problem are important where in multi agent systems based approaches, the independency of scheduling process, negotiation of agents to remove the interference of event and resources with each other, flexibility of agents to combine with different types of heuristic methods in scheduling of common and single events per department, we could use datasets with different sizes as a test bed for above mentioned methods in order to find a reasonable and complete viewpoint on the structure of these algorithms [7].

5. Conclusion

In this paper, after comprehensive investigation of available approaches in the study of the UCTTP problem, we have focused on methods based on multi agent systems as a distributed architecture of the UCTTP problem. In this method, we have a general look at related works and how to apply them in solving the UCTTP problem where the advantages of using multi agent systems based approach could include increasing the independency of scheduling each department, independence of departments in scheduling, scalability in a distributed environment and to prevent collision among events/resources and unplanned allocation by negotiation among agents in a distributed environment.

References

[1] Obit, J. H., “Developing Novel Meta-heuristic, Hyper-heuristic and Cooperative Search for Course Timetabling Problems,” Ph.D. Thesis, School of Computer Science University of Nottingham, 2010.

[2] Asmuni, H., “Fuzzy Methodologies for Automated University Timetabling Solution Construction and Evaluation,” Ph.D. Thesis, School of Computer Science University of Nottingham, 2008.

[3] Redl, T. A., “A Study of University Timetabling that Blends Graph Coloring with the Satisfaction of Various Essential and Preferential Conditions,” Ph.D. Thesis, Rice University, Houston, Texas, 2004.

[4] Feizi-Derakhshi, M. R., Babaei, H., & Heidarzadeh, J., “A Survey of Approaches for University Course TimeTabling Problem,” Proceedings of 8th International Symposium on Intelligent and Manufacturing Systems (IMS 2012), Sakarya University Department of Industrial Engineering, Adrasan, Antalya, Turkey, 307-321, 2012.

[5] Gotlib, C.C., “The Construction of Class-Teacher TimeTables,” Proc IFIP Congress, 62, 73-77, 1963.

[6] Lewis, M. R. R., “Metaheuristics for University Course Timetabling,” Ph.D. Thesis, Napier University, 2006.

[7] Rossi, d. O., Sampels, M., Birattari, M., Chiarandini, M., Dorigo, M., Gambardella, L. M., Knowles, J., Manfrin, M., Mastrolilli, M., Paechter, B., Paquete, L., Stutzle, T., “A Comparison of the Performance of Different Metaheuristics on the Timetabling Problem,” Springer-Verlag Berlin Heidelberg, PATAT 2002, LNCS 2740, 329–351, 2003.

[8] Srinivasan, S., Singh, J., & Kumar, V., “Multi-Agent based Decision Support System Using Data Mining and Case Based Reasoning,” IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 4, No 2, 2011.

[9] Obit, J. H., Landa-Silva, D., Ouelhadj, D., Khan Vun, T., & Alfred, R., “Designing a Multi-Agent Approach System for Distributed Course TimeTabling,” IEEE, 2011.

[10] Wangmaeteekul, P., “Using Distributed Agents to Create University Course TimeTables Addressing Essential Desirable Constraints and Fair Allocation of Resources,” Ph.D. Thesis, School of Engineering & Computing Sciences Durham University, 2011.

[11] De Werra, D., “An Introduction to TimeTabling,” European Journal of Operational Research, 19, 151-162, 1985.

[12] Selim, S.M., “Split Vertices in Vertex Colouring and Their Application in Developing a Solution to the Faculty Timetable Problem,” The Computer Journal, Vol. 31, No. 1, 76-82, 1988.

[13] Hafizah, A. R., & Zaidah, I., “Bipartite Graph Edge Coloring Approach to Course Timetabling,” IEEE, 229-234, 2010.

[14] Dandashi, A., & Al-Mouhamed, M., “Graph Coloring for Class Scheduling,” Department of Computer Science, University of Balamand, Koura, Lebanon, 2010.

[15] Asham, G.M., Soliman, M.M., & Ramadan, A.R., “Trans Genetic Coloring Approach for Timetabling Problem,” Artificial Intelligence Techniques Novel Approaches & Practical Applications, IJCA, 17-25, 2011.

[16] Daskalaki, S., Birbas, T., & Housos, E., “An integer programming formulation for a case study in university timetabling,” European Journal of Operational Research, 153 (2004), 117–135, 2004.

[17] Daskalaki, S., & Birbas, T., “Efficient solutions for a university timetabling problem through integer programming,” European Journal of Operational Research, 160 (2005), 106–120, 2005.

[18] Bakir, M. A., & Aksop, C., “A 0-1 integer programming approach to a university timetabling problem,” Hacettepe Journal of Mathematics and Statistics, Vol 37 (1), 41-55, 2008.

[19] Zhang, L., & Lau, S., “Constructing university timetable using constraint satisfaction programming approach,” IEEE Proceedings of the 2005 International Conference on Computational Intelligence for Modeling, Control and Automation, and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC’05), 28-30 November, No. 2, 55-60, 2005.

[20] Khonggamnerd, P., & Innet, S., “On Improvement of Effectiveness in Automatic University Timetabling Arrangement with Applied Genetic Algorithm,” IEEE, 2009.

[21] Alsmadi, O. MK., Abo-Hammour, Z. S., Abu-Al-Nadi, D. I., & Algsoon, A., “A Novel Genetic Algorithm Technique for Solving University Course Timetabling Problems,” IEEE, 2011.

[22] Socha, K., Knowles, J., & Samples, M., “A Max-Min Ant System for the University Course Timetabling Problem,” Proceedings of the 3rd International Workshop on Ant Algorithms (ANTS 2002), Lecturer Notes in Computer Science 2463, Springer-Verlag, 1-13, 2002.

[23] Mayer, A., Nothegger, C., Chwatal, A., & Raidl, G., “Solving the Post Enrolment Course Timetabling Problem by Ant Colony Optimization,” In Proceedings of the 7th International Conference on the Practice and Theory of Automated Timetabling, 2008.

[24] Ayob, M., Jaradat, G., “Hybrid Ant Colony Systems For Course Timetabling Problems,” IEEE 2nd Conference on Data Mining and Optimization 27-28 October 2009, Selangor, Malaysia, 120-126, 2009.

[25] Jat, N. S., & Shengxiang, Y., “A Memetic Algorithm for the University Course Timetabling Problem,” IEEE 20th IEEE International Conference on Tools with Artificial Intelligence, 427-433, 2008.

[26] Alvarez, R., Crespo, E., & Tamarit, J. M., "Design and Implementation of a Course Scheduling System Using Tabu Search," European Journal of Operational Research, 137, 512-523, 2002.
[27] Aladag, C. H., Hocaoglu, G. A., & Basaran, M., "The effect of neighborhood structures on tabu search algorithm in solving course timetabling problem," Expert Systems with Application, 36, 12349–12356, 2009.
[28] Tuga, M., Berretta, R., & Mendes, A., "A Hybrid Simulated Annealing with Kempe Chain Neighborhood for the University Timetabling Problem," 6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007), 2007.
[29] Aycan, E., & Ayav, T., "Solving the Course Scheduling Problem Using Simulated Annealing," IEEE, 2008.
[30] Joudaki, M., Imani, M., & Mazhari, N., "Using improved Memetic algorithm and local search to solve University Course Timetabling Problem (UCTTP) ," Islamic Azad University, Doroud, Iran, 2010.
[31] Shengxiang, Y., & Jat, N.S., "Genetic Algorithms with Guided and Local Search Strategies for University Course Timetabling," IEEE Transactions on Systems, MAN, and Cybernetics-PART C: Applications and Reviews, Vol. 41, No. 1, 2011.
[32] Abdullah, S., Burke, E.K., & McColloum, B., "An Investigation of Variable Neighborhood Search for University Course Timetabling," In The 2nd Multidisciplinary Conference on Scheduling: Theory and Applications, NY, USA, 413-427, 2005.
[33] Abdullah, S., Burke, E.K., & McColloum, B., "Using a Randomised Iterative Improvement Algorithm with Composite Neighborhood Structures for University Course Timetabling," Metaheuristic – Program in Complex Systems Optimization, 153-172, 2007.
[34] Kostuch, P., "The University Course Timetabling Problem with a Three-Phase Approach," In Lecture Notes in Computer science, Springer-Berlin / Heidelberg, 109-125, 2005.
[35] Abdullah, S., Burke, E.K., & McColloum, B., "A Hybrid Evolutionary Approach to the University Course Timetabling Problem," In Proceedings of CEC: The IEEE Congress on Evolutionary Computation, 1764-1768, 2007.
[36] Abdullah, S., & Hamdan, R., "A Hybrid Approach for University Course Timetabling," IJCSNS International Journal of Computer Science and Network Security, Vol.8, No.8, 2008.
[37] Turabieh, H., & Abdullah, S., "Incorporating Tabu Search into Memetic Approach for Enrolment-based Course Timetabling Problems," IEEE 2nd Conference on Data Mining and Optimization 27-28 October 2009, Selangor, Malaysia, 115-119, 2009.
[38] Shahvali Kohshori, M., & Saniee Abadeh, M. "Hybrid Genetic Algorithms for University Course Timetabling," IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 2, No 2, 2012.
[39] Amintoosi, M., & Haddadnia, J., "Fuzzy C-means Clustering Algorithm to Group Students in A Course into Smaller Sections," ACM, 147–160, 2005.
[40] Rachmawati, L., & Srinivasan, D., "A Hybrid Fuzzy Evolutionary Algorithm for A Multi-Objective Resource Allocation Problem," IEEE Proceedings of the Fifth International Conference on Hybrid Intelligent Systems, 2005.
[41] Asmuni, H., Burke, E.K., & Garibaldi, J.M., "Fuzzy multiple heuristic ordering for course timetabling," The Proceedings of the 5th United Kingdom Workshop on Computational Intelligence (UKCI05), London, UK, 302-309, 2005b.
[42] Golabpour, A., Mozdorani Shirazi, H., Farahi, A., kootiani, M., & beige, H., "A fuzzy solution based on Memetic algorithms for timetabling," IEEE International Conference on MultiMedia and Information Technology, 108-110, 2008.
[43] Chaudhuri, A., & Kajal, D., "Fuzzy Genetic Heuristic for University Course Timetable Problem," Int. J. Advance. Soft Comput. Appl., Vol. 2, No. 1, ISSN 2074-8523, 2010.
[44] Shahvali Kohshori, M., Saniee Abadeh, M., & Sajedi, H., "A Fuzzy Genetic Algorithm with Local Search for University Course Timetabling," Department of Computer. Science and Research Branch, Islamic Azad University, Khouzestan,Iran, 2011.
[45] Shatnawi, S., Al -Rababah, K., & Bani-Ismail, B., "Applying a Novel Clustering Technique Based on FP- Tree to University Timetabling Problem: A Case Study," IEEE, 2010.
[46] Gaspero, L.D., Missaro, S., & Schaerf, A., "A Multi-agent Architecture for Distributed Course Timetabling," Proceedings of the 5th International Conference on the Practice and Theory of Automated Timetabling (PATAT '04), 471-474, 2004.
[47] Kaplansky, E., & Meisels, A., "Negotiation among Scheduling Agents for Distributed TimeTabling," In Submitted to the 5th International Conference on Practice and Theory of Automated Timetabling (PATAT '04), Pittsburgh, 2004.
[48] Meisels, A., & Kaplansky, E., "Scheduling Agents-Distributed TimeTabling Problems (DisTTP)," Springer- Verlog Berlin Heldelberg, LNCS 2740, 166 – 177, 2003.
[49] Pedroso, J.P., "A Multi-Agent System for Automated TimeTabling with Shared Resources," Faculdade de Ciencias da Universidade do Porto Departamento de Ciencia de Computadores Rua do Campo Alegre 8234150-180 Porto, Portugal, 2003.
[50] Kaplansky, E., Kendall, G., Meisels, A., & Hussin, N., "Distributed Examination Timetabling," Proceeding of the 5th International Conference on the Practice and Theory of Automated Timetabling (PATAT '04), 511-516, 2004.
[51] Yanga, Y., Paranjape, R., & Benedicenti, L., "An Agent Based General Solution Model For the Course TimeTabling Problem," Hakodate, Hokkaido, Japan, AAMAS'06, ACM, 2006.
[52] Yanga, Y., Paranjape, R., & Benedicenti, L., "An Examination of Mobile Agents System Evolution in the Course Scheduling Problem," Electronic and Software Systems Engineering University of Regina, Regina, Canadian, 2004.
[53] Xiang, Y., & Zhang, W., "Distributed University Timetabling with Multiply Sectioned Constraint Networks," Proceedings of the Twenty-First International FLAIRS Conference, 2008.
[54] Nandhini, & Kanmani, S., "Implementation of Class Timetabling Using Multi Agents," IEEE, 2009.
[55] Oprea, M., "MAS_UP-UCT: A Multi-Agent System for University Course Timetable Scheduling," International Journal of Computers, Communications & Control Vol. II, No. 1, 94-102, 2007.
[56] Yanga, Y., Paranjape, R., Benedicenti, L., & Reedc, N., "A System Model for University Course TimeTabling Using Mobile Agents," Multi-agent and Grid Systems – An International Journal (2), IOS Press, 267–275, 2006.
[57] Yanga, Y., & Paranjape, R., "A multi-agent system for course timetabling," Intelligent Decision Technologies. Computer Science and Artificial Intelligence, IOS Press, Vol. 5, No. 2, 113-131, 2011.

[58] Strnad, D., & Guid, N., "A Multi-Agent System for University Course TimeTabling," Applied Artificial Intelligence: An International Journal,Vol 21,Issue 2,137-153, 2007.

Hamed Babaei received the B.Sc. and M.Sc. degrees in computer engineering from University of Tabriz, Tabriz, Iran, in 2007 and 2011, respectively. He is currently teaching in

Department of Computer Engineering, Islamic Azad University of Ahar, Ahar, Iran.
His current research interests include evolutionary algorithms, Meta-heuristic approaches, Distributed data base, Distributed systems, and their applications for timetabling problems.

Amin Hadidi received the B.Sc. degrees in mechanical engineering from University of Tabriz, Iran, in 2009 and M.Sc. degrees in mechanical engineering from University of Tarbiyat Modares, Tehran, Iran, 2011, respectively. He is currently working toward the Ph.D. degree from the Department of mechanical engineering, Sahand University of Technology, Tabriz, Iran. Of course, He is currently teaching in Department of mechanical Engineering, Islamic Azad University of Ahar, Ahar, Iran.
His current research interests include BBo algorithms, Meta-heuristic approaches in mechanical engineering, and their applications for timetabling problems and mechanical engineering.