Lagrangian relaxation based on hybrid tabu search and firefly algorithm for multi-projects discrete time-cost trade-off

Mohammad Rohaninejad, M.Sc. in Industrial Engineering, MAPSA (Abdal Industrial Project Management Co.)

Reza Tavakkoli-Moghaddam, School of Industrial Engineering, University of Tehran, Tehran, Iran

Donya Delghandi , B.Sc., MAPSA(Abdal Industrial Project Management Co.)

Keywords:
Lagrangian relaxation, tabu search algorithm, firefly algorithm, time-cost trade-off, multi-project environment.

Abstract

This paper considers a particular case of the discrete time-cost trade-off in multiple project environmentregarding limited resources capacities.To formulate this problem, a new mixed-integer programming (MIP) model is presented that minimizes the sum of tardiness cost, total indirect cost and total direct cost.The continuous or discrete time-cost trade-off problem is strongly NP-hard, and finding a feasible solution for capacitated form of this problem with large sizesmakesan exponential worst-case complexity. So, a procedure based on the Lagrangian relaxation of the capacity constraints and sub-gradient optimization is proposed, and a hybrid tabu search-firefly algorithm is used to implement this approach. Finally, the computational results are provided for evaluating the performance and effectiveness of the proposed solution approaches.For small-sizedproblems, the performance of the proposed algorithm is tested against optimal solutions, and for larger problems it is compared with the best solution that provided by the CPLEX solver.

۱. Introduction

Cost effective project scheduling is an important issue in the planning process of a project. While the general scheduling of project activities is carried out before the submission of a tender, the executive project scheduling is performed before or during the realization of a project. In scheduling of projects, the project duration can be compressed (crashed) by expediting some of its activities in several ways including; increasing crew size above the normal level, working overtime, orusingalternative construction methods. The crashing alternatives come at an additional cost. This trade-off between time and cost has been studied extensively since development of the critical path method [١]. Today, the balancing between time and cost is one of the serious challenges of project managers.

The time‑cost trade-off problem has been extensively studied decades ago under the assumption of continuous time‑cost relationships [٢, ٣]. On the contrary, the literature for the case where the time–cost modes are defined at discrete points (representing distinct alternatives) is relatively recent, despite the fact that the discrete time cost modes are considered to be a more realistic model of real projects [٤]. In discrete time-cost trade-off problem (DTCTP), activity s may be executed in j_s different modes, in which each mode has an especial time and costs. In each mode, the amount of resource usage is reduced by increasing the duration of activities in a discrete way.

This paperstudies a periodic version of the DTCTP considering the resource constraints. So far some investigators (e.g., [٥] and [٦]) considered the time-cost trade-off problem with a resource constraint. However, they assumed the resources constraint for the whole projectwas constant. This condition is not real life. But the amount of availability of resources is different throughout the project life cycle. Appropriate response to this issue is periodic project planning. Also, according to the best of our knowledge, the DTCTP has not been considered in the multi-project environment. This paper considers a particular case of the DTCTP in multiple project environments, called multi-project discrete time-cost trade-off (MPDTCT) problem. By considering the problem in a multi-project environment and determining the assignment of modes to each activity, it is required that the executing period of each activity or priority of projects activities relative to each other to be determined. So, the MPDTCT determines the sequence of activities in different projects and assigns an execution mode to a activity in such a way that the cost of projects including the tardiness cost, indirect cost and direct cost are minimized.

De et al. [٧] showed that the DTCTP is strongly NP-Hard. Under above conditions, the MPDTCT problem becomes much more complicated than the DTCTP, because the MPDTCT problem has a much larger solution space than the DTCTP. Due to the high complexity of the problem, one novel hybrid meta-heuristic algorithm is developed to achieve optimal or near-optimal solutions. The proposed approach is developed based on a hierarchical combination of tabu search and firefly algorithm. This hybrid algorithm uses the rules of the Lagrangian relaxation method for solving the problem. It is very hard to find a feasible solution with a limited capacity of resources. This issue is harder for large-sized problems. Therefore, the Lagrangian relaxation method is a suitable alternative for this problem.

The literature on the DTCTP is rather sparse. In any case, exact algorithms and meta-heuristics for solving dynamic and mixed-integer programming modelshave been proposed for solving. The use of meta-heuristics has wide applications for solving the DTCTP. The genetic algorithm (GA) has been used extensively in solving time-cost or quality trade-off problem, especially in the DTCTP. Rayes and Kandil [٨] and Fallah-Mehdipour et al. [٩] applied this algorithm to solving the problem. Yang et al. [١٠] and Zhang and Li [١١] employed particle swarm optimization (PSO) for the DTCTP. The ant colony algorithm (ACO) is one of the most widely used algorithms for the DTCTP. A number of researchers (e.g., Ng and Zhang [١٢], Xiong and Kuang [١٣] and Afshar et al. [١٤]) used this algorithm for solving the problem. Anagnostopoulos and Kotsikas [١٥] proposed a novel simulated annealing (SA) algorithm, which solves the total cost minimization problem in activity networks in the case that discrete time–cost execution modes are allowed on the project activities. Sonmez and Bettemir [١٦] presented a hybrid strategy based on GA, SA, and quantum SAalgorithms for the DTCTP.

According to the best of our knowledge, the tabu search (TS) algorithm was not used for the DTCTP. The main reason could be the type of the solution space of this problem. TSis a fast and robust algorithm for problems with a permutation variable, such as sequence operation or activities. The firefly algorithm (FA) is one of the new meta-heuristic methods presented by Yang [۱۷]. This algorithm is a population-based algorithm and it mimics the social behavior of fireflies in nature. Since this algorithm is not used to solve project time and cost trade off problems. This algorithm was applied very limited and few researchers have used this algorithm to solve scheduling problems. Sayadi et al. [۱۸] and Vahedi-Nouri et al. [۱۹] used this algorithm for flow shop scheduling problems.

In the literature of the DTCTP, the use of exact methods has received little attentionbecause of the complexity of the problem. Vanhouckeet al. [۲۱] proposed dynamic programming and branch-and-bound methods, respectively. They are proposed a branch-and-bound algorithm, in which each activity starts at its normal duration at the root node of the tree. The algorithm iteratively searches the critical network and crashes the critical activities. Also Vanhoucke [۲۲] developed a new branch-and bound algorithm, in which the lower bounds are still calculated by the procedure of Demeulemeester et al. [۲۳]. Skutella [۲٤] proposed an approximate algorithm for the budget problem. To date, little research has been done to solve the problem using a Lagrangian relaxation. The Lagrangian relaxation-based heuristic method for activity on arc (AoA) networks was developed by Akkan [۲٥].

The rest of this paper is as follows. Section ۲ presents the mixed-integer programming (MIP) model. Section ۳ proposes the Lagrangian relaxation method based on tabu search and firefly algorithm. The Section ٤ provides the computation results, and finally Section ٥ outlines the conclusion and some suggestions for future studies.

۲. Notation and model formulation

In this section, a new mixed-integer programming (MIP) model is presented for the problem that has M resources and N projects. The number of activities required for project n is denoted by S_n and the total number of projects activities is denoted by S ($S = \sum_{n=۱}^{N} S_n$). The other conditions of the model are as follows:

- The resources are available at moment zero.
- The activities are planned in as soon as possible mode.
- All resources are assigned at the beginning of each activity
- The activities are not resumable.
- There is no priority for any project.
- There is no constraint for any activities in start and finish time.

The parameters of the model are as follows.

TC_n	Tardiness cost of the *n*-th project
DD_n	Delivery date of the *n*-th project
IC_n	Indirect cost of the *n*-th project for per time unit.
$DC_{n,s}^{j}$	Direct cost of the s-th activity of the *n*-th project when this activity is performed with the mode j.
$D_{n,s}^{j}$	Duration of the *s*-th activity of the *n*-th project when this activity is performed with the mode j.
$R_{n,s}^{k,j}$	The capacity needed on resource *k* to perform of the *s*-th activity of the *n*-th projectwhen this activity is performed with the mode *j*.
$L_{n,s,r}$	Lag/Lead time between the *s*-th and the *r*-th activities of the *n*-th project.
C_k^t	capacity of resource *k* in period *t*.
w_k	The set of activities that need to source *k*.
lp	Length of each period.
G	A very large positive number

The variables of the model are given below.

$y_{n,s}^{t,j}$	A binary variable that assumes value ١ if the *s*-th activity of the *n*-th project is performed with the mode *j* and in period *t*; otherwise, it is equal to ٠.
$ft_{n,s}$	Finish time of the *s*-th activity of the *n*-th project
$st_{n,s}$	Start time of the *s*-th activity of the *n*-th project
tr_n	Tardiness of the *n*-th project
fp_n	Finish time of the*n*-th project

Based on the above assumptions and notations, the MIP model is as follows:

$$\text{Min} \sum_{n=١}^{N} TC_n \,.\, tr_n + \sum_{n=١}^{N}\sum_{s=١}^{s_n}\sum_{j=١}^{j_s}\sum_{t=١}^{T} DC_{n,s}^{j} . y_{n,s}^{t,j} + \sum_{n=١}^{N} fp_n . IC_n \qquad (١)$$

s.t.

$$tr_n \geq fp_n - DD_n \qquad n = ١,..,N \qquad (٢)$$

$$fp_n \geq ft_{n,s} \qquad n = ١,..,N\,;\, s = ١,..,s_n \qquad (٣)$$

$$\sum_{j=١}^{j_s}\sum_{t=١}^{T} y_{n,s}^{t,j} = ١ \qquad n = ١,..,N\,;\, s = ١,..,s_n \qquad (٤)$$

$$ft_{n,s} \geq st_{n,s} + \sum_{j=١}^{j_s}\sum_{t=١}^{T} y_{n,s}^{t,j} . D_{n,s}^{j} \qquad n = ١,..,N\,;\, s = ١,..,s_n \qquad (٥)$$

$$st_{n,s} - lp.\,t - G\left(١ - \sum_{j=١}^{j_s} y_{n,s}^{t,j}\right) \leq ٠ \qquad n = ١,..,N\,;\, s = ١,..,s_n ;\, t = ١,..,T \qquad (٦)$$

$$st_{n,s} - \big(lp.\,(t-١)\big) + G\left(١ - \sum_{j=١}^{j_s} y_{n,s}^{t,j}\right) > ٠ \qquad n = ١,..,N\,;\, s = ١,..,s_n ;\, t = ٢,..,T \qquad (٧)$$

$$\sum_{n=١}^{N}\sum_{s=١}^{s_n}\sum_{j=١\,|\,(s,j)\in w_k}^{j_s} R_{n,s}^{k,j} . y_{n,s}^{t,j} \leq C_k^t \qquad k = ١,..,K\,;\, t = ١,..,T \qquad (٨)$$

$$st_{n,s} + \sum_{j=١}^{j_s}\sum_{t=١}^{T} y_{n,s}^{t,j} . D_{n,s}^{j} + L_{n,s,r} \leq st_{n,r} \qquad (٩)$$

$n = ١,..,N\,;\, s,r = ١,..,s_n$
If activity *s* is connected with activity *r* by finish to start mode

$$st_{n,s} + L_{n,s,r} \leq st_{n,r} \qquad (١٠)$$

$n = ١,..,N\,;\, s,r = ١,..,s_n$
If activity *s* is connected with activity *r* by start to start mode

$$st_{n,s} + L_{n,s,r} \geq st_{n,r} + \sum_{j=١}^{j_r}\sum_{t=١}^{T} y_{n,r}^{t,j} . D_{n,r}^{j} \qquad (١١)$$

$n = ١,..,N\,;\, s,r = ١,..,s_n$
If activity *s* is connected with activity *r* by start to finish mode

$$st_{n,s} + \sum_{j=١}^{j_s}\sum_{t=١}^{T} y_{n,s}^{t,j} . D_{n,s}^{j} + L_{n,s,r} \leq st_{n,r} + \sum_{j=١}^{j_r}\sum_{t=١}^{T} y_{n,r}^{t,j} . D_{n,r}^{j} \qquad (١٢)$$

$n = ١,..,N\,;\, s,r = ١,..,s_n$
If activity *s* is connected with activity *r* by finish to finish mode

$$y_{n,s}^{t,j} \in \{٠,١\} \text{ and } ft_{n,s} \geq ٠ \text{ and } st_{n,s} \geq ٠ \text{ and } tr_n \geq ٠ \text{ and } fp_n \geq ٠ \qquad (١٣)$$

$n = ١,..,N\,;\; s = ١,..,s_n;\; t = ١,..,T;\; j = ١,..,j_s$

The objective function (۱) indicates the problem's objective function, which comprises the sum of tardiness cost, indirect cost and direct costs. Constraint (۲) determines the amount of tardiness of project *n*. Constraint (۳) determines the actual project duration. The constraint (٤) responds that what is the implementation mode of each activities and what is the setup period of each activities; and constraint (٥) specify the finish time for each activities. Constraints (٦) and (٧) specify whether the activity *s* has the setup conditions in period *t*. Constraint (٨) guarantees that the capacity limitation of resources is abided by. Constraints (٩) through (١٢) are relationship constraints with different mode FS, SS, SF and FF respectively. Constraints (١٣) specify the ranges of model variables.

۳. Solution approach

Due to the complexity of solving the optimization of the MPDTCT and complexity of finding feasible solutions in large-sized problems, we propose a Lagrangian relaxation based hybrid tabu search-firefly approach to solve the MPDTCT problem. This algorithm is briefly called LR-TS-FF. The advantages of our approach are as follows:

- Our approach first converts the full problem in to an uncapacitated project continuous TCTP to fully exploit existing solution methods for uncapacitated problems.
- On the other hand for producing good quality feasible solutions, we provide a lower bound to gage its optimality.
- To find a suitable upper bound for proposed procedure a strong and rationale heuristic method is presented.
- Lagrangian relaxation based hybrid TS-FF algorithm has proven to be an effective approach to solve large optimization problems.

۳،۱. Lagrangian relaxation

The Lagrangian relaxation is a very practical technique for solving optimization problems, especiallynonlinear programming problem, integer programming problems and problems with an embedded network structure. This method converts the original problem to a simpler problem by relaxing some of the problem's constraints that are hard constraints usually. The method penalizes violations of inequality constraints using a Lagrange multiplier, which imposes a cost on violations. These added costs are used instead of the strict inequality constraints in the optimization. In practice, this relaxed problem can often be solved more easily than the original problem.

۳،۲. Lower bound (reformulation of the problem's model)

Consider the Lagrangian relaxation with respect to the capacity constraints (٨) and let Lagrange multiplier$\lambda_{k,t} \geq ۰ , k = ۱, \dots , K ; t = ۱, \dots , T$. The resulting Lagrangian problem is given by:

$$g(\lambda_{k,t}) = Min \sum_{n=۱}^{N} TC_n . tr_n + \sum_{n=۱}^{N}\sum_{s=۱}^{s_n}\sum_{j=۱}^{j_s}\sum_{t=۱}^{T} DC_{n,s}^{j} . y_{n,s}^{t,j} + \sum_{n=۱}^{N} fp_n . IC_n + \sum_{k=۱}^{K}\sum_{t=۱}^{T} \lambda_{k,t} \left[\sum_{n=۱}^{N}\sum_{s=۱}^{s_n}\sum_{j=۱}^{j_s} R_{n,s}^{k,j} . y_{n,s}^{t,j} - C_k^t \right] \quad (۱٤)$$

s.t.

Constraints (۲) to (٧) and (٩) to (١٣)

۳،۳. Hybrid tabu search-firefly algorithm

Generally, the solution space of the MPDTCT problem consists of two sub problems.

۱) Determining the setup period for each activity.

۲) Determining the amount of resources assigned for each activity or determines which modes are used to perform each activity.

In this paper, for solving the problem with these sub problems we propose a hybrid meta-heuristic algorithm based on the tabu search algorithm and firefly algorithm which is called "TS-FF" in brief. In this hybrid algorithm, the tabu search algorithm is used to improve the sub-problem of determining the setup period and the firefly algorithm is employed to improve the sub-problem of determining the amount of resources assigned for each activity. At every step (iteration) of the lagrangian method this algorithm is performed in predefined iteration. This algorithm has a duty to provide a reasonable lower bound for the lagrangian method. So, the fitness function in TS-FF is calculated according to equation (۱۴). The TS-FF algorithm is created as a result of an integrated combination of these two algorithms. This hybrid algorithm improves both variables space of the problem simultaneously. In TS-FF after each step of the tabu search algorithm in operation sequence sub problem the best neighborhood (Z^*) is selected. Then the firefly algorithm is run to determine the appropriate values of the amount of resources assigned for each activity (U^*). The firefly algorithm sets this variable to Z^*. Next the sub problem ۱ is searched again by tabu search algorithm, while the variables of sub problem ۲ is equal to U^*.Figures ۱ and ۲ illustrate the pseudo code and execution procedure of the TS-FF algorithm respectively.

Set $F^* =$ *a very large positive number*
Select an initial solution $z_۱$ *in* Z ; $(z_۱ \in Z)$ *and* $z_۱ \rightarrow U^*$
Select an initial solutions $E=\{U_{۱,۱}, U_{۱,۲}, \dots, U_{۱,e}\}$ *in* U; $(E \in U)$
For $it = ۱$: *max iteration*
    For all $o_{it,k} \in E(k = ۱, \dots, e)$ *compute objective function* $F(z^*, U_{it,k})$
    If $F(z^*, U_{it,k}) < F^*$ ***then***
        $F(z^*, U_{it,k}) \rightarrow F^*$
        $U_{it,k} \rightarrow U^*$
    End if
    Explore the sub neighborhood $V(z^*)$ *and for all* $z \in V(z^*)$
    If $F(z, U^*) < F^*$ *(if the move to z is not tabu or if the move is tabu but passes the aspiration criterion)* ***then***
        $F(z, U^*) \rightarrow F^*$
        $z \rightarrow z^*$
    Endif
    Update tabu list
    Create a set of solutions $E=\{U_{it+۱,۱}, U_{it+۱,۲}, \dots, U_{it+۱,e}\}$ *in* U; $(E \in U)$ *by firefly operators*
End for

Fig. ۱. Pseudo code of TS-FF algorithm

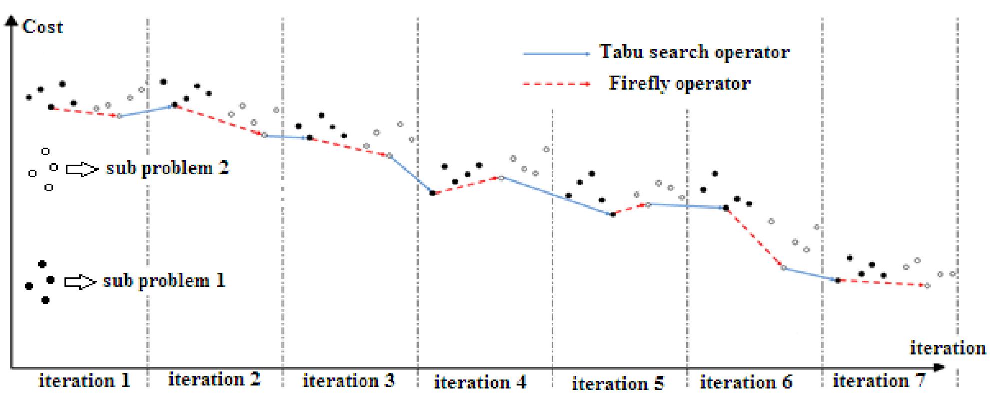


Fig. ۲. Execution procedure of TS-FF algorithm

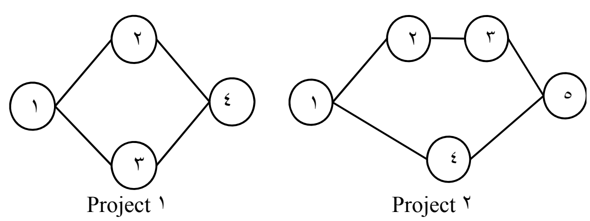
In the rest of this section, these hybrid algorithms and their components are fully described.

۳,۳,۱. Tabu search algorithm

Tabu search (TS) has been successfully applied to a large number of combinatorial optimization problems, especially in a production scheduling domain. The TS procedure is generally simple, in which an initial solution is first created and set as a current solution. Then the neighbors of the initial solution is searched by local search, fitness functions of the neighbors are computed, and the best of them that is not tabu or passes the aspiration criterion is selected and moved to it. To avoid the cycling danger, this move s added to a tabu list, and the oldest move is removed from the tabu list if it is overloaded. A new solution is stored as the current solution and its fitness function replaced with the best fitness function if it is better than the current best solution. These procedures are repeated until a stopping criterion is met.

۳,۳,۲. Encoding schema for sub problem ۱ (setup period for each activities)

A proper encoding scheme, which is indicative of the characteristics of a solution, has considerable influence on the performance of a meta-heuristic method.The encoding scheme for sub-problem ۱ is represented by a matrix with dimensions of ۲ and S, where S is the total number of activities for all projects. In this matrix, the elements of the first row include the three components (n, s), which n is the number of project and s is the number of project activity. The arrangement of the matrix's row components delineates the sequences of activity executions relative to one another. Also,the elements of the second row include the setup period number. The sequence of activities indicates their priority in the use of shared resources. An example of the encoding scheme for the problem with ۲ projects and ۹ activities is presented in Fig. ۳.



۲،۱	۱،۱	۲،۲	۲،٤	۱،۳	۱،۲	۲،۳	۲،٥	۱،٤
۱	۱	۲	۲	۱	۲	۳	۳	۳

Fig. ۳. Encoding scheme for sub-problem ۱

۳,۳,۳. Initial solution for TS

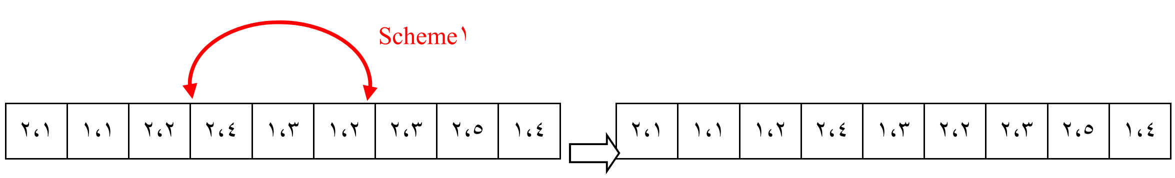
In our proposed algorithms, the initial solution has little influence on the final solution quality; however, they affect the running time. Therefore, a population of solutions is randomly generated and the best of them is selected for the initial solution.

۳,۳,٤. Neighborhood structure

The neighborhood structure used in this paper is generated by a pair-wise interchange. A neighborhood solution is generated by the following two ways:

- Scheme۱: Exchange sequence of two activities is selected randomly.
- Scheme۲: One random activity is selected and its setup period (*t*) is equal to (*t*-۱) or (*t*+۱) randomly.

Controlling and maintaining the priority terms of a project are an important principle during the neighborhood search. Fig. ٤ shows a sample of the neighborhood structures in the MPDTCT.



۱	۱	۲	۲	۱	۲	۳	۳	۳

۱	۱	۲	۲	۱	۲	۲	۳	۳

Scheme۲

Fig. ٤.Sample of the neighborhood structures in the MPDTCT

۳,۳,٥. Tabu list:

The length of the tabu list determines the remained time limit on memory for elements, and plays an important role in the search process. Nevertheless, if the list is too long the search can be inhibited, whereas if it is too short cycling cannot be avoided. In this paper, the length of the tabu list is allowed to vary dynamically during the course of the search and it is selected between $\frac{۱}{۸}S$ and $\frac{۱}{٤}S$, where S is the total number of activities for all projects. In our approach, the tabu list size changes after the every given number of iterations (b=۰,۲max iter).

- For the first *b*iterations, the length of the list is equal to $\frac{۱}{۸}$S.
- From *b*to ۲*b*iterations, the length of the list is equal to $\frac{۱}{۷}$S.
- From ۲*b*to ۳*b*iterations, the length of the list is equal to $\frac{۱}{٦}$S.
- From ۳*b*to ٤*b*iterations, the length of the list is equal to $\frac{۱}{٥}$S.
- From ٤*b*to ٥*b*iterations, the length of the list is equal to $\frac{۱}{٤}$S.

۳,۳,٦. Firefly algorithm

The firefly algorithm(FA) is a population-based algorithm and it mimics the social behavior of fireflies in nature. In this algorithm, the movement of firefly *i* toward the more attractive (i.e., brighter) firefly *j* is determined through the following relation [۲٦]:

$$x_i = x_i + \beta_{.} e^{-\gamma r_{ij}^{۲}}\left(x_j - x_i\right) + \alpha\left(rand - \frac{۱}{۲}\right) \quad (۱۵)$$

where$\beta_{.} e^{-\gamma r_{ij}^{m}}$is the attractiveness function whose value an decreases with the increase of the Euclidean distances between x_i and x_j in two fireflies (r_{ij}). $\beta_{.}$is the attractiveness at $r_{ij} = ۰$, and γ is the fixed light absorption coefficient in the environment. Expression $\alpha\left(rand - \frac{۱}{۲}\right)$ is for the randomization of movement, whereα is the randomization parameter, and "*rand*" is a function that generates random numbers with uniform distribution in the [۰, ۱] interval.

۳,۳,۷. Encoding schema for sub problem ۲ (quantity of resources assignment):

The encoding scheme for increasing the amount of per resources for each activity is represented by matrix *U* with dimensions of *K* and *S*, where *K* is the total number of resources and *S* is the total number of activities in all projects. This scheme is illustrated in Fig. ٥, where $u_{k,s}$is the amount of resource*k*assigned to activity *s*.

activities

resources

$$\begin{matrix} u_{۱۱} & u_{۱۲} & \dots & u_{۱S} \\ u_{۲۱} & u_{۲۲} & \dots & u_{۲S} \\ \vdots & \vdots & \cdots & \vdots \\ u_{K۱} & u_{K۲} & \dots & u_{KS} \end{matrix}$$

Fig. ٥. Encoding scheme for sub problem ۲

In the matrix U, if not used the resource m for Item s then $u_{k,s}$ is equal to null.

۳,۳,۸. Firefly operator

In this paper, the amount of resources assigned to activities in current solution θ is determined by:

$$U^{(\theta)} = U^{(\theta)} + \beta . e^{-\gamma r_{b\theta}^{۲}}\left(U^{(b)} - U^{(\theta)}\right) + \alpha_s - (\alpha_s - \alpha_f)(\frac{CI - ۱}{TI - ۱})\left(rand - \frac{۱}{۲}\right) \quad (۱۶)$$

In the above relation,$\alpha_s$and α_f are primal and terminal values of parameter α, respectively. Also*CI* and *TI* are the current iteration number and total iteration number of solving the process,respectively.

In ۸۰٪ of times,matrix $U^{(b)}$ is equal to the best local location, which is achieved by fireflies in the previous iteration. At other times, it is equal to the best overall location, which is achieved by fireflies in the whole of search process. In per iteration, the arrays obtained by this operator are modified as follows:

- If a firefly operator produces an array $u_{k,s}$ of the matrix $U^{(\theta)}$ while $(u_{k,s} \neq R_{n,s}^{k,j} \mid j = ۱, \dots, j_s)$ then put $u_{k,s}$equal to $R_{n,s}^{k,j_x}$ while $\{ \left|u_{k,s} - R_{n,s}^{k,j_x}\right| = \min_{j=۱}^{j_s} \left|u_{k,s} - R_{n,s}^{k,j}\right| \}$

۳,٤. Upper bound

At every step of the sub-gradient method feasibility procedure is applied to generate an upper bound for Lagrangian problem. So, after obtaining the lower bound, we propose a heuristic method to find a feasible solution at per iteration of the Lagrangian procedure. Since the proposed method the capacity constraints (۸) are relaxed. Therefore, it is necessary to reduce the use of resources by increasing the duration of activities to fit available capacity at each period. At iteration “it”, if the Lagrangian solution is not feasible for primal problem, this solution is modified using the heuristic procedure described in the next sub-section to generate a new feasible solution for a primal problem. If the value of the new feasible solution is better than the incumbent upper bound then the new value becomes the incumbent upper bound. The procedure for finding a feasible solution is summarized as follows:

Procedure for upper bound

Step ۱: set $t = ۱$ and $k = ۱$

Step ۲: Control the capacity exceeded on resource *k* in period *t*

$$\delta_k^t = C_k^t - \sum_{n=۱}^{N} \sum_{s=۱}^{s_n} \sum_{j=۱}^{j_s} R_{n,s}^{k,j} . y_{n,s}^{t,j} \quad (۱۷)$$

Step ۳: If $\delta_k^t \leq ۰$then be calculated the free float “ff” for a set of activities in period *t* that make use of resource *k*. This set is called$Q_۱$.

Step ٤: Find the activity *s* from project *n* in the $Q_۱$ with maximum free float $(ff(s) = max_{x \in Q}\{ff(x)\})$ where activity *s* is performed in mode *i*.

Step ٥: If $ff(s) > ۰$, then find a mode $j \in (۱, \dots, j_s)$

That $D_{n,s}^j - D_{n,s}^i \leq ff(s)$ and$R_{n,s}^{k,j} - R_{n,s}^{k,i} - \delta_k^t = min_{x=۱}^{j_s}\{R_{n,s}^{k,x} - R_{n,s}^{k,i} - \delta_k^t\}$

If there is no each activity with the above conditions, then go to step ۸.

Step ٦: Increase in duration of activity *s* from $D_{n,s}^i$ to $D_{n,s}^j$ and activity *s* is removed from the$Q_۱$. Also,$\delta_k^t = \delta_k^t - (R_{n,s}^{k,i} - R_{n,s}^{k,j})$and update the projects scheduling.

Step ۷: If $\delta_k^t \leq ۰$, then go to Step ٤; else, go to Step ۱٤.

Step ۸: Calculated the total float "*tf* " for a set of activities in period *t* that make use of resource *k*. This set is called$Q_۲$.

Step ۹: Find the activity s from project n in the $Q_۲$ with maximum total float $(tf(s) = max_{x\in Q}\{tf(x)\})$ where activity *s* is performed in mode *i*.

Step ۱۰: If $tf(s) > ۰$, then find a mode $j \in (۱, \dots, j_s)$

That $D_{n,s}^{j} - D_{n,s}^{i} \leq tf(s)$and $R_{n,s}^{k,j} - R_{n,s}^{k,i} - \delta_k^t = min_{x=۱}^{j_s}\{R_{n,s}^{k,x} - R_{n,s}^{k,i} - \delta_k^t\}$

If there is no each activity with the above conditions, then go to Step ۱۳.

Step ۱۱: Increase in duration of activity s from $D_{n,s}^{i}$ to $D_{n,s}^{j}$ and activity s is removed from the$Q_۲$. Also,$\delta_k^t = \delta_k^t - (R_{n,s}^{k,i} - R_{n,s}^{k,j})$and update the projects scheduling.

Step ۱۲: If $\delta_k^t \leq ۰$, then go to Step ۹; else, go to Step ۱۴.

Step ۱۳: Find the activity *s* from project *n* in period *t* with the later start time while it uses of resource *k*. Then, the activity *s* is shifted from period *t* to the *t*+ ۱. Also, $\delta_k^t = \delta_k^t - (R_{n,s}^{k,i} - R_{n,s}^{k,j})$and update the projects scheduling. Go to Step ۷.

Step ۱۴: If $k < K$, then$k = k + ۱$ go to Step ۲ else

If $t < T$then$t = t + ۱$and$k = ۱$ go to Step ۲ else

Stop the algorithm.

As indicated above steps, the proposed heuristic to generate the upper bound is based on the following three phases.

۱) Effective use of the free float (Steps ۳ to ۷).
۲) Effective use of the total float (Steps ۸ to ۱۲).
۳) Transition from one period to the next period (Step ۱۳).

۳,۵. Updating the Lagrangian multipliers

The Lagrangian multipliers λ_k are updated at per iteration using standard sub-gradient optimization by Fisher [۲۷].

$$\lambda_{k,t}^{(it+۱)} = \max\{۰, \lambda_{k,t}^{(it)} + \beta^{(it)}\left(\sum_{n=۱}^{N}\sum_{s=۱}^{s_n}\sum_{j=۱}^{j_s} R_{n,s}^{k,j}.y_{n,s}^{t,j} - C_k^t\right) \tag{۱۸}$$

$$\beta^{(it)} = \Delta_{it-۱}\frac{Z_{UB}^{*} - Z_{LB}}{\left(\sum_{k=۱}^{K}\sum_{t=۱}^{T}\lambda_{k,t}\left[\sum_{n=۱}^{N}\sum_{s=۱}^{s_n}\sum_{j=۱}^{j_s} R_{n,s}^{k,j}.y_{n,s}^{t,j} - C_k^t\right]\right)} \tag{۱۹}$$

In Eq. (۱۹), Z_{UB}^{*} is the best upper bound so far.$Z_{LB}$is the lower bound at current iteration. Also,$\Delta_{it-۱}\in (۰,۲]$and we start with $\Delta_۰ = ۲$ and multiply $\Delta_{it-۱}$ by ۰,۵ if the improvement in the Lagrangean bound (Z_{UB}^{*}) is not more than ۰,۲٪ in five consecutive iterations.

۳,۶. Stopping criteria

The Lagrangian relaxation procedure is terminated once one of the following stopping criteria is met.

- Optimality gap: The optimality gap between the lower and upper bounds is less than a defined value.
- Maximum iteration count: A defined number of iterationsis performed.
- Step size multiplier: $\lambda$is less than a defined epsilon.

۳,۷. Pseudo code of the LR-TS-FF algorithm

Fig. ٦ shows pseudo code of LR-TS-FF algorithm.

۱) *Initialize the subgradient procedure parameters*

```
For it = ١: itmax (maximum iterations number of  LR-TS-FF algorithm
        Solve the Lagrangean sub problems by hybrid tabu search-firefly algorithm
        Compute the new lower bound value
        If  the lagrangean solution is feasible for primal problem then
                Update the lower and upper bound value
        Else
                Generate a feasible solution (upper bound) for primal problem using heuristic procedure
                Compute the objective function value of the new feasible solution
                Update the lower and upper bound value
                Update the Lagrangean multipliers
        End if
End for
```

Fig. ٦. Pseudo code of the LR-TS-FF algorithm

٤. Computational results

In thissection, the performance and effectiveness of the proposed approaches and the credibility of the proposed mathematical model are evaluated and compared. The proposed MIP model is coded in the GAMS ٢٣,٤ software and solved by the CPLEX solver. The LR-TS-FFis coded by the MATLAB ٧,١ software and run on a PC with ٢,٦٦ GHz processor and ٤ G of RAM.

٤,١. Random instances generation

Sincethere is no benchmark for the MPDTCT problem under the conditions considered in this paper, it is necessary to generate random instances for the problem in order to verify the proposed mathematical model and to investigate the quality of the results obtained through the proposed meta-heuristic algorithm. Table ١ illustrates the way these instances are produced.

Table ١. Manner of producing random instances

Parameter	Notation	Produced by:
Upper bound of resource k	UR_k	Uniform distribution between [١٠٠-٩٠٠]
Available Capacity	C_k^t	(Beta distribution with $\alpha = ٢$ and $\beta = ١$) . UR_k
capacity needed to perform activities	$R_{n,s}^{k,j}$	A(١- (٠,۶۵- $(\frac{max_{j=١}^{j_s} D_{n,s}^j - D_{n,s}^j}{max_{j=١}^{j_s} D_{n,s}^j})^٢$) {A= $\frac{(\sum_{t=١}^{T} C_k^t)}{(number\ of\ activities\ that\ need\ to\ resource\ k)}$ * $(٠.٥ + (Beta\ distribution\ with\ \alpha = ٣\ and\ \beta = ١)\ \}$
Tardiness costs	TC_n	Uniform distribution between [١٠٠-۴٠٠]
Delivery date	DD_n	(Beta distribution with $\alpha = ٢$ and $\beta = ٤$) . T.lp
Indirect cost	IC_n	Uniform distribution between [١٠-٢٠]
Direct cost	$DC_{n,s}^j$	B(١- (٠,۶۵- $(\frac{max_{j=١}^{j_s} D_{n,s}^j - D_{n,s}^j}{max_{j=١}^{j_s} D_{n,s}^j})^٢$)){B= Uniform distribution [٣٠-۶٠] }
Duration of activity	$D_{n,s}^j$	C+ μ_j . C {C= Uniform distribution [۵-٢٠] ; $\mu_١ = -.٠۵; \mu_{j+١} = \mu_j + ٠.۵$ }
Lag/Lead time	$L_{n,s,r}$	Normal distribution between [$\mu = ٠$;$\delta =$٠,٣ $min_{j=١}^{j_s} D_{n,s}^j$)]

In addition, each activity is randomly assigned to a resource. Ultimately, ١٠ instances with the above assumption are generated in small-sizedproblems (STP) and large-sizedproblems (LTP), and each sample is labeled with$(\alpha:\beta:\gamma:\delta)$, which indicate the number of projects, total number of activities, number of resource and the number of periods in the planning horizon,respectively.

٤,٢. Performance evaluation

We use three criteria to evaluate the quality of the algorithm.

- The gap between the lower bound (LB) and the upper bound (UB)is calculated by:

$$GAP۱ = ۲ * \frac{UB - LB}{UB + LB} \quad (۲۰)$$

- The distances between the upper bound (UB) and the optimal solution (OP) are denoted as GAP۲, and between the lower bound (LB) and the optimal solution (OP) are denoted as GAP۳. The distances are calculated as follows:

$$GAP۲ = ۲ * \frac{UB - OP}{UB + OP} \quad (۲۱)$$

$$GAP۳ = ۲ * \frac{OP - LB}{OP + LB} \quad (۲۲)$$

٤,۳. Example

This section presents a numerical example (STP۳:۲۱:۲:۳) for better understanding of the problem. The AON networks of STP۳:۲۱:۲:۳ is shown in Fig. ٧. In this figure, the connection mode between the activities and their lag/lid are shown on the arcs, and the black nodes are start and finish nodes. In STP۳:۲۱:۲:۳, the length of each period is equal to ۳۰ time unit, and capacity of the resource number ۱ and ۲ in period ۱ are ۳٤٦ and ۲۱٦ in period ۲ are ۲٥۳ and ۲٨۹ and in period ۳ are ٤٥٤ and ۲۱۱, respectively. Other information is shown in Tables ۲ to ۳.

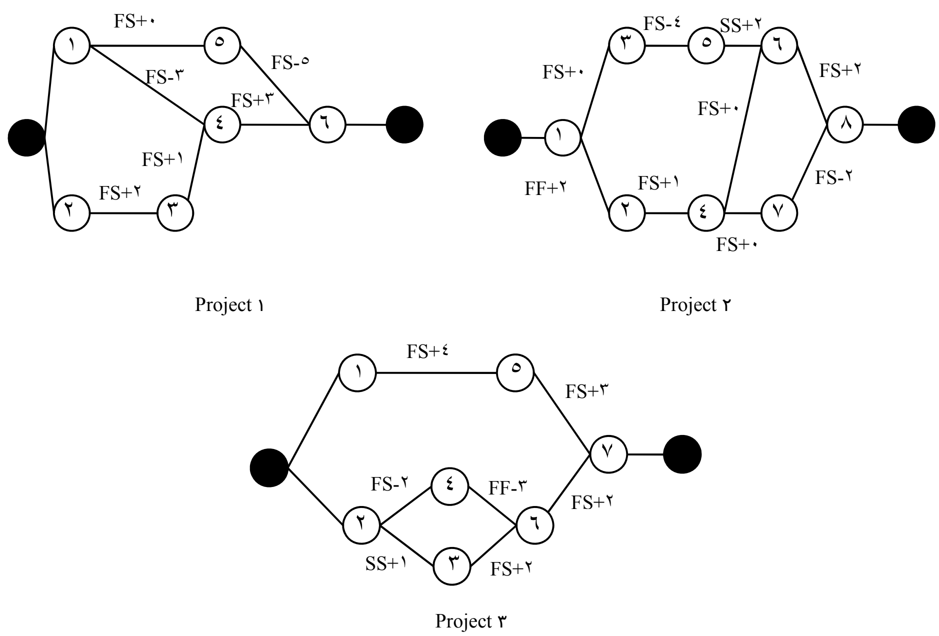


Fig. ٧. AON networks of the STP۳:۲۱:۲:۳ instance

Table۲.Parameters value for STP۳:۲۱:۲:۳ (Duration-Direct cost- Need to resource۱-Need to resource ۲)

(Project/ activity)	Duration-Direct cost- Need to resource۱-Need to resource ۲			
	Mode۱	Mode۲	Mode۳	Mode٤
(۱/۱)	٤ - ۳۱,٧ - ۱٥٥,٤ - ٨۹,۱	٨ - ۲۲,٧ - ۱۱۱,٥ - ٦۳,۹	۱۲ - ۱٦,۳ - ٨۰,۱ - ٤٥,۹	۲۰ - ۱۱,۲ - ٥٥ - ۳۱,٥
(۱/۲)	٤ - ٥٥,٤ - * - ٨٨,۱	٨ - ۳۹,٨ - * - ٦۳,۲	۱۲ - ۲٨,٦ - * - ٤٥,٤	۲۰ - ۱۹,٦ - * - ۳۱,۲
(۱/۳)	٤ - ۳٦,٦ - ۱٦٥,۳ - ٥٨,٤	٨ - ۲٦,۳ - ۱۱٨,٦ -	۱۲ - ۱٨,۹ - ٨٥,۲ -	۲۰ - ۱۳ - ٥٨,٥ - ۲۰,٧

		٤١,٩	٣٠,١	
(١/٤)	٦,٥ - ٤٥,٥ - ١٤٥,٥ - *	١٣ - ٣٢,٧ - ١٠٤,٤ - *	١٩,٥ - ٢٣,٥ - ٧٥ - *	٣٢,٥ - ١٦,١ - ٥١,٥ - *
(١/٥)	٩ - ٤٨,٥ - * - ١٠١	١٨ - ٣٤,٨ - * - ٧٢,٤	٢٧ - ٢٥ - * - ٥٢	٤٥ - ١٧,٢ - * - ٣٥,٧
(١/٦)	٣,٥ - ٣٧,٦ - ١٣٧,٦ - *	٧ - ٢٧ - ٩٨,٧ - *	١٠,٥ - ١٩,٤ - ٧٠,٩ - *	١٧,٥ - ١٣,٣ - ٤٨,٧ - *
(٢/١)	٦ - ٤٩,٥ - ١٢٠,٨ - ٨٢,٢	١٢ - ٣٥,٥ - ٨٦,٦ - ٥٨,٩	١٨ - ٢٥,٥ - ٦٢,٢ - ٤٢,٣	٣٠ - ١٧,٥ - ٤٢,٧ - ٢٩,١
(٢/٢)	٥ - ٤٢,٦ - * - ١٠١	١٠ - ٣٠,٥ - * - ٧٢,٤	١٥ - ٢١,٩ - * - ٥٢	٢٥ - ١٥,١ - * - ٣٥,٧
(٢/٣)	٩,٥ - ٤٨,٥ - ١١٧,٨ - *	١٩ - ٣٤,٨ - ٨٤,٥ - *	٢٨,٥ - ٢٥ - ٦٠,٧ - *	٤٧,٥ - ١٧,٢ - ٤١,٧ - *
(٢/٤)	٢,٥ - ٥٣,٥ - * - ٧٩,٢	٥ - ٣٨,٣ - * - ٥٦,٨	٧,٥ - ٢٧,٥ - * - ٤٠,٨	١٢,٥ - ١٨,٩ - * - ٢٨
(٢/٥)	٦,٥ - ٣٠,٧ - ١٣٩,٦ - ١٠٧,٩	١٣ - ٢٢ - ١٠٠,١ - ٧٧,٤	١٩,٥ - ١٥,٨ - ٧١,٩ - ٥٥,٦	٣٢,٥ - ١٠,٩ - ٤٩,٤ - ٣٨,٢
(٢/٦)	٦ - ٤٦,٥ - * - ٦٨,٣	١٢ - ٣٣,٤ - * - ٤٩	١٨ - ٢٤ - * - ٣٥,٢	٣٠ - ١٦,٥ - * - ٢٤,٢
(٢/٧)	٨ - ٤٠,٦ - ١٤٩,٥ - *	١٦ - ٢٩,١ - ١٠٧,٢ - *	٢٤ - ٢٠,٩ - ٧٧ - *	٤٠ - ١٤,٤ - ٥٢,٩ - *
(٢/٨)	٧ - ٥٤,٥ - ١٣٢,٧ - ١٠٣	١٤ - ٣٩,١ - ٩٥,١ - ٧٣,٨	٢١ - ٢٨,١ - ٦٨,٣ - ٥٣	٣٥ - ١٩,٣ - ٤٦,٩ - ٣٦,٤
(٣/١)	٩ - ٥٢,٥ - ١٦٠,٤ - *	١٨ - ٣٧,٦ - ١١٥ - *	٢٧ - ٢٧ - ٨٢,٦ - *	٤٥ - ١٨,٦ - ٥٦,٧ - *
(٣/٢)	٩,٥ - ٥٢,٥ - * - ٩٩	١٩ - ٣٧,٦ - * - ٧١	٢٨,٥ - ٢٧ - * - ٥١	٤٧,٥ - ١٨,٦ - * - ٣٥
(٣/٣)	٦,٥ - ٤٦,٥ - ٩٩ - ٩٩	١٣ - ٣٣,٤ - ٧١ - ٧١	١٩,٥ - ٢٤ - ٥١ - ٥١	٣٢,٥ - ١٦,٥ - ٣٥ - ٣٥
(٣/٤)	١٠ - ٥٦,٤ - ١١٥,٨ - *	٢٠ - ٤٠,٥ - ٨٣,١ - *	٣٠ - ٢٩,١ - ٥٩,٧ - *	٥٠ - ٢٠ - ٤١ - *
(٣/٥)	٩ - ٤٨,٥ - ١٥١,٥ - *	١٨ - ٣٤,٨ - ١٠٨,٦ - *	٢٧ - ٢٥ - ٧٨ - *	٤٥ - ١٧,٢ - ٥٣,٦ - *
(٣/٦)	٦,٥ - ٥٨,٤ - * - ١٠٥,٩	١٣ - ٤١,٩ - * - ٧٦	١٩,٥ - ٣٠,١ - * - ٥٤,٦	٣٢,٥ - ٢٠,٧ - * - ٣٧,٥
(٣/٧)	٣ - ٣٤,٧ - ١١٧,٨ - *	٦ - ٢٤,٩ - ٨٤,٥ - *	٩ - ١٧,٩ - ٦٠,٧ - *	١٥ - ١٢,٣ - ٤١,٧ - *

*The activity does not use the related resource.

Table٣.Parameters value for STP٣:٢١:٢:٣ (Delivery date - Indirect cost -Tardiness costs)

(Parameter)	Projects		
	Project١	Project٢	Project٣
DD_n	٥٥	٧٥	٨٠
IC_n	١٦	١٨	٢٠
TC_n	١٢٠	١٥٠	٢٠٠

The computational result for STP٣:٢١:٢:٣ is reported in Table ٤.Fig. ٨ shows the optimum solution of the STP٣:٢١:٢:٣. In this solution, the finish time of project ١ to ٣, are equal to ٧٠,٥, ٧٧,٥ and ٧٨,٥, respectively and the total cost is equal to ٦٨٥٠,٩.

Table٤. Performance oftheproposed procedure

CPU Time (S) CPLEX	CPU Time (S) LR-TS-FF	Optimal solution	Lower bound	Upper bound	GAP١ (%)	GAP٢ (%)	GAP٣ (%)
١٧,٤	٦,٢	٦٨٥٠,٩	٦٧٣٢,٧	٦٨٥٠,٩	١,٧٪	٠٪	١,٧٪

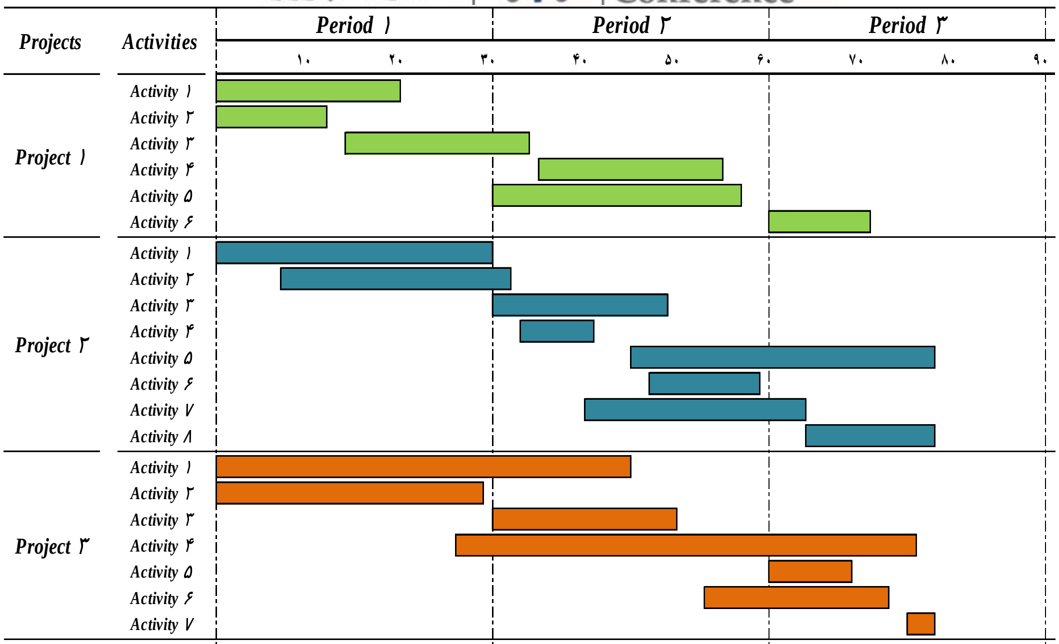


Fig. ٨.Optimum solution of the STP٣:٢١:٢:٣ test case

٤,٤. Evaluation of the performances of the proposed procedure

This section presents the comparison and performance evaluation of the proposed procedure for the considered problem. The obtained results of solving the small and large-sized instances are listed in Table ٥. Note that the blank space in this table indicates that the CPLEX solver cannot find the optimal solution in ٦٠ min. According to this table it can be found that the TS-FF hybrid algorithm enjoys a better situation compared to the CPLEX especially in large-sized instances.Regarding to the GAP criterion between the best solution that achieved by CPLEX and TS-FF upper bound, we are confident that TS-FF algorithm is able to provide optimal or most closely optimal solution in problems with ٤٠ to ٥٠ activities. Also, the proposed algorithm is superior to CPLEX in terms that it has much different run time and this algorithm can be solved the MPDTCT problem in a reasonable time. It is well known the CPLEX solver does not find a feasible solution for LTP٥:١٥٠:٤:٥, LTP٦:٢٠٠:٥:٦, LTP٨:٣٠٠:٦:٦. It proves that finding a feasible solution is difficult for the MPDTCT problem.

Table ٥.Comparison and performance evaluation of the proposed procedure in small and large-sized instance

Instances	CPLEX		LR-TS-FF					
	CPU Time (S)	Best solution	CPU Time (S)	Lower bound	Upper bound	GAP١ (%)	GAP٢ (%)	GAP٣ (%)
STP٢:١٥:٢:٢	٣,٦	٥١٢١,٣	١,٧	٥٠٥٣,٢	٥١٢١,٣	١,٣٪	٠٪	١,٣٪
STP٣:٢١:٢:٣	١٧,٤	٦٨٥٠,٩	٦,٢	٦٧٣٢,٧	٦٨٥٠,٩	١,٧٪	٠٪	١,٧٪
STP٣:٣٠:٢:٣	٣٦٨,٢	٨٧٢٤,٨	٤٥,٤	٨٥٣٧,٥	٨٧٢٤,٨	٢,١٪	٠٪	٢,١٪
STP٤:٤٠:٣:٣	٢١٥٦,٨	١١٧٨٢,٧	١٦٢,١	١١٤٢٥,٢	١١٩٣١,٥	٤,٣٪	١,٢٪	٣٪
STP٤:٥٠:٣:٤	>٣٦٠٠	١٧٤٥٢,٣	٢٩١,٢	١٤١٦٩,٨	١٥١٢١,٨	٦,٥٪		
LTP٤:٧٠:٣:٤	>٣٦٠٠	٣٤٧٠٥,٦	٩٥٣,٦	٢٠٣١٢,٣	٢١٧٤٢,٢	٦,٨٪		
LTP٥:١٠٠:٤:٥	>٣٦٠٠	٨٠٧٢٤,٨	١٤٨٨,٦	٣٠١٥٦,٣	٣٢٤٠٨,٦	٧,٢٪		
LTP٥:١٥٠:٤:٥	>٣٦٠٠	*	٣١٢٠,٨	٤٣٧٥٢,٥	٤٨٢٦١,٢	٩,٨٪		
LTP٦:٢٠٠:٥:٦	>٣٦٠٠	*	٣٦٠٠	٦٤٠٧٦,٢	٧١٨٢٢,٤	١١,٤٪		

LTP٨:٣٠٠:٦:٦	>٣٦٠٠	*	٣٦٠٠	١١٤٩١٩,٨	١٣٢٤٨٥,٦	١٤,٢٪

*A feasible solution was not found

٥. Conclusion

This paper addressed a special version of the project discrete time-cost trade-off problem (DTCTP) in a multiple project environment regarding limited resources capacities. To formulate this problem, a new mixed-integer programming (MIP) model was presented. Then, the mathematical properties of a Lagrangean relaxation of the model were developed by a Lagrangean meta-heuristic procedure that exploits these properties presented. This procedure works based on tabu search and firefly algorithm. Also, an efficient and effective heuristic algorithm is proposed to find the good upper bound. The computational results indicate the combination of Lagrangean and meta-heuristic procedure appears to be efficient and robust for solving the problem, especially for large real-life problems.

Reference

[١] Vanhoucke, M., Debels, D. (٢٠٠٧). The discrete time/cost trade off problem: Extensions and heuristic procedures. Journal of Scheduling, ١٠, ٣١١–٣٢٦.

[٢] Fulkerson, D.R. (١٩٦١). A network flow computation for project cost curve. Management Science, ٧ (٢) ١٦٧-١٧٨.

[٣] Falk,J.E.,Horowitz,J.L.(١٩٧٢).Critical path problems with concave cost-time curves, Management Science, ١٩ (٤) ٤٤٦–٤٥٥.

[٤] Anagnostopoulos,K.P.,Kotsikas, L.(٢٠١٠).Experimental evaluation of simulated annealing algorithms for the time-cost trade-off problem, Applied Mathematics and Computation, ٢٦٠–٢٧٠.

[٥] OyaIcmeli.,S., Erenguc,S. (١٩٩٦). The resource constrained time/cost tradeoff project scheduling problem with discounted cash flows, Journal of Ope.Manag, ٢٥٥–٢٧٥.

[٦] Wuliang, P.,Chengen, W. (٢٠٠٩).A multi-mode resource-constrained discrete time–cost tradeoff problem and its genetic algorithm based solution, International Journal of Project Management, ٢٧ (٦) ٦٠٠–٦٠٩.

[٧] Dunne, E.J.,De P., Ghosh J.B., Wells, C.E. (١٩٩٧). Complexity of the discrete time/cost trade-off problem for project networks. Operations Research, ٤٥, ٣٠٢–٣٠٦.

[٨] El-Rayes, K., &Kandil, A. (٢٠٠٥). Time-cost-quality trade-off analysis for highway construction. Journal of Construction Engineering and Management, ١٣١(٤),٤٧٧–٤٨٦.

[٩] Fallah-Mehdipour, E., Bozorg Haddad, O., Tabari, M.R., Mariño, M.A. (٢٠١٢).Extraction of decision alternatives in construction management projects:Application and adaptation of NSGA-II and MOPSO. Expert Systems with Applications, ٣٩, ٢٧٩٤–٢٨٠٣.

[١٠] Yang, X., Yuan, J., Yuan, J., Mao, H. (٢٠٠٧). A modified particle swarm optimizer with dynamic adaptation. Applied Mathematics and Computation, ١٨٩،١٢٠٥–١٢١٣.

[١١] Zhang, H., Li, H. (٢٠١٠). Multi-objective particle swarm optimization forconstruction time-cost tradeoff problems. Construction Management and Economics, ٢٨(١), ٧٥–٨٨.

[١٢] Ng, S. T., Zhang, Y. (٢٠٠٨). Optimizing construction time and cost using ant colony optimization approach. Journal of Construction Engineering and Management, ١٣٤(٩), ٧٢١–٧٢٨.

[١٣] Xiong, Y., Kuang, Y. (٢٠٠٨). Applying an ant colony optimization algorithm-based multiobjective approach for time-cost trade-off. Journal of Construction Engineering and Management, ١٣٤(٢), ١٥٣–١٥٦.

[١٤] Afshar, A., Ziaraty, A. K., Kaveh, A., Sharifi, F. (٢٠٠٩). Nondominated archiving multicolony ant algorithm in time-cost trade-off optimization. Journal of Construction Engineering and Management, ٦٦٨-٦٧٤.

[١٥] AnagnostopoulosL, K.P., Kotsikas.١ September ٢٠١٠. Experimental evaluation of simulated annealing algorithms for the time–cost trade-off problem, App.Math.Com, ٢١٧:٢٦٠–٢٧٠.

[١٦] RifatSonmez, ÖnderHalisBettemir.(٢٠١٢).A hybrid genetic algorithm for the discrete time-cost trade-off problem, Expert Systems with Applications, ٣٩, ١١٤٢٨–١١٤٣٤.
[١٧] Yang,X.S. (٢٠٠٨).Nature-inspired metaheuristic algorithms, Luniver Press
[١٨] Sayadi, M.K.,Ramezanian, R.,Ghaffari-Nasab, N.(٢٠١٠) A discrete firefly meta-heuristic with local search for makespan minimization in permutation flow shop scheduling problems, Int. J. of Industrial Engineering Computations, ١, ١–١٠.
[١٩] VahediNouri,B.,Fattahi, P., Ramezanian,R.(٢٠١٣). Hybrid firefly-simulated annealing algorithm for the flow shop problem with learning effects and flexible maintenance activities, Int. J. Prod. Res., DOI:١٠,١٠٨٠/٠٠٢٠٧٥٤٣,٢٠١٢,٧٥٠٧٧١.
[٢٠] Demeulemeester, E., Herroelen, W., Elmaghraby, S.E. (١٩٩٦)Optimal procedures for the discrete time/cost trade-off problem in project networks. European Journal of Operational Research, ٥٠–٦٨.
[٢١] Vanhoucke, M.,Demeulemeester,E.,Herroelen,W. (٢٠٠٢) Discrete time/cost trade-offs in project scheduling with time-switch constraints, Journal of the Op.Res.So, ٥٣, ٧٤١–٧٥١.
[٢٢] Vanhoucke, M. (٢٠٠٥) New computational results for the discrete time/cost trade-off problem with time-switch constraints, European Journal of Operational Research, ١٦٥ (٢) ٣٥٩–٣٧٤.
[٢٣] Demeulemeester, E.,De Reyck, B.,Foubert, B.,Herroelen,W., Vanhoucke,M. (١٩٩٨) New computational results for the discrete time/cost trade-off problem in project networks, Journal of the Operational Research Society, ٤٩, ١١٥٣–١١٦٣.
[٢٤] Skutella, M. (١٩٩٨)Approximation algorithms for the discrete time–cost trade-off problem. Mathematics of Operations Research, ٢٣, ١٩٥–٢٠٣.
[٢٥] Akkan, C. (١٩٩٨)A Lagrangian heuristic for the discrete time–cost tradeoff problem for activity-on-arc project networks, Working Paper, Koc University, Istanbul.
[٢٦] Yang, X.-S. (٢٠٠٩) Firefly algorithms for multimodal optimization, in: Stochastic Algorithms: Foundations and Applications, SAGA, Lecture Notes in Computer Sciences, ٥٧٩٢, ١٦٩–١٧٨.
[٢٧] Fisher, M.L. (٢٠٠٤)Thelagrangian relaxation method for solving integer programming problems. Management Science, ٥٠ (١٢) ١٨٦١–١٨٧١.