

DOI: 10.7596/taksad.v5i3.571

The optimization of the time-cost tradeoff problem in projects with conditional activities using of the multi-objective charged system search algorithm (SMOCSS)

M.K. Sharbatdar¹, S. Talatahari², M.R. Mousavi³

Abstract

The appropriate planning and scheduling for reaching the project goals in the most economical way is the very basic issue of the project management. As in each project, the project manager must determine the required activities for the implementation of the project and select the best option in the implementation of each of the activities, in a way that the least final cost and time of the project is achieved. Considering the number of activities and selecting options for each of the activities, usually the selection has not one unique solution, but it consists of a set of solutions that are not preferred to each other and are known as Pareto solutions. On the other hand, in some actual projects, there are activities that their implementation options depend on the implementation of the prerequisite activity and are not applicable using all the implementation options, and even in some cases the implementation or the non-implementation of some activities are also dependent on the prerequisite activity implementation. These projects can be introduced as conditional projects. Much researchs have been conducted for acquiring Pareto solution set, using different methods and algorithms, but in all the done tasks the time-cost optimization of conditional projects is not considered. Thus, in the present study the concept of conditional network is defined along with some practical examples, then an appropriate way to illustrate these networks and suitable time-cost formulation of these are presented. Finally, for some instances of conditional activity networks, conditional project time-cost optimization conducted multi-objectively using known meta-heuristic algorithms such as multi-objective genetic algorithm, multi-objective particle swarm algorithm and multi-objective charged system search algorithm.

Keywords: Time-cost tradeoff problem (TCTP), multi-objective optimization, charged system search algorithm (CSS), project management, simple multi-objective charged system search (SMOCSS).

¹ Author for correspondence. Faculty of Civil Engineering, Semnan University, Semnan, Iran, msharbatdar@semnan.ac.ir

² Engineering, University of Tabriz, Tabriz, Iran Department of Civil, talatahari@tabrizu.ac.ir

³ Faculty of Civil Engineering, Semnan University, Semnan, Iran, rm_Mousavi@semnan.ac.ir

1. Introduction

One of the main issues that the project manager must define is the planning and the scheduling of the project in order to achieve project goals (services, designs, programs, etc.) in the most economical way using the accurate and suitable resources and facilities (human resource, machines, materials, budget, programs, policies, etc.). In other words, project managers are looking for delivering projects in the least possible time and cost. These two objectives are in conflict with each other, as decreasing project time requires more resources and as a result spending more cost. Given the variety of implementation options for each activity, there are several options for completing the project that are not preferred to each other and are known as Pareto solutions (Hindelang, T.J., et al 1979. Elmaghraby, S.E. 1993. Feng, C.W., et al 1997). On the other hand, studying the planning methods and examining the actual projects at hand in construction, industry, etc. we confront with other activities in the project activity network that their implementation or non-implementation depend on its prerequisite activity and the implementation of an activity depends on the prerequisite activity implementation and are not applicable using all the implementation options. For example, in the stage of plastering joinery of the building if a prepared covering wall (such as Knauf) is used, whitewash step can be eliminated. In such projects, while the project manager has the authority of the implementation or non-implementation of an activity and the method of implementation, he is not able to select the appropriate process accurately in the planning stage and cannot draw a precise CPM network of activities and the absolute run time for the project. Then, the project manager, based on the time-cost optimization chooses whether an activity should be implemented or eliminated, thus in such a case, drawing conditional network of activities is essential in planning stage.

In the present research, the concept of conditional network is defined along with some practical examples and then a proper solution is provided for the illustration of these networks and appropriate time-cost formulation. Finally, for some instances of conditional activity networks, conditional project time-cost are optimized using three known meta-heuristic algorithms, also known as multi-objective genetic algorithm (Zheng, D.X.M., et al, 2004. Daisy X.M., et al, 2005), multi-objective particle swarm optimization algorithm (Coello Coello, C.A., et al, 2004. Carlos A., et al, 2002) and multi-objective charged particle system algorithm (Kaveh A., et al, 2010 (a, b, c). Sharbatdar M.K., et al, 2016). The results showed that all of the algorithms are capable of finding all of the dominant possible solutions, but in

terms of relative implementation time, multi-objective charged system search (SMOCSS) algorithm is better than the other algorithms applied in this paper.

2. Conditional activity network

The main factor for the creation of a network is providing the features and specifications of substances, tools, materials that the project manager has the authority of selecting it. In other words, in conditional projects the project manager should consider relating conditional statements to the existence or nonexistence of a conditional activity and the practical options of each conditional activity of the project appropriately and optimally in terms of time and cost. Thus, he should consider that the existence or nonexistence of a special option in one activity and even the existence or nonexistence of an activity depends on the implementation of its prerequisite activity (or even its prior activities). For the further clarification of conditional networks concept, we continue the discussion providing two examples of actual projects with this subject.

Example 1: Industrial scientific projects

In advanced scientific projects the phenomenon of shielding against harmful radiations and harmful substances and is very important and plays a vital role. In one special kind of shield metal plates or high-density sheets of glass or a type of prepared fiber can be used and in the junction stage for metal plates a special adhesive in special condition or screws and rivets with special technology or special welding can be used, while for glass sheets welding technology is not possible, also in prepared fiber the junction stage is eliminated. Thus, the project manager will illustrate a network for the time-cost optimization of these activities a part of which is presented in fig.1.

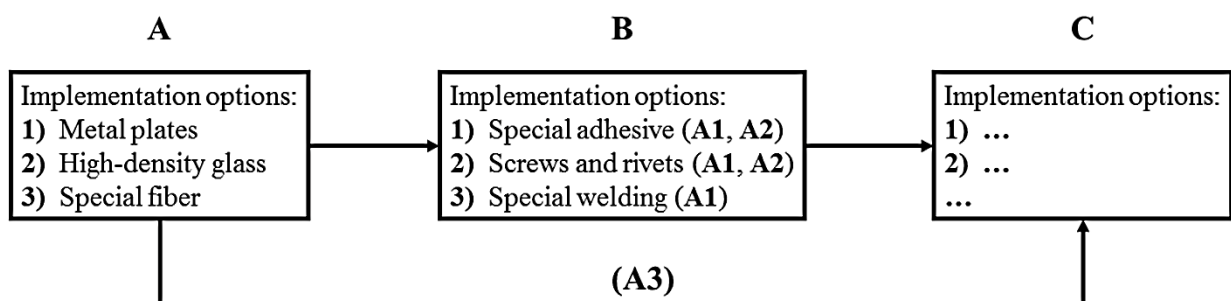


Fig.1: a part of conditional activity network related to the industrial scientific projects example.

Example 2: Construction projects

In the construction projects many cases can be found in which illustrating project activities in planning stage leads to a conditional activity network. Among them are the following

(Butterfield. D. 2011. McCabe. G.B., et al, 2003):

- In most of the buildings, first, the wall is refinished (plaster and whitewash) and then it is decorated (painting, etc.). But using wall covering as one of the implementation options in the refinish (activity) stage can lead to the elimination of decorating activity.
- Using Knauf in the activities related to the wall will eliminate refinish activities.
- In most kitchens in the refinishing stage tiling is performed and then, for decorating cabinets are used, but using wall covering and MDF cabinets, tiling is removed.
- The use of false ceiling in the refinishing stage of the ceiling leads to the elimination of plastering, whitewashing and painting.
- In large scale projects the selection of one special implementation option can even affect a subnetwork. For example, the activities of a girder skeleton building are different from the activities related to the implementation of piles and blocks.

3. Illustrating conditional activity networks

In the start of the planning of a project it is essential to define the tasks or activities that should be performed in that project and their dependencies must be determined. Thus, network display of a project is one of the primary steps in planning and is the foundation and the basis of other planning activities. In order to prevent any vagueness in the illustration of conditional activity networks, we provide some rules as follows, which stem in the concepts of conditional activity networks:

Rule 1): As in the activity on nodes (AON) networks (Levin. R. 1980), activities are displayed as nodes with solid lines (in the form of square or rectangle or circle or oval) and the connections and dependencies in between are displayed using vectors with solid lines (straight or broken or curved lines). In the nodes, each of which representing a special activity, a number or Latin letter(s) is written as an attribute of that activity.

Rule 2): The activity in the starting of the vector is the prerequisite of the activity in the ending of the vector.

Rule 3): Each network has only one starting activity and one ending activity.

Rule 4): The existence of rings in conditional activity networks is not acceptable.

Rule 5): Conditional activities are activities that their implementation affects the implementation options of at least one of the network activities or their implementation options are affected by the implementation of at least one of the prior activities. Conditional

activities are shown with knots with dotted lines. Thus, in a conditional activity network there are at least two knots with dotted lines.

Rule 6): As discussed in the previous sections a conditional activity can affect its next activity (activities) in two ways:

First effect: leads to the elimination of some of their implementation options.

Second effect: leads to the elimination of one or more of the next activities.

To show the first effect method, the related number of each of the implementation options of the conditional activity is written in the knot and in front of each, the implementation options of the prerequisite activity that eliminates those implementation options are written. For example, see fig.2:

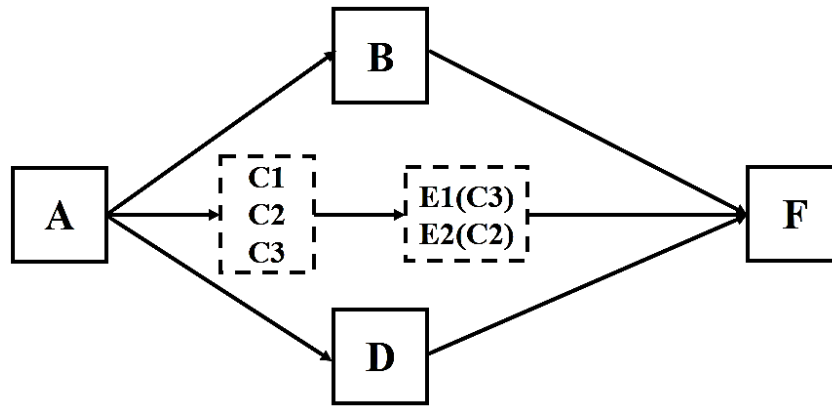


Fig.2: illustrating conditional network with two conditional activities.

According to the mentioned rules, it is observed that in the fig.2 C activity with three implementation options and E activity with two implementation options are conditional activities, in a way that the implementation of C activity creates limitations in the implementation of E activity and the first implementation option of E activity (E1) is practical when C activity is implemented with its first or second implementation option and if C activity is implemented with its third implementation option, E activity is not practical with its first implementation option and only one implementation option (E2) will be available. So, if C activity is implemented with its second implementation option, E activity cannot be implemented with its second implementation option and only one implementation option will be available (E1).

In order to show the second effect, like the first effect method, implementation options of the conditional activities and the implementation options affecting the prerequisite activity are written in the knot. For example, suppose activity network of fig.3.

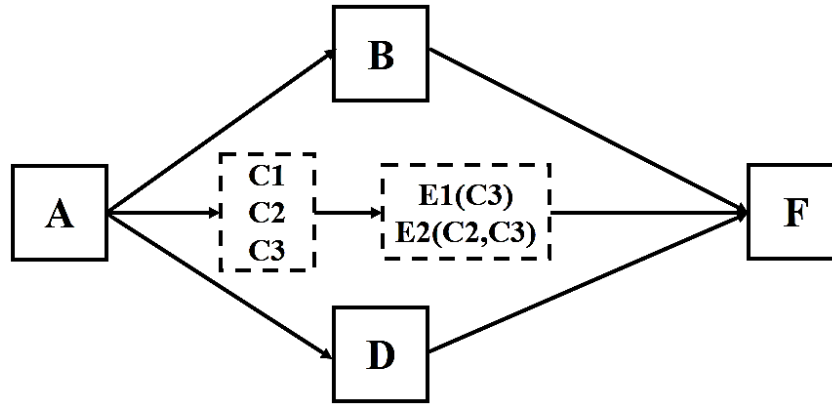


Fig.3: the implementation method of the prerequisite activity can lead to the elimination of a number of implementation options of an activity or even the elimination of that activity.

According to the conditional activity network of fig.3, A, B, C, D, F activities are implementable with all their implementation options, but E activity has conditional limitations in the implementation method and is depended on the implementation of C activity. In the implementation method of E activity, we are confronted with the following conditions:

- If C activity is implemented with its first implementation option (C1), E activity is implementable with both of its implementation options (E1, E2).
- If C activity is implemented with its second implementation option (C2), E activity is implementable only with its first implementation option (E1).
- If C activity is implemented with its third implementation option (C3), E activity is not implementable with any of its implementation options (E1, E2) and actually E activity is eliminated.

4. The time-cost formulation of conditional activity networks

As the time-cost formulation in TCTP issues (Feng, C.W., et al, 1997. Zheng, D.X.M., et al, 2004. Daisy X.M., et al, 2005) does not include the effect of the method of implementation of an activity on other activities and their implementation options in unconditional projects, this formulation cannot be used directly in conditional projects and must be revised. In order to better understand the conditional activities network, suppose fig.4 as a hypothetical example that its implementation activities are shown in table 1.

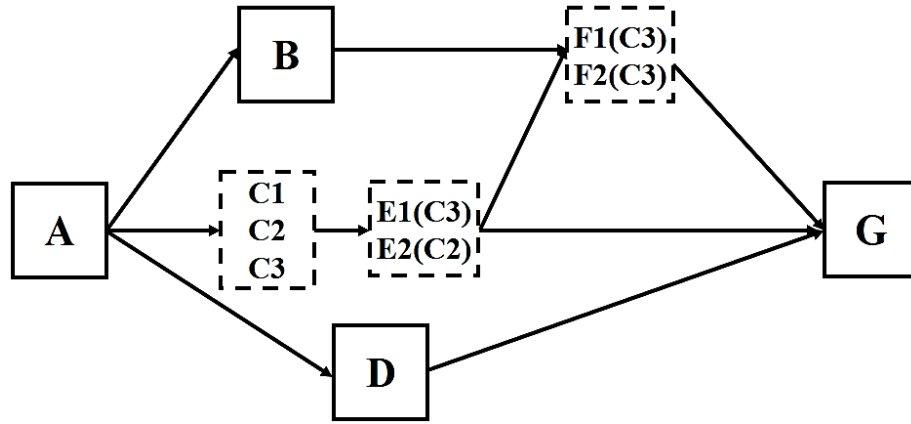


Fig.4: conditional activity network with three conditional activities.

Table 1: implementation options of the conditional activity network fig.4.

No	Pr	Op	Ti	Cost	No	Pr	Op	Ti	Cost
A	-	1	7	1150	D	A	1	6	2250
		2	10	900			2	8	1750
		3	12	600			3	10	1500
B	A	1	12	330	E	C	1	5	370
		2	14	270			2	8	240
		3	16	220	F	B	1	2	450
		4	18	180			2	3	350
C	A	1	5	1500	G	D	1	7	2000
		2	7	1350			2	9	1600
		3	11	1450			3	12	1100

In the activity network of the above example for each of the activities one of its implementation options cannot be selected and time and cost of the path cannot be calculated as some of the paths are not practical due to their nature and activity network and their conditional inheritance, that the following items can be mentioned as an example:

$$\text{I) } A1 \rightarrow \begin{Bmatrix} B3 \\ C2 \\ D3 \end{Bmatrix} \rightarrow E2 \rightarrow F1 \rightarrow G3$$

$$\text{II) } A3 \rightarrow \begin{Bmatrix} B4 \\ C3 \\ D1 \end{Bmatrix} \rightarrow E1 \rightarrow F2 \rightarrow G1$$

Existence of such paths in the conditional activity networks causes the complexity of time-cost trade off problem, in a way that the non-recognition of those leads to the non-recognition of optimized path set and even in some cases leads to the elimination of some of the optimal practical solutions. So, the first step in time-cost optimization of conditional activity networks, is the elimination of unauthorized paths from all of the implementation paths of the

project and determining authorized paths. For this purpose, we define mu parameter for each of the activities Contractual as follows:

First contract: For all of the unconditional activities and the activities that their implementation options are not derived from the previous activities' implementation this parameter equals to 1 (one). For example, in the conditional activity network of fig.4:

$$mu(A) = mu(B) = mu(C) = mu(D) = mu(G) = 1 \quad (1)$$

Second contract: For the conditional activities considering the conditionality of the activity, this parameter equals 1 for authorized modes and for unauthorized ones it equals zero. For example, in the conditional activity network of fig.4:

$$\left\{ \begin{array}{l} \text{if } C1 \text{ or } C2 \& E1 \text{ or } C3 \& E2 \Rightarrow mu(E) = 1 \\ \quad \text{else } mu(E) = 0 \\ , \\ \text{if } C3 \Rightarrow mu(F) = 0 \\ \quad \text{else } mu(F) = 1 \end{array} \right. \quad (2)$$

By determine the parameter value of mu for each of the activities, authorized and unauthorized paths can be determined easily. For this purpose and for the simplicity the next parameter is defined as Mu as the product of mu values for all of the activities:

$$Mu = \prod mu \quad (3)$$

Finally, a path is authorized that its Mu value equals 1, otherwise, the path is unauthorized and should be eliminated from the project implementation path set. The efficiency of the introduced parameters are observable in the paths below for the conditional activity network of fig.4.

a) Example of unauthorized path:

$$A1 \rightarrow \left\{ \begin{array}{l} B3 \\ C2 \\ D3 \end{array} \right\} \rightarrow E2 \rightarrow F1 \rightarrow G3 \rightarrow Mu = 1 \times 1 \times 1 \times 1 \times 0 \times 1 \times 1 = 0$$

b) Example of unauthorized path:

$$A3 \rightarrow \left\{ \begin{array}{l} B4 \\ C3 \\ D1 \end{array} \right\} \rightarrow E1 \rightarrow F2 \rightarrow G1 \rightarrow Mu = 1 \times 1 \times 1 \times 1 \times 1 \times 0 \times 1 = 0$$

c) Example of authorized path

$$A2 \rightarrow \left\{ \begin{matrix} B1 \\ C1 \\ D2 \end{matrix} \right\} \rightarrow E2 \rightarrow F2 \rightarrow G2 \quad \rightarrow \quad Mu = 1 \times 1 \times 1 \times 1 \times 1 \times 1 \times 1 = 1.$$

Now, by solving the problem of eliminating the unauthorized paths from the implementation paths of the conditional activity network, the appropriate formulation of time-cost optimization for these networks could be described as follows:

$$\begin{cases} \min T \\ \min C \end{cases} \quad (4)$$

where T parameter shows total time and C parameter shows total project implementation cost and is defined as follows:

$$\begin{cases} T = \max_{L_k \in L} \left[\sum_{i \in L_k} t_i^{(k)} x_i^{(k)} \right] \\ C = \sum_{i \in A} dc_i^{(k)} x_i^{(k)} + T \times ic_i^{(k)} \end{cases} \quad (5)$$

where,

$t_i^{(k)}$: shows the time of i activity in the implementation mode with its k th implementation method.

$x_i^{(k)}$: is the 0, 1 variable indicator of i activity and $x_i^{(k)}=1$ if the i th activity is implemented with its k th implementation method and $x_i^{(k)}=0$ if the i th activity is not implemented with its k th implementation method. Thus the total indicator of each of the activities' variable equals 1.

L_k : indicates the sequence of the activities in the k th path and $L=\{ L_k | k=1, 2, \dots, m \}$, in which m represents the number of authorized paths in project network and L is the set of authorized paths of project implementation in which $Mu=1$.

A : activities implemented in the authorized project paths.

$dc_i^{(k)}$: represents the direct cost of the i th activity in the k th implementation method.

$ic_i^{(k)}$: represents the indirect cost in time unit of the i th activity in the k th implementation method.

Presenting the results of time-cost optimization of conditional activity networks

The conditional activity networks are introduced in the previous section and their necessity was discussed providing some examples, also the illustration method of such projects and their time-cost formulation was presented. In this section, some instances of conditional activity networks are solved using heuristic algorithms and the results are

compared in terms of implementation time and the number of Pareto solutions, as in the previous section.

Example 1:

The first example designed for this case is the example of fig.4 which includes three conditional activities in which A, B, C, D, G activities are practical with all of their implementation options, but E and F activities have conditional limitations as follows:

- If C activity is implemented with its first implementation option (C1), E and F activities are practical with all of their modes.
- If C activity is implemented with its second implementation option (C2), F activity is practical with all of its modes, but E activity is only implementable with its first mode (E1) and selecting the implementation option of E2 is unauthorized.
- If C activity is implemented with its third implementation option (C3), E activity is implementable with its second mode (E2) and E1 implementation option is unauthorized. Also, F activity is not practical in any of its modes, i. e. this activity will be eliminated.

This example is solved with three meta-heuristic algorithms of NSGA-II, MOPSO and SMOCSS and the results were compared in terms of relative implementation time and the number of solutions in the dominant solution set. As it is presented in table 2 all of these algorithms are capable of finding all of the possible dominant solutions in all of three discussed examples in this section, but in terms of relative implementation time SMOCSS algorithm is much better than the other old algorithms.

Table 2: the number of dominant solutions and the relative implementation time of the algorithms in solving conditional activity networks.

Network	the number of dominant solutions			Run time Run time of SMOCSS algorithm		
	NSGA-II	MOPSO	SMOCSS	NSGA-II	MOPSO	SMOCSS
Example 1	14	14	14	1.07	1.04	1
Example 2	17	17	17	1.11	1.08	1
Example 3	15	15	15	1.14	1.10	1

Pareto solution set resulted from SMOCSS algorithm of the example 1 is presented in fig.5 and table 3.

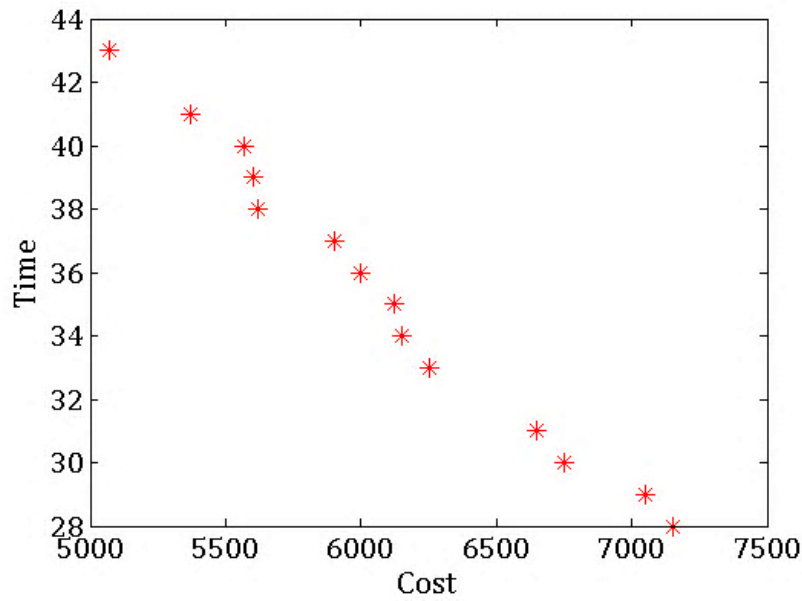


Fig.5: Pareto set for the conditional activity network of the example 1.

Table 3: the values of time and cost in the Pareto set of the conditional activity network of the example 1.

No.	Time	Cost
1	28	7150
2	29	7050
3	30	6750
4	31	6650
5	33	6250
6	34	6150
7	35	6120
8	36	6000
9	37	5900
10	38	5620
11	39	5600
12	40	5570
13	41	5370
14	43	5070

Example 2:

The second example of conditional activity network, which is presented in fig.6, is important in terms of the simultaneous selection of two implementation options of two independent activities which leads to the elimination of one (or more) of the implementation options of the next activities. For example, if D activity is implemented with its second implementation option (D2) and F activity is implemented with its first implementation option (F1), G activity is not implementable with its first implementation option (G1). In time-cost optimization of these networks, paths which are not complying with the conditional rules are eliminated from the authorized paths of the project.

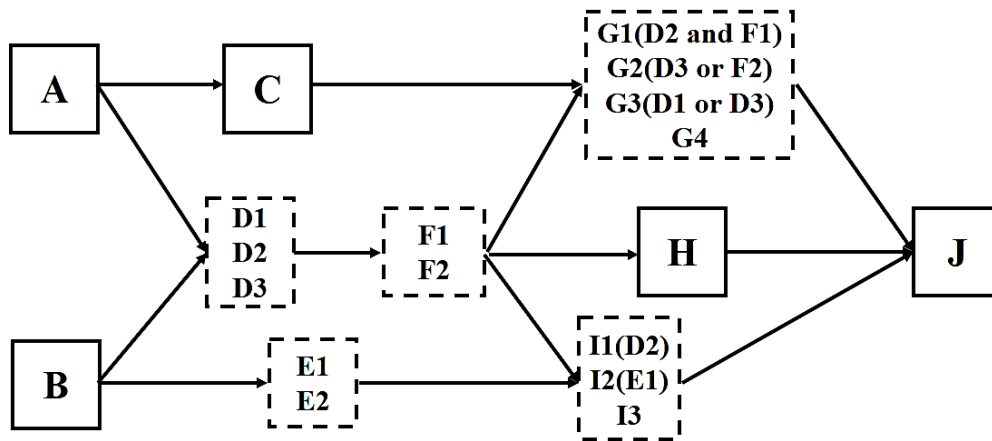


Fig.6: The example 2, conditional activity network with five conditional activities.

As it is observable from fig.6, in the conditional activity network of the example 2, A, B, C, D, E, F, H, J activities are implementable with all of their implementation options, but G and I activities have limitations as follows:

- If D activity is implemented with its second implementation option (D2) and F activity is implemented with its first implementation option (F1), G activity is not implementable with its first implementation option (G1).
- If D activity is implemented with its third implementation option (D3) or F activity is implemented with its second implementation option (F2), G activity is not implementable with its second implementation option (G2).
- If D activity is implemented with its first or third implementation options (D1 or D3) G activity is not implementable with its third implementation option (G3).
- If D activity is implemented with its second implementation options (D2), I activity is not implementable with its first implementation option (I1).
- If E activity is implemented with its first implementation options (E1) I activity is not implementable with its second implementation option (I2).

According to the mentioned explanations about the limitations in the method of finishing the project related to the second example, selecting some of the project implementation paths is not possible and in the optimization process these paths are eliminated. Implementation options of the activities related to this example is presented in table 4.

Table 4: implementation options of the conditional activity network of the example 2.

No	Pr	Op	Ti	Cost	No	Pr	Op	Ti	Cost
A	-	1	15	37000	F	D	1	23	48000
		2	19	31500			2	28	35000
		3	25	28000	G	F	1	16	23000
B	-	1	12	15500			2	18	19000
		2	15	12500			3	21	15000
C	A	1	20	61000			4	24	11000

		2	21	53000	H	F	1	12	31000
		3	26	40000			2	17	25000
		4	29	34000			3	24	17000
D	A B	1	7	9000	I	E F	1	2	7000
		2	10	7000			2	3	5500
		3	12	5000			3	4	4000
E	B	1	11	16000	J	G H I	1	12	13000
		2	14	10000			2	14	11000

Pareto solution set for the conditional activity network of the second example resulted from SMOCSS algorithm is presented in fig.7 and table 5.

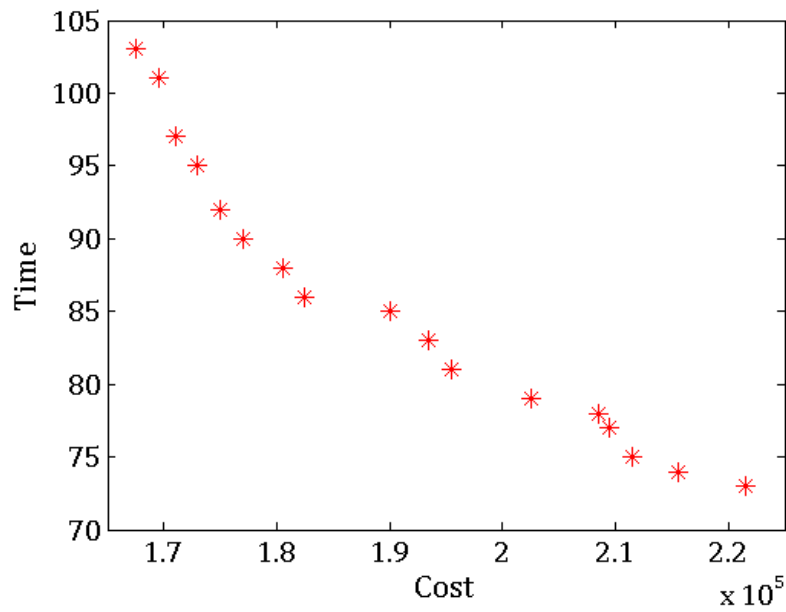


Fig.7: Pareto solution set of the conditional activity network of the example 2.

Table 5: the values of time and cost in the Pareto set of the conditional activity network of the example 2.

No.	Time	Cost
1	73	221500
2	74	215500
3	75	211500
4	77	209500
5	78	208500
6	79	202500
7	81	195500
8	83	193500
9	85	190000
10	86	182500
11	88	180500
12	90	177000
13	92	175000
14	95	173000
15	97	171000
16	101	169500
17	103	167500

Example 3:

in the last example for the conditional activity network (fig.8), the method of implementation of an activity simultaneously influences several activities, in a way that the implementation of D activity with its first implementation option (D1) leads to the elimination of F, G and I activities.

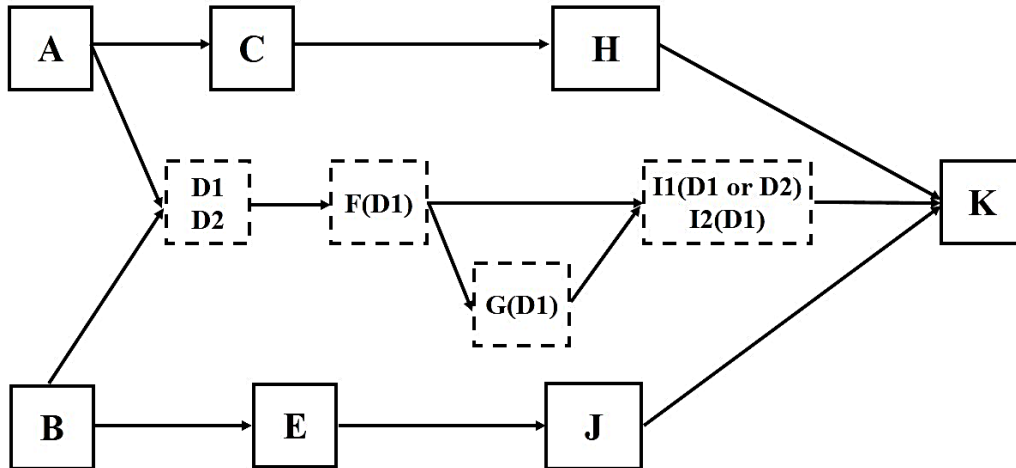


Fig.8: the example 3, conditional subnetwork of the conditional activity network with four conditional activities.

As it is observable from fig.8, in the conditional activity network of the example 3, A, B, C, D, E, F, H, J, K activities are implementable with all of their implementation options, but I, G, F activities have conditional limitations as follows:

- If D activity is implemented with its first implementation option (D1) I, G, F activities with all their implementation options will be eliminated.
- If D activity is implemented with its second implementation option (D2) I activity is not implementable with its first implementation option (I1).

The implementation options of activities related to this example is presented in table 6.

Table 6: implementation options of the conditional activity network of the third example.

No	Pr	Op	Ti	Cost	No	Pr	Op	Ti	Cost
A	-	1	5	1200	F	D	1	2	1300
		2	6	1050			2	3	1100
		3	8	900			3	4	850
B	-	1	4	5000	G	F	1	3	1200
		2	5	4000			2	4	900
		3	6	3000	H	C	1	3	4700
C	A	1	6	2000			2	5	4100
		2	7	1700			3	8	3400
		3	9	1300	I	F	1	2	1700
		4	10	1100			2	3	1300
D	A	1	10	5400	J	E	1	2	2800
	B	2	5	1300			2	3	2550
E	B	1	2	5500			3	4	2250
		2	4	4200			4	5	2100
		3	5	3700	K	H	1	2	2300
		4	7	2800			2	5	1400

Pareto solution set for the conditional activity network of the third example resulted from SMOCSS algorithm is presented in fig.9 and table 7.

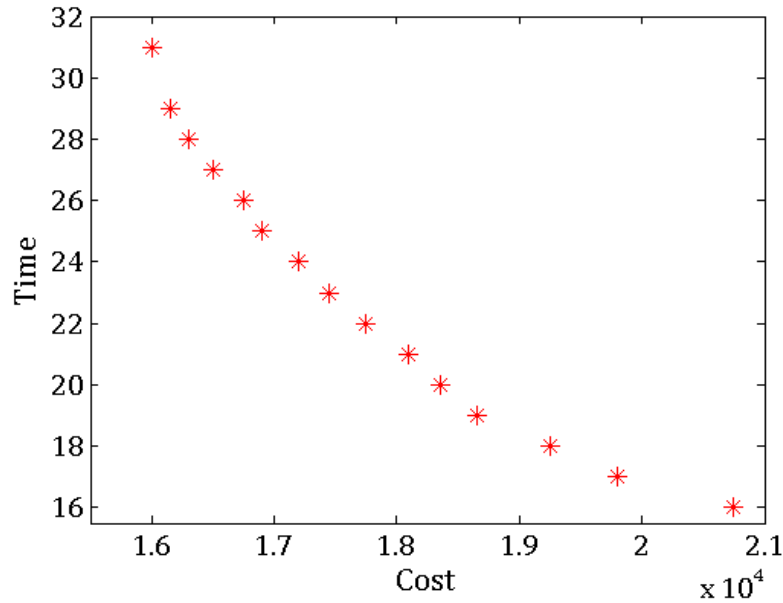


Fig.9: Pareto solution set of the conditional activity network of the example 3.

Table 7: the values of time and cost in the Pareto set of the conditional activity network of the example 3.

No.	Time	Cost
1	16	20750
2	17	19800
3	18	19250
4	19	18650
5	20	18350
6	21	18100
7	22	17750
8	23	17450
9	24	17200
10	25	16900
11	26	16750
12	27	16500
13	28	16300
14	29	16150
15	31	16000

Conclusion

In this paper, by studying planning methods and examining the actual projects at hand in construction, industry, etc. and observing some conditional activities in projects, the conditional activity network is introduced that is a new type of planning and scheduling method of project planning. Then, for illustrating such networks combined rules of illustrating knot and vector networks with some new rules, that are stemmed in the conditional activity networks, for preventing from any vagueness. Then, for time-cost formulation in TCTP

problems of conditional projects, a new parameter was defined for the elimination of unauthorized paths from all the project implementation paths and determining authorized paths.

Finally, some examples of conditional activity networks were designed and solved by using of meta-heuristic algorithms and the results were compared in terms of implementation time and the number of Pareto solutions. The results of time-cost optimization of conditional networks in the examples showed that all of the applied algorithms are capable of finding all of the dominant solutions, but SMOCSS algorithm is better than NSGA-II and MOPSO algorithms in terms of relative implementation time.

References

- Hindelang, T.J., Muth, J.F. (1979). A dynamic programming algorithm for decision CPM networks. *Operat Res*, pp. 225-241.
- Elmaghraby, S.E. (1993). Resource allocation via dynamic programming in activity networks, *Eur. J. Operational Res.*, 64. 199-215.
- Feng, C.W., Liu, L. & Burns, S.A. (1997). Using genetic algorithms to solve construction time-cost trade-off problems', *Journal of Computing in Civil Engineering*, ASCE, 11(3). 184-189.
- Zheng, D. X.M., Thomas N.g., S.T. & Kumaraswamy, M.M. (2004). Applying a genetic algorithm-based multi-objective approach for time-cost optimization', *Journal of Construction Engineering and Management*. 130(2). 168-176.
- Daisy X. M., Zheng, S., Thomas N.g., & Mohan M. Kumaraswamy. (2005). Applying Pareto Ranking and Niche Formation to Genetic Algorithm-Based Multiobjective Time-Cost Optimization. *J. Construction. Eng. Manage.* 131(1), 81-91.
- Coello Coello, C. A., Pulido, G. T., & Lechuga, M. S. (2004). Handling multiple objectives with particle swarm optimization. *IEEE Transaction on Evolutionary Computation*, 8(3), 256–279.
- Carlos A. Coello Coello C. A., David A., Van Veldhuizen, Gary B. Lamont. (2002). *Evolutionary Algorithms for Solving Multi-Objective Problems*. Boston. Kluwer Academic Publishers.
- Kaveh A., Talatahari S. (2010). A novel heuristic optimization method: charged system search. *Acta Mech*; 213: 267–289.
- Kaveh, A., & Talatahari, S. (2010). Optimal design of skeletal structures via the charged system search algorithm. *Structural and Multidisciplinary Optimization*. 37. 893–911.
- Kaveh, A., & Talatahari, S. (2010). Charged system search for optimum grillage systems design using the LRFD-AISC code. *Journal of Constructional Steel Research*. (66), 767–771.

Sharbatdar M.K., Talatahari S., Mousavi. M.R. (2016). Developing New Charged System Search-based Algorithm: Application in The Time-Cost Trade-off Problems. TURKISH ONLINE JOURNAL OF DESIGN ART AND COMMUNICATION. 4.

Butterfield. D. (2011). Painting and Decorating, 6th Ed. Wiley-Blackwell. ISBN: 978-1-4443-3501-9.

McCabe. G.B., Kennedy. J.R. (2003). Planning the Modern Public Library Building. Westport, Conecticut. London. The libraries unlimited library management collection.

Levin. R. (1980). Planning and control with PERT/CPM. M.Y. McGraw-Hill.