Multi-Objective Optimization of Construction Project Time-Cost-Quality Trade-off Using Differential Evolution Algorithm

A. Sathya Narayanan [1] ***and C. R. Suribabu*** [2]

[1] M. Tech. (Construction Engineering and Management) Student
[2] Professor, School of Civil Engineering, SASTRA University, Thanjavur -613401, Tamilnadu, India.
E-Mail: suribabu@civil.sastra.edu

ABSTRACT

Time and cost are among the important aspects considered for every construction project. Many research approaches have been followed to model time-cost relationship. There is a constant rise in the use of innovative contract methods which provide incentives for maximizing quality. There is an increasing pressure to improve the project performance due to the innovative contracting methods which necessitate developing models incorporating quality along with time and cost. A main contractor normally subcontracts most of the tasks of a project for improving project performance. It is always a complex and challenging task for a main contractor, to choose a correct bid which satisfies the time, cost and quality requirements of a project. In the present study, a differential evolution algorithm is used to solve this multi-objective time-cost-quality optimization problem. Two case studies are analyzed and the results obtained compared with the existing approaches to test the applicability and efficiency of the algorithm. It is evident from the results that the differential evolution algorithm performs efficiently in locating the optimal solution with minimum function evaluation.

KEYWORDS: Time-cost-quality trade-off, Multi-objective optimization, Differential evolution optimization.

INTRODUCTION

The main objective of construction project planning and control is to execute the project within the anticipated time while satisfying the quality requirements apart from minimum cost. Use of innovative contracting is gaining importance among contractors as it brings the incentives in terms of quality of work being executed. In case of warranties contract, contractors are liable for the performance of the project. This forces the contractor to improve the quality of the project. This kind of innovative contracting method places a huge pressure on the contractor to maximize the quality along with time and cost. Trade-off between these conflicting aspects of the project is a challenging job and planners are faced with numerous possible combinations for project delivery. For example, the number of possible combinations in a project with 18 activities and 4 possible resource utilization options for each activity will be more than 6 billion (El-Rayes et al., 2005). Hence, an efficient searching tool is vital to evaluate best alternatives from options for decision makers.

Various optimization approaches have been used to solve the construction scheduling problem, and they can be classified as mathematical, heuristic and meta-heuristic methods (Zhou et al., 2013). Burns et al. (1996) used a hybrid of linear programming and

Accepted for Publication on 6/4/2014.

integer programming. Li and Love (1997), Hegazy (1999) and Leu et al. (2001) used genetic algorithm. Li and Love (1999) used a mixture of machine learning and genetic algorithm. All these models targeted minimizing the cost without due attention to reduce the time simultaneously. Several researchers made an attempt to balance the completion time and cost for improving the performance of construction projects. Notably, Feng et al. (1997), Leu and Yang (1999) and Zheng et al. (2005) used genetic algorithm for solving this time-cost trade-off problem. Feng et al. (2000) used simulation techniques and genetic algorithm. Zheng et al. (2004) used adaptive weight approach and genetic algorithm. Zheng and Thomas (2005) used fuzzy sets theory and non-replaceable front for the stochastic time-cost optimization problem. Afshar et al. (2009) used ant colony algorithm to solve this time-cost trade-off problem. Recent trends concentrated on the need for incorporating quality along with the traditional time-cost optimization since incentives are provided for maximizing quality.

Use of traditional optimization techniques appears more common to solve the time-cost-quality (TCQ) trade-off problems. Babu and Suresh (1996) used three inter-related linear programming models by assuming the relationships among the project completion time, project cost and quality. Khang and Myint (1999) applied the method proposed by Babu and Suresh (1996) to a real construction project and then investigated its practical applicability and efficiency. El-Rayes and Kandil (2005) used genetic algorithm for solving the time-cost-quality trade-off problem, and measurable quality indicators for each activity in the project were introduced in order to quantify the construction quality, whereas there was no clear measurement approach in the previous studies. Afshar et al. (2006) and Shrivastava et al. (2012) used ant colony optimization for optimizing time-cost-quality and time-cost-quality-quantity, respectively. Santosh et al. (2013) used a fuzzy cluster genetic algorithm approach to optimize time, cost and quality. Lakshminarayanan et al. (2010) converted the quality parameter into a risk factor based on the comparative study and opinion analysis from project managers, building construction contractors and construction consultants, and it was solved by ant colony optimization approach to minimize the time-cost-risk of the project. In this paper, a differential evolution approach is used to model the multi-objective time-cost-quality optimization problem.

Problem Description and Formulation

Any construction project generally consists of several activities that need to be completed within the specified duration. Due to precedence relationships among those activities, the project forms an activity network. Main contractor usually allots certain or all activities to subcontractors due to the limitation in their own capacity and resources. They float bids and receive bids with respect to both duration and cost point of view from different subcontractors. There is a chance of getting several subcontractors for each activity of the project. The capacity of each subcontractor is usually assessed based on the amount that they have quoted, completion of the task and quality of work that they may render. The problem mainly consists of selecting appropriate resource utilization options for each activity to obtain minimum cost and time and maximum quality in the project in overall. The objective function defining time can be expressed as:

$$Min\,T = \max_{L_p \in L} \sum_{i \in L_p} \sum_{j=1}^{m_j} d_{ij} x_{ij} \tag{1}$$

where d_{ij} represents the duration of activity i when performing the jth option; and x_{ij} is the index variable of activity i when performing the jth option. If x_{ij} =1 , then activity i performs the jth option, and if x_{ij} = 0, then activity i does not. The sum of index variables must be equal to one. L is the set of all network paths {1,2,…..p}. L_p is the activity sequence on the pth path, and m_i is the number of subcontracting option for activity i, for i = 1,…,N.

The cost of a project consists of both direct and indirect cost. Direct cost is the sum of direct cost of all activities within the project network. Indirect cost is the expenditure on management during project implementation, which depends on the project duration. The indirect cost will be higher in case of longer projects. The objective function depicting the total cost of the project can be expressed as:

$$\text{Min } C = \sum_{i=1}^{N} \sum_{j=1}^{m_j} c_{ij} x_{ij} + IC \times T \qquad (2)$$

where, c_{ij} is the cost of subcontracting option for activity i for the jth option; IC is the indirect cost of the activities per day.

The construction quality quantification as a function of different resource utilizations is a challenging work because of difficulty in measuring the impact of performing activities on the quality of activities. Some indicators have been investigated and identified in recent studies for developing contractor prequalification systems based on quality (Anderson and Russell, 2001). The identified quality indicators were obtained from performance-based models which correlate the future performance of each activity to its quality indicators. The objective function expressing the quality of the project can be expressed as:

$$\text{Max } Q = \sum_{i=1}^{N} wt_i \sum_{l=1}^{L} wt_{i,l} \; q_{i,j,l} \; x_{ij} \qquad (3)$$

where, $q_{i,j,l}$ is the performance quality indicator l in activity i using the jth resource utilization option. $wt_{i,l}$ is the weight of quality indicator (l) compared with other indicators in activity i. wt_i is the weight of activity i compared to other activities of the project.

Multi-objective optimization is established using the following approaches.

Method 1

$$\text{Min } Z = \left[W_t * \left[\frac{T - T_{min} + \gamma}{T_{max} - T_{min} + \gamma} \right] + W_c * \left[\frac{C - C_{min} + \gamma}{C_{max} - C_{min} + \gamma} \right] + W_q * \left[\frac{Q_{max} - Q + \gamma}{Q_{max} - Q_{min} + \gamma} \right] \right] \qquad (4)$$

where,

W_t, W_c and W_q are the adaptive weights for time, cost and quality given by:

$$W_t = \frac{V_t}{V} \; ; \; W_c = \frac{V_c}{V} \; ; \; W_q = \frac{V_q}{V}.$$

V_t, V_c and V_q are the criteria for time, cost and quality, respectively:

$$V_t = \frac{T_{min}}{T_{max} - T_{min}} ; V_c = \frac{C_{min}}{C_{max} - C_{min}} \; ;$$

$$V_q = \frac{Q_{max}}{Q_{max} - Q_{min}}.$$

V is the cumulative criterion given by:

$$V = V_t + V_c + V_q;$$

T, C and Q are the objective time, cost and quality in the respective sequence of solution. T_{min}, C_{min}, Q_{min}, T_{max}, C_{max} and Q_{max} are the minimum and maximum time cost quality obtained when the problem is optimized as single objectives.

Method 2

$$\text{Min } U = \left[W_t * \left[\frac{T - T_{min}}{T_{min}} \right]^2 + W_c * \left[\frac{C - C_{min}}{C_{min}} \right]^2 + W_q * \left[\frac{Q_{max} - Q}{Q_{max}} \right]^2 \right]^{1/2} \qquad (5)$$

where,

$$W_t = \frac{1}{T_{max} - T_{min}} ; W_c = \frac{1}{C_{max} - C_{min}} \; ;$$

$$W_q = \frac{1}{Q_{max} - Q_{min}}.$$

Differential Evolution Algorithm

Differential Evolution (DE) algorithm is a search-based stochastic optimization algorithm introduced by Storn and Price (1995) for solving complex continuous optimization problems as an improvement over genetic algorithm. The DE algorithm uses population-based solution exploration with the help of crossover, mutation and selection operators. DE explores the best candidate solutions iteratively until the stopping criterion is reached. It requires an initial population containing individuals or vectors (candidate solutions) that can be generated randomly. The fitness value of each candidate solution obtained from the initial population is calculated according to the chosen objective function. Two candidate solutions are selected randomly from the population, and the vector difference between them is calculated, and its weighted value is calculated by a multiplying factor called mutation (0 to 1), and the resulting weighted vector is added with the third randomly selected candidate solution which needs to be selected from the population other than the earlier selected two candidate solutions. The new candidate solution so obtained from the above process is called noisy vector. This noisy vector is now subjected to crossover process with a target vector selected randomly from the population. The candidate solution obtained at the end of crossover process is called the trial vector. The vector having best fitness among trial and target vectors is considered as a candidate solution to the next generation. The number of candidate solutions for the next generation for the chosen population size is obtained by repeating the above-mentioned procedure a number of times equal to the population size. The entire process is repeated either a predefined number of generations or until specified termination criterion is achieved.

The stepwise procedure is illustrated as follows:

1. Initial candidate solutions are generated randomly for the chosen population size (pop_size) to form the initial population and account this as first generation (G = 1). The expression for creating random solution is as follows:

$$d_{i,j}^{0} = d_j^{(L)} + r_{i,j}^{G}(d_j^{(U)} - d_j^{(L)}),\ \forall i = 1\ to\ s,\ \forall j = 1\ to\ n \quad (6)$$

where $r_{i,j}^{G}$ denotes a uniformly distributed random value within the range from 0.0 to 1.0. $d_j^{(U)}$ and $d_j^{(L)}$ are upper and lower limits of variable d_j.

2. In the next step, weighted vector is calculated by multiplying mutation factor F with differential vector obtained by finding the difference between two randomly selected vectors from population.

$$w_j^G = F * \left(d_{A,j}^G - d_{B,j}^G\right) \qquad \forall j = 1\ to\ n \quad (7)$$

The weighing factor is usually selected between 0.4 and 1.0.

3. The population of trial vectors $P^{(G+1)}$ is generated as follows:

$$d_{i,j}^{G+1} = \begin{cases} d_{C,j}^{(G)} + w_j^G & if\ r_{i,j} \le C_r; \quad \forall j = 1\ to\ n \\ d_{i,j}^{(G)} & otherwise \end{cases} \quad (8)$$

where

$i = 1,........,pop\_size$

$A \in [1,........,pop\_size],\quad B \in [1,..........,pop\_size],$

$C \in [1,..........,pop\_size],\ A \ne B \ne C \ne i,$

$C_r \in [0\ to\ 1],\ F \in [0\ to\ 1], r \in [0\ to 1].$

C_r is crossover constant, which assists for differential perturbation in order to select the pipe diameter either from noisy vector or from target vector to form a new population for the next generation.

4. The population of the next generation $P^{(G+1)}$ is created as follows (Selection):

$$x_{i,j}^{G+1} = \begin{cases} d_{i,j}^{(G+1)} & if\ f(d_i^{(G+1)}) \le f(d_i^{(G)}) \\ d_{i,j}^{(G)} & otherwise \end{cases} \quad (9)$$

where $f(d_i^{(G)})$ represents the cost of the i^{th} individual in the G^{th} generation.

The superior performance of differential evolution over other competing algorithms has been reported by Suribabu (2010) for the design of water distribution network problem.

Application Example 1

A project example is analyzed to illustrate the use of the present optimization model and explain its capabilities. The example consists of seven activities, where each has a number of possible resource utilization options as shown in Table 1. The example is originally obtained from Feng et al. (1997). The same example was later investigated by Zheng et al. (2005), Afshar et al. (2006) and Lakshminaryanan et al. (2010) using different optimization approaches. The data presented in Table 1 is obtained from Afshar et al. (2006). For the sake of comparison, the indirect cost is assumed to be zero.

Table 1. Detailed data of the example

Activity	Preceding activity	Resource options	Duration (days)	Cost ($)	Weight (%)	Qualtiy (%)
1		1	14	23000	8	98
		2	20	18000		89
		3	24	12000		84
2	1	1	15	3000	6	99
		2	18	2400		95
		3	20	1800		85
		4	30	1200		70
		5	60	600		59
3	1	1	15	4500	14	98
		2	22	4000		81
		3	33	3200		63
4	1	1	12	45000	19	94
		2	16	35000		76
		3	20	30000		64
5	2,3	1	22	20000	17	99
		2	24	17500		89
		3	28	15000		72
		4	30	10000		61
6	4	1	14	40000	19	100
		2	18	32000		79
		3	24	18000		68
7	5,6	1	9	30000	17	93
		2	15	24000		71
		3	18	22000		67

Application Example 2

Another example introduced by Feng (1997) to illustrate construction time-cost trade-off has been considered in the present study to evaluate the efficiency of the proposed multi-objective models with differential evolution algorithm. El-Rayes and Kandil

(2005) gave the quality indicators for this example and same is present in Table 2.

Table 2. Complete data with quality indicator for Example 2

Activity	Preceding activity	Resource options	Time (days)	Cost ($)	Activity Weight (%) (wt_i)	Indicator Weight ($wt_{i,l}$)	Qualtiy Perform-ance ($q_{i,j,l}$)	Indicator Weight ($wt_{i,l}$)	Qualtiy Perform-ance ($q_{i,j,l}$)	Indicator Weight ($wt_{i,l}$)	Qualtiy Perform-ance ($q_{i,j,l}$)
1	0	1	14	2400	3	50	100	30	96	20	98
		2	15	2150			90		89		89
		3	16	1900			86		77		84
		4	21	1500			75		72		73
		5	24	1200			63		60		65
2	0	1	15	3000	5	40	98	40	94	20	99
		2	18	2400			87		94		95
		3	20	1800			81		92		85
		4	23	1500			77		72		70
		5	25	1000			60		66		59
3	0	1	15	4000	8	70	100	15	97	15	98
		2	22	4000			80		82		81
		3	33	3200			62		60		63
4	0	1	12	45000	11	50	99	35	91	15	94
		2	16	35000			74		71		76
		3	20	30000			59		63		64
5	1	1	22	20000	10	60	100	20	97	20	99
		2	24	17500			93		89		89
		3	28	15000			77		71		72
		4	30	10000			61		64		61
6	1	1	14	40000	11	50	95	25	95	25	100
		2	18	32000			76		74		79
		3	24	18000			59		62		68
7	5	1	9	30000	10	30	97	30	99	40	93
		2	15	24000			70		73		71
		3	18	22000			61		62		67
8	6	1	14	220	1	100	95	0	NA	0	NA
		2	15	215			83				
		3	16	200			75				
		4	21	208			68				
		5	24	120			61				

9	6	1	15	300	1	50	100	50	99	0	NA
		2	18	240			97		92		
		3	20	180			81		88		
		4	23	150			71		75		
		5	25	100			63		64		
10	6	1	15	450	1	60	94	40	97	0	NA
		2	22	400			79		83		
		3	33	320			63		69		
11	7	1	12	450	2	70	96	30	95	0	NA
		2	16	350			72		75		
		3	20	300			61		66		
12	5,9,10	1	22	2000	3	50	99	35	98	15	95
		2	24	1750			89		85		87
		3	28	1500			70		71		79
		4	30	1000			62		61		63
13	3	1	14	4000	7	40	99	40	96	20	97
		2	18	3200			73		71		76
		3	24	1800			60		62		63
14	4,10	1	9	3000	6	80	100	10	95	10	98
		2	15	2400			79		82		81
		3	18	2200			63		67		66
15	12	1	16	3500	7	70	100	30	98	0	NA
16	13,14	1	20	3000	3	30	97	30	96	40	98
		2	22	2000			89		85		87
		3	24	1750			81		79		78
		4	28	1500			72		73		74
		5	30	1000			67		60		62
17	11,14,15	1	14	4000	6	70	98	20	97	10	99
		2	18	3200			73		75		72
		3	24	1800			62		65		61
18	17,16	1	9	3000	5	30	98	45	99	25	94
		2	15	2400			75		77		71
		3	18	2200			63		66		67

Model Implementation

To provide optimal trade-off among cost, time and quality for decision makers, the formulated model with three objectives is solved using differential evolution algorithm. In the present study, computer code for differential evolution has been implemented in Eclipse Java platform. The termination criterion for the optimization process is arbitrarily set to 100, and the population size (p), crossover probability (Cr) and mutation factor (F) are assigned as 30, 0.5 and 0.8, respectively. The best set of solutions with its subcontracting plan is given in Table 3. The best solutions are generated by considering a single objective function alone. Most promising solution in terms of cost, time and quality from the generated solution is collected and listed in Table 3. T_{min}, $C_{min,}$, $Q_{min,}$ T_{max}, C_{max} and Q_{max} values are selected from Table 3 to evaluate weighted multi-objective functions (i.e., Eq. 4 & Eq. 5).

Table 3. Best set of solutions with subcontracting plan for example 1

Time (days)	Cost (%)	Quality (%)	Resource Utilization Option						
60	143500	90	1	1	1	1	1	3	1
60	165500	97	1	1	1	1	1	1	1
63	131000	85	1	1	1	2	2	3	1
63	133500	87	1	1	1	2	1	3	1
65	141300	86	1	3	1	3	1	2	1
65	142300	90	1	3	1	1	1	3	1
66	128500	90	1	1	1	2	3	3	1
69	136900	86	1	2	1	1	1	3	2
75	118000	76	1	1	2	3	4	3	1
74	112500	75	1	1	1	3	4	3	2
78	142200	86	1	1	3	1	1	3	1
81	106900	77	3	2	1	3	4	3	1
84	101500	73	3	1	1	3	4	3	2
85	108500	76	3	1	2	3	2	3	2
87	99500	73	3	1	1	3	4	3	3
91	101000	71	3	1	2	3	4	3	2
94	97800	70	3	3	2	3	4	3	3
105	97000	67	3	3	3	3	4	3	3
132	95800	65	3	5	3	3	4	3	3

RESULTS AND DISCUSSION

Table 3 provides the list of best solutions obtained by solving the model with single objective function. It can be seen from Table 3 that when least cost solution is selected, its time and quality need to be compromised. If quality alone is considered, then cost needs to be compromised, but minimum time is feasible. The multi-objectiveness of the present problem is handled in two different weighted objective functions. For any population-based algorithm, the selection process is considered as one of the crucial parts. In case of single objective optimization problem, the solution obtained in the present generation is carried to the next generation if the obtained solution is better than the initial population. In the DE algorithm, this selection process is carried out after obtaining trial vectors. Vector having least cost in case of minimization problem is selected by comparing trial vector cost and target vector cost chosen from the initial population. The solution having least cost between these two vectors enters to the next generation. This process in the present problem is handled by comparing with the weighted objective function values (i.e., Z as per Eq. 4 and U as per Eq. 5), which represent the sum of weighted values of cost, time and quality objective functions. That is; the solution having least Z and U values will be entering to the next generation. Table 4 shows the optimal solution obtained from the two models. First solution in Table 4 is obtained through the first approach (i.e., by Z as per Eq.4), and the second solution is obtained using the second approach (i.e., by U as per Eq. 5). Ten trial runs for each method are made by changing the random seed value. In the first approach, the minimum number of iterations at which the generation's average time, cost and quality reach the solution time, cost and quality is seventeen. In the second approach, it is reached at fourteenth generation. Hence, the second approach performance is more commendable than the first approach in terms of algorithm performance.

Table 4. Optimal solutions for seven activity problem

Method	Time (Days)	Cost ($)	Quality (%)	No. of trial runs by changing random seed	Population size	Total iterations	Minimum number of iterations at which the generation's average equals the solution
1	60	165500	97	10	30	30	17
2	60	165500	97	10	30	30	14

The selection of appropriate weights can also be done based on the importance predicted by the construction planner or decision maker. This can be implemented by considering either equal weight for each objective function or different weightage to each objective function according to the priority and importance considered by the construction manager. Apart from weightage assigned based on the above-mentioned procedure, investigation is also made by changing the values of weights in the first and the second approaches. With different combinations of weightage values in both methods, several trial runs are also made and obtained solutions are presented in Table 5. The proposed multi-objective optimization model is found to be more sensitive to weights when its value is altered. First method has given a distinct result for each weight combination. Out of eight combinations in the first method, only two solutions have the same result. But in case of the second method, five solutions are found to be the same for same combinations of the weightage. This indicates that the influence of formulated objective function to satisfy the multi-objectiveness is also an important factor in the multi-objective optimization model.

Table 5. Optimal solutions based on different proportions of weight (Example 1)

Method	W_t	W_c	W_q	Time (days)	Cost (%)	Quality (%)	Resource Utilization Option						
1	0.333	0.334	0.333	66	128500	90	1	1	1	2	3	3	1
1	0.2	0.2	0.6	66	150500	96	1	1	1	2	3	1	1
1	0.6	0.2	0.2	60	143500	90	1	1	1	1	1	3	1
1	0.2	0.6	0.2	86	104500	83	3	1	1	3	3	3	3
1	0.5	0.3	0.2	67	123500	88	1	1	1	3	3	3	1
1	0.5	0.2	0.3	66	138500	94	1	1	1	1	3	3	1
1	0.3	0.5	0.2	77	112500	87	3	1	1	3	3	3	1
1	0.3	0.2	0.5	66	150500	96	1	1	1	2	3	1	1
2	0.333	0.334	0.333	67	123500	88	1	1	1	3	3	3	1
2	0.2	0.2	0.6	67	123500	88	1	1	1	3	3	3	1
2	0.6	0.2	0.2	67	123500	88	1	1	1	3	3	3	1
2	0.2	0.6	0.2	78	107500	77	3	1	1	3	4	3	1
2	0.5	0.3	0.2	68	118500	78	1	1	1	3	4	3	1
2	0.5	0.2	0.3	67	123500	88	1	1	1	3	3	3	1
2	0.3	0.5	0.2	74	113500	78	2	1	1	3	4	3	1
2	0.3	0.2	0.5	67	123500	88	1	1	1	3	3	3	1

The performance of the present approach is compared with the solution of multi-objective optimization model (MOOM) by Lakshminarayanan et al. (2010) and multi-objective ant colony algorithm (MOACO) model proposed by Afshar et al. (2006). Table 6 shows the results of the present approach and other methods. Direct comparison shows that differential evolution approach provided the same time, higher cost and high quality with respect to the second solution for a smaller number of function evaluation and same time, cost and quality with respect to the first solution.

Table 6. Comparison of solutions between present approach and other approaches (Example 1)

Solution	Models	Time (days)	Cost ($)	Quality (%)	Resource Option						
1	MOOM	60	165500	97	1	1	1	1	1	1	1
2	*50,30- MOACO	60	155500	92	1	1	1	2	1	1	1
3	30,30 – DE APPRAOCH (Method 1)	60	165500	97	1	1	1	1	1	1	1
4	30,30- DE APPRAOCH (Method 2)	60	165500	97	1	1	1	1	1	1	1

50, 30 – 50 is the number of iterations and 30 is the population size, respectively.

The time-cost-quality optimization problem is an extension to the time-cost optimization problem. So, it is valid to compare the performance of differential evolution approach with other time-cost optimization problems. For converting the time-cost-quality trade-off problem into a time-cost trade-off problem, the weights of quality in multi-objective optimization equations of both methods are made zero. By solving the time-cost trade-off problem, method 1 gives a solution of time 68 days, cost $118500 and quality 78%. Its resource utilization option is [1,1,1,3,4,3,1]. Solving by method 2 gives the solution of time 60 days, cost $143500 and quality 90%. The resource utilization option obtained for this case is [1,1,1,1,1,3,1]. For comparison with the proposed approach, three more works have been taken from the literature in addition to already compared approaches for time, cost and quality, and the results are listed in Table 7. Geem (2010) used harmony search approach to optimize the same problem. Santhosh et al. (2013) investigated the same problem using fuzzy-clustering-based genetic algorithm approach (FCGA). Zheng and Thomas (2005) proposed modified adaptive weight approach (MAWA) to handle multi-objectiveness of the problem. Method 1 gives a solution with higher time, but it gives a much lesser cost when compared to all other models. Method 2 gives better results when compared to MAWA, MOACO and MOOM in both cost and time. It gives the same time and cost as FCGA-APPROACH and HS-APPROACH with lesser functional evolutions. While comparing the two solutions obtained using method 2 for time and cost trade-off with time-cost-quality trade-off, time-cost trade-off problem gives time and cost as 60 days and $143500, respectively, with 90% quality. But, in case of optimal solution obtained by solving it as a time-cost-quality trade-off problem, the solution has 97% quality at the additional expense of $22,000. If same comparison is made with method 1, time-cost trade-off problem gives time and cost as 68 days and $118500, respectively, with 78% quality. For time-cost-quality trade-off problem, optimal solution saves 8 days and has 97% quality at the additional expense of $47,000**.**

Table 7. Comparison of the results obtained by MAWA, MOOM, MOACO, HS approach, FCGA approach with the proposed DE approach (Example 1)

Solution	Model	Time (days)	Cost ($)
1	*50,50-MAWA	61	173000
2	100,50-MAWA	61	173000
3	30-MOACO	61	173000
4	MOOM	60	165500
5	30-MOACO	60	155500
6	1000,30-HS-APPROACH	60	143500
7	200-FCGA-APPROACH	60	143500
8	30,30-DE-APPRAOCH (Method 1)	68	118500
9	30,30-DE-APPRAOCH (Method 2)	60	143500

*50, 50 – 50 is the number of iterations and 50 is the population size, respectively.

Figs. 1 to 3 show the average value of objective time, cost and quality, respectively, for both methods. Method 2 reaches the minimum time a little faster as compared to method 1. Average cost and quality increase more or less the same for both solutions. In both cases, method 2 has performed slightly better. But, its ability to adopt to change in adoptive weight is not as good as that of method 1.

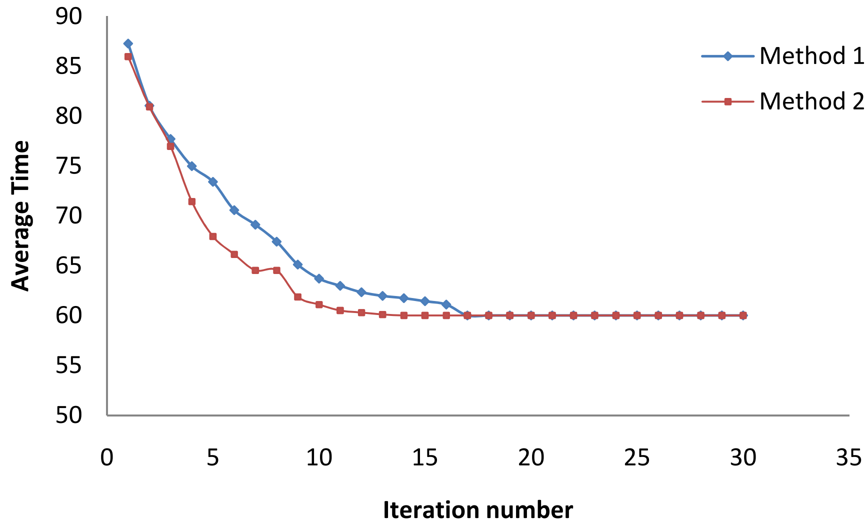

Figure (1): Average objective time value in the population in the present approach (Example 1)

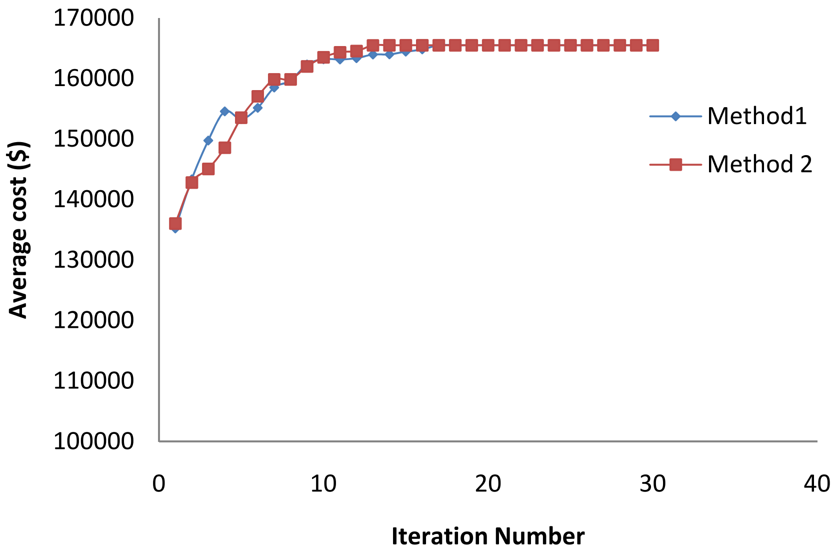

Figure (2): Average objective cost value in the population in the present approach (Example 1)

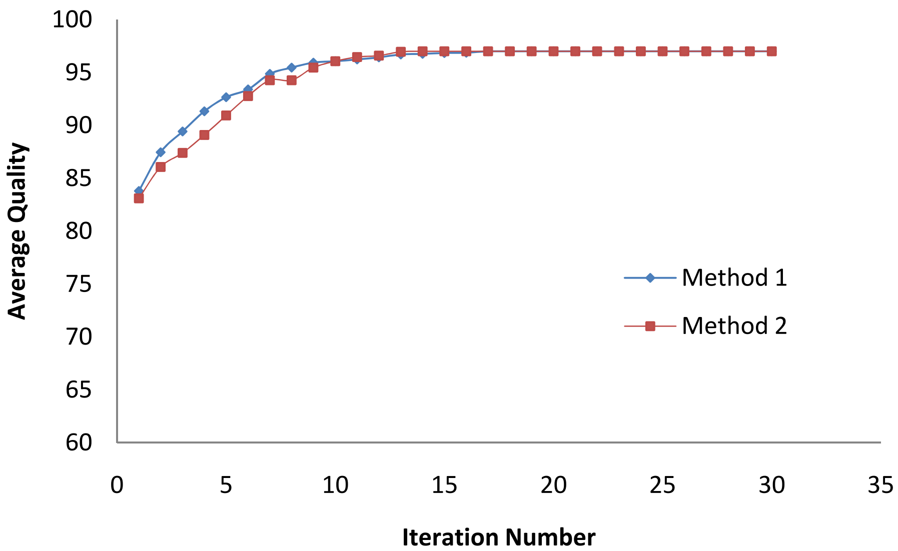

Figure (3): Average objective quality value in the population in the present approach (Example 1)

The results of example 2 are presented in Table 8. As this example is relatively of larger size in terms of search space, the number of iterations is set to 200 instead of 30. Ten trial runs are made by changing the random seed. The best solution found in these trial runs is tabled. It can be seen from Table 8 that the results obtained in both methods have the same time, but are of different cost and quality. The first method provided least cost and relatively lesser quality solution compared to method 2. Further, comparison is made with the solution obtained using Genetic Algorithm (GA) by El-Rayes and Kandil (2005). It is observed from Table 9 that the optimal solutions obtained using the proposed models are distinct from the solutions obtained from earlier studies. However, the time required to complete the project is found to be the same for all three solutions. From these three solutions, it is possible to note the influence of quality on the cost. For converting the time-cost-quality trade-off problem into a time-cost trade-off problem, the weights of quality in multi-objective optimization equations of both methods are made zero. By solving as time-cost trade-off problem, method 1 gives a solution of time 114 days, cost $105270 and quality 70%. The obtained resource utilization option is [1,5,3,3,4,3,3,5,1,1,3,1,3,3,1,5,1,1]. By method 2, it gives the solution of time 104 days , cost $ 132270 and quality 75%. The resource utilization option obtained in this case is [1,5,3,3,3,1,3,5,1,1,3,1,3,3,1,5,1,1]. In the time-cost trade-off optimization, when method 1 of the present work is compared with Harmony Search (HS) approach presented by Geem (2010), the solution of the present work gives a solution with 9 days extra time and saves $22050. Similarly, when method 2 solution is compared with HS approach, it saves a day at an expense of $4950. By changing the weights as shown in Table 11, optimal solutions are obtained using both methods. Like in the first example, method 2 has less ability to adopt to changing weights than method 1.

Table 8. Optimal solution for 18 activities example

Method	Time (Days)	Cost ($)	Quality (%)	No. of trial runs by changing ranom seed	Pop. size	Total iterations	Minimum number of iterations at which the generation's average equals the solution
1	104	152620	92	10	30	200	90
2	104	167770	96	10	30	200	55

Table 9. Comparison of solutions of present approach with El-Rayes and Kandil (2005)

Sol. No.	Models	Time (days)	Cost ($)	Quality (%)	Resource Option																	
1	GA	104	166320	95	1	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1
2	*200,30 – DE (Method 1)	104	152620	92	1	3	1	3	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	200,30- DE (Method 2)	104	167770	96	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	2	1	1

*200, 30 – 200 is the number of iterations and 30 is the population size, respectively.

Table 10. Comparison of the results generated by HS APPROACH (Geem, 2010)

Solution	Models	Time (days)	Cost ($)
1	HS	105	127320
2	*200,30 – DE APPROACH (Method 1)	114	105270
3	200,30 – DE APPROACH (Method 2)	104	132270

*200, 30 – 200 is the number of iterations and 30 is the population size, respectively.

Table 11. Optimal solutions under different weights (Example 2)

Method	W_t	W_c	W_q	Time (days)	Cost ($)	Qu (%)	Resource Option																	
1	0.333	0.334	0.333	114	112220	81	1	2	1	3	4	3	3	1	1	1	1	1	1	1	1	2	1	1
1	0.2	0.2	0.6	104	167770	96	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	2	1	1
1	0.6	0.2	0.2	104	142870	88	1	3	1	3	2	1	2	1	1	1	1	1	1	1	1	3	1	1
1	0.2	0.6	0.2	114	107565	76	1	5	1	3	4	3	3	2	1	2	1	1	3	1	1	5	1	1
1	0.5	0.3	0.2	105	133115	84	1	3	1	3	4	1	3	2	1	1	1	1	1	1	1	4	1	1
1	0.5	0.2	0.3	104	152620	92	1	3	1	3	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	0.3	0.5	0.2	114	110615	80	1	3	1	3	4	3	3	2	1	1	1	1	1	1	1	5	1	1
1	0.3	0.2	0.5	104	167770	96	1	1	1	1	1	1	1	1	1	2	1	1	1	1	1	2	1	1
2	0.333	0.334	0.333	114	110570	80	1	3	1	3	4	3	3	1	1	2	1	1	1	1	1	5	1	1
2	0.2	0.2	0.6	114	119570	84	1	3	1	3	4	3	1	1	1	2	1	1	1	1	1	2	1	1
2	0.6	0.2	0.2	114	110570	80	1	3	1	3	4	3	3	1	1	2	1	1	1	1	1	5	1	1
2	0.2	0.6	0.2	114	108315	77	1	3	1	3	4	3	3	2	1	1	2	1	3	1	1	5	1	1
2	0.5	0.3	0.2	114	110570	80	1	3	1	3	4	3	3	1	1	2	1	1	1	1	1	5	1	1
2	0.5	0.2	0.3	114	112220	81	1	2	1	3	4	3	3	1	1	1	1	1	1	1	1	2	1	1
2	0.3	0.5	0.2	114	110570	80	1	3	1	3	4	3	3	1	1	2	1	1	1	1	1	5	1	1
2	0.3	0.2	0.5	114	114770	82	1	1	1	3	4	3	2	1	1	2	1	1	1	1	1	2	1	1

Qu denotes Quality

Figs. 4 to 6 show the average values of objective time, cost and quality, respectively, for both methods. Methods 1 and 2 reach two distinct solutions. Method 1 converges to a solution with lesser cost and relatively lesser quality compared to method 2 . Method 2 converges a little faster than method 1. But, its ability to adopt to change in adoptive weight is not as good as that of method 1.

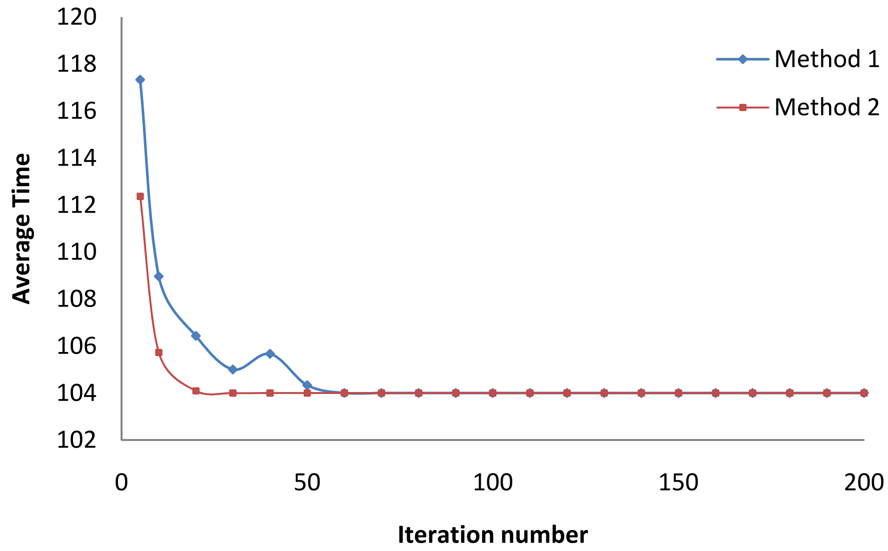

Figure (4): Average objective time value in the population in the present approach (Example 2)

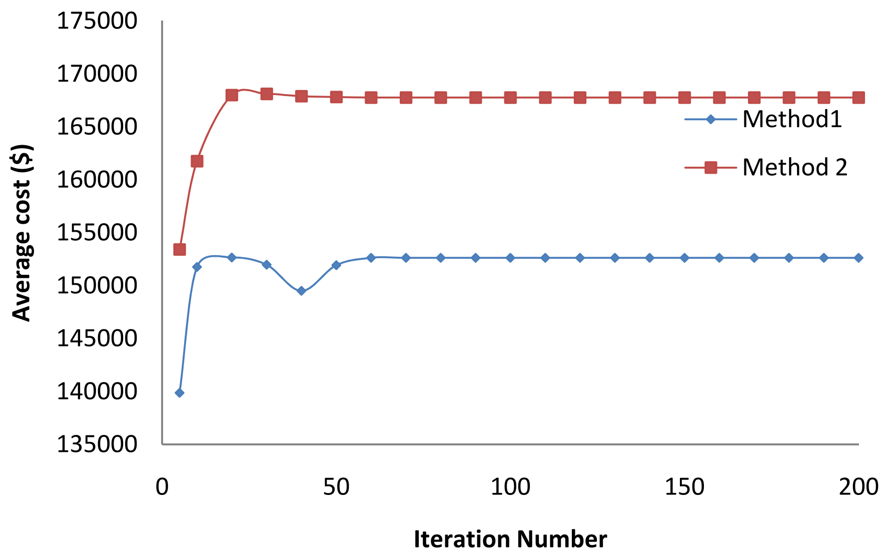

Figure (5): Average objective cost value in the population in the present approach (Example 2)

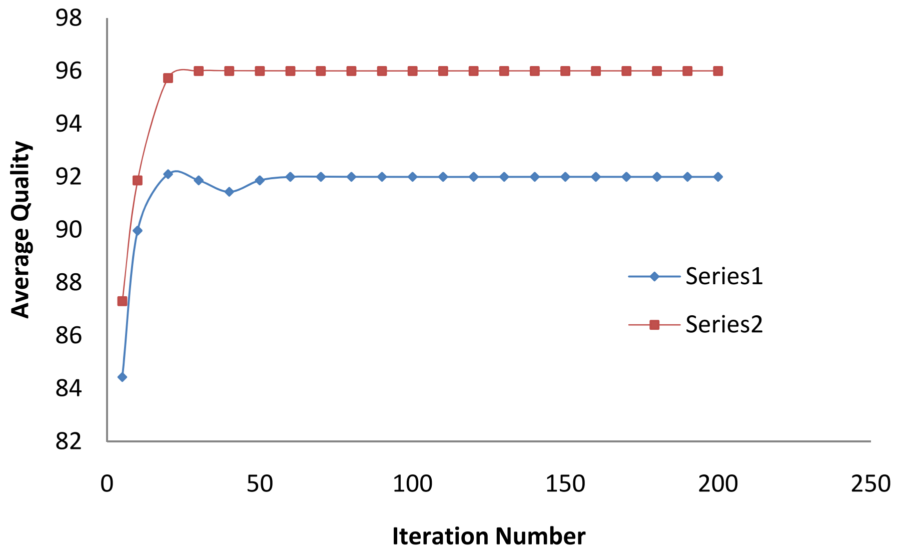

Figure (6): Average objective quality value in the population in the present approach (Example 2)

CONCLUSIONS

Differential evolution approach is applied to optimize the multi-objective time, cost and quality optimization problem. The model is designed to select optimal subcontracting plans that minimize time and cost of the projects while maximizing quality. The capability of the present approach in generating best general optimum solution is tested by comparing it with other existing approaches. It is noted that the differential evolution approach is capable of generating efficient results comparatively. The present approach provides an interesting substitute for the solution of construction multi-objective optimization problems.

REFERENCES

Afshar, A., Ziaraty, A.K., Kaveh, A., and Sgarifi, F. (2009). "Nondominated archiving multicolony ant algorithm in time–cost trade-off optimization." Journal of Construction Engineering and Management, 135 (7), 668-674.

Afshar, A., Kaveh, A., and Shoghli, O.R. (2006). "Time-cost trade-off analysis using multi-colony ant algorithm." Research Rep., Hydroinformatics Research Center, Iran Univ. of Science and Technology, Tehran, Iran.

Anderson, A., and Russell, J. (2001). "Guidelines for warranty, multi-parameter and best value contracting." NCHRP Report No. 451, National Cooperative Highway Research Program, Washington, DC.

Babu, A.J.G., and Suresh, N. (1996), "Project management with time, cost and quality considerations". European Journal of Operations Research, 88, 320-327.

Burns, S.A., Liu, L., and Feng, C.W. (1996). "The LP/IP hybrid method for construction time-cost trade-off analysis." Construction Management and Economics, 14 (3), 265-276.

El-Rayes, K., and Kandil, A. (2005). "Time-cost-quality trade-off analysis for highway construction." Journal of Construction Engineering and Management, 131 (4), 477-486.

Feng, C., Liu, L., and Burns, S. (2000). "Stochastic construction time-cost trade-off analysis." Journal of Computing in Civil Engineering, 14 (2), 117-126.

Feng, C.W., Liu, L., and Burns, S.A. (1997). "Using genetic algorithms to solve construction time-cost trade-off problems." Journal of Construction Engineering and Management, 123 (3), 184-189.

Geem, Z.W. (2010). "Multi-objective optimization of time–cost trade-off using harmony search." Journal of Construction Engineering and Management, 136 (6), 711-716.

Hegazy, T. (1999). "Optimization of construction time–cost trade-off analysis using genetic algorithms." Canadian Journal of Civil Engineering, 26 (6), 685-697.

Khang, D.B., and Myint, Y.M. (1999). "Time, cost and quality trade-off in project management: a case study." International Journal of Project Management.

Lakshminarayanan, S., Gaurav, A., and Arun, C. (2010). "Multiobjective optimization of time-cost-risk using ant colony optimization." International Journal of Project Planning and Finance, 1 (1), 22-38.

Leu, S.S., and Yang, C.H. (1999). "GA-based multicriteria optimal model for construction scheduling." Journal of Construction Engineering and Management, 125 (6), 420-427.

Li, H., and Love, P. E. (1997). "Using improved genetic algorithms to facilitate time–cost optimization." Journal of Construction Engineering and Management, 123 (3), 233-237.

Li, H., Cao, J. N., and Love, P. E. (1999). "Using machine learning and GA to solve time–cost trade-off problems." Journal of Construction Engineering and Management, 125 (5), 347–353.

Ng, S. T., and Zhang, Y. (2008). "Optimizing construction time and cost using ant colony optimization approach." Journal of Construction Engineering and Management, 134 (9), 721-728.

Santosh, M., Benyoucef, L., Young-Jun Son, and Tiwari, M.K. (2013). "A fuzzy clustering-based genetic algorithm approach for time–cost–quality trade-off problems: a case study of highway construction project." Engineering Applications of Artificial Intelligence, 26, 1953-1966.

Shrivastava, R., Singh, S., and Dubey, G.C. (2012). "Multi-objective optimization of time-cost- quality-quantity using multi-colony ant algorithm." International Journal of Contemporary Mathematical Sciences, 7 (16), 773-784.

Storn, R., and Price, K. (1995). "Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces." Technical Report, International Computer Science Institute, 8, 22-25.

Suribabu, C.R. (2010). "Differential evolution algorithm for optimal design of water distribution networks." Journal of Hydro-informatics, 12 (1), 66-82.

Zheng, X.M., Ng, S.T., and Kumaraswamy, M.M. (2004). "Applying a genetic algorithm-based multiobjective approach for time-cost optimization." Journal of Construction Engineering and Management, 130 (2), 168-176.

Zheng, X.M., Ng, S.T., and Kumaraswamy, M.M. (2005). "Applying Pareto ranking and niche formation to genetic algorithm-based multi-objective time–cost optimization." Journal of Construction Engineering and Management, 131 (1), 81–91.

Zheng, X.M., and Ng, S.T. (2005). "Stochastic time-cost optimization model incorporating fuzzy sets theory and non-replaceable front." Journal of Construction Engineering and Management, 131 (2), 176-186.

Zhou, J., PED Love, Wang, X., Teo, K.L., and Irani, Z. (2013). "A review of methods and algorithms for optimizing construction scheduling." Journal of Operation Research Society, 64, 1091-1105.