

CHAOTIC INITIALIZED MULTIPLE OBJECTIVE DIFFERENTIAL EVOLUTION WITH ADAPTIVE MUTATION STRATEGY (CA-MODE) FOR CONSTRUCTION PROJECT TIME-COST-QUALITY TRADE-OFF

Min-Yuan CHENG, Duc-Hoc TRAN, Minh-Tu CAO

*Department of Civil and Construction Engineering, National Taiwan University of Science and Technology,
#43, Sec. 4, Keelung Rd., 106 Taipei, Taiwan*

Received 16 Dec 2012; accepted 18 Apr 2013

Abstract. Time, cost and quality are three factors playing an important role in the planning and controlling of construction. Trade-off optimization among them is significant for the improvement of the overall benefits of construction projects. In this paper, a novel optimization model, named as Chaotic Initialized Multiple Objective Differential Evolution with Adaptive Mutation Strategy (CA-MODE), is developed to deal with the time-cost-quality trade-off problems. The proposed algorithm utilizes the advantages of chaos sequences for generating an initial population and an external elitist archive to store non-dominated solutions found during the evolutionary process. In order to maintain the exploration and exploitation capabilities during various phases of optimization process, an adaptive mutation operation is introduced. A numerical case study of highway construction is used to illustrate the application of CA-MODE. It has been shown that non-dominated solutions generated by CA-MODE assist project managers in choosing appropriate plan which is otherwise hard and time-consuming to obtain. The comparisons with non-dominated sorting genetic algorithm (NSGA-II), multiple objective particle swarm optimization (MOPSO), multiple objective differential evolution (MODE) and previous results verify the efficiency and effectiveness of the proposed algorithm.

Keywords: evolutionary optimization, multiple objective analysis, differential evolution, time-cost-quality trade-off.

Introduction

Time and Cost are two essential factors in construction project planning. The ability to minimize time and cost of a project could determine the level of success of a construction company (Zahraie, Tavakolan 2009). Many methodologies have been proposed to solve the time-cost trade-off (TCT) problem by deciding an optimal combination of construction methods for all the activities (Zhang, Xing 2010). These existing techniques for the TCT problem include the heuristic methods (Siemens 1971; Elmaghraby 1993), the mathematical programming models (De *et al.* 1995; Burns *et al.* 1996) and the evolutionary algorithm (Feng *et al.* 1997; Hegazy 1999; Li *et al.* 1999; Yang 2007). In general, less expensive resources or technologies would result in longer project duration. For instance, using productive resources or technologies may save time, but this causes an increase in the cost. On the other hand, reduction of either time or cost may decrease quality of construction projects. Consequently, the relevant construction projects such as highways, tunnels, and bridges may age or deteriorate faster than expected, increasing the maintenance and rehabilitation cost. Hence, how to determine an optimal

combination of execution methods consisting of suitable construction technologies and resource utilization plans to minimize time, cost and maximize quality simultaneously, that is, time-cost-quality trade-off (TCQT) problem, is becoming a crucial issue for construction planners (Zhang, Xing 2010).

Recently, scant studies on TCQT problem have been undertaken. At first Babu and Suresh (1996) suggested that the quality of a completed project may be affected by project crashing. They developed three inter-related linear programming models to study the trade-offs among time, cost and quality. This method is applied by Khang and Myint (1999) to study an actual cement factory construction project in Thailand. Attempting to use evolutionary methods, El-Rayes and Kandil (2005) presented a multiple objective optimization model in search for an optimal resource utilization plan that minimizes construction cost and time while maximizing its quality jointly. The model was developed based on genetic algorithm (GA) to provide the capability of quantifying and considering quality in construction. Zhang and Xing (2010) proposed a Pareto-based multi-objective Particle Swarm Optimization (PSO) for the fuzzy TCQT problem.

Differential Evolution (DE) is one of the most popular evolutionary algorithms which can be used in a broad variety of highly nonlinear and complex optimization problems. DE is simply structured, easy to use but it has shown great robustness and a fast convergence for solving single objective global optimization problems (Storn, Price 1997; Price *et al.* 2005). In light of these advantages, several researchers attempt to extend DE to solve multi-objective problems (Das, Suganthan 2011; Zhou *et al.* 2011). Superior performance of DE over other multiple objective algorithms has been verified in many reported research works (Reddy, Kumar 2007; Wu *et al.* 2010; Xuexia *et al.* 2012). Despite many reported impressive performances of DE on benchmark functions and practice applications, no research has been done on DE to deal with TCQT problem. This inspires us to apply DE algorithm in solving time-cost-quality trade-off problem.

Therefore, this research develops a novel Multiple Objective Differential Evolution algorithm to facilitate the time-cost-quality trade-off analysis. On the basis of classical DE, this proposed algorithm utilizes the advantages of chaos sequences to generate an initial population and adopts an external elitist archive to store non-dominated solutions found during the evolutionary process. In order to maintain the exploration and exploitation capabilities of the proposed model, an adaptive mutation operation is introduced. It will be shown that the proposed algorithm can attain fast convergence without losing the diversity of solutions on the Pareto front. The paper is structured as follows. Section two is a brief review of literature related to the establishment of the new multi-optimization model while the third and fourth sections present in detail the overall pictures of the newly proposed multi-optimization model. The performance of the newly developed model is demonstrated using numerical experiment and result comparisons in section five. Conclusions are drawn in the last section of this article.

1. Literature review

1.1. Background of time cost quality problem

As mentioned previously, TCQT problem mainly concentrates on selecting an optimal combination of construction methods for all activities in order to arrive at an optimal compromise among time, cost and quality for the project. Three major objective functions are formulated in the following to enable the evaluation of project performance in construction time, cost and quality, respectively (El-Rayes, Kandil 2005).

The first objective considered is the minimization of total project duration. This objective can be calculated by the following expression:

$$\text{Minimize project time} = \sum_{i=1}^l T_i^n, \quad (1)$$

where: T_i^n is duration of activity i $\{i = 1, 2, \dots, l$ in the critical path using resource utilization (n). In this study,

project duration is estimated according to the procedure used by El-Rayes and Moselhi (2001).

Another objective considered is the minimization of total project cost. This objective can be calculated by the subsequent expression:

$$\text{Minimize project cost} = \sum_{i=1}^l \left[(M_i^n + D_i^n \times R_i^n) + (B_i^n) \right], \quad (2)$$

where: M_i^n is material cost of activity (i); D_i^n is duration of each activity (i) using resource utilization (n); R_i^n is daily cost rate in \$/day of resource utilization (n) in activity (i); B_i^n is subcontractor lump sum cost for resource utilization (n) in activity i , if any.

The third, which is also the last objective to be considered, is the maximization of project's overall quality. To compute the value of this objective, the following expression is used:

$$\text{Maximize project quality} = \sum_{i=1}^l wt_i \sum_{k=1}^K wt_{i,k} \times Q_{i,k}^n, \quad (3)$$

where: $Q_{i,k}^n$ is performance of quality indicator (k) in activity (i) using resource utilization (n); $wt_{i,k}$ is weight of quality indicator (k) compared to other indicators in activity (i); and wt_i is weight of activity (i) compared to other activities in the project.

1.2. Review of multiple objective optimization

This section briefly introduces some basic definitions related to this work. All of these are given with respect to minimization problem as every maximization problem can be converted into minimization problem.

1.2.1. Problem definition

The multi-objective optimization problem can be formally defined as:

$$\min_{X \in D} f(X) = [f_1(X), f_2(X), \dots, f_l(X)]; \quad (4)$$

$$\text{s.t } g_i(X) \geq 0; \quad i = 1, \dots, m; \quad (5)$$

$$h_j(X) = 0; \quad j = 1, \dots, p; \quad (6)$$

$$D = \{X \mid g(X) \geq 0, h(X) = 0\}, \quad (7)$$

where: $f(X)$ is the objective vector, k is the number of objective functions; $g_i(X)$ is set of inequality constraints; $h_j(X)$ is set of equality constraints. The m and p is the number of inequality and equality constraints, respectively. A solution $X(x_1, x_2, \dots, x_n)^T$ is a vector of n decision variables in feasible region D . The multi-objective optimization problem is to determine those vectors X , which yield the optimum values of all the objective

functions, from the set D of all vectors which satisfy (5) and (6).

1.2.2. Dominance and Pareto front

Pareto dominance is formally defined as follow (Deb et al. 2002):

Solution $X_1(x_{1,1}, x_{1,2}, \dots, x_{1,n})^T$ dominates $X_2(x_{2,1}, x_{2,2}, \dots, x_{2,n})^T$ if both the conditions are satisfied:

1. $\forall i \in (1, 2, \dots, k): f_i(X_1) \leq f_i(X_2)$. The solution X_1 is no worse than X_2 in all objective;
2. $\exists i \in (1, 2, \dots, k): f_i(X_1) < f_i(X_2)$. The solution X_1 is strictly better than X_2 in at least one objective.

So, while comparing two different solutions X_1 and X_2 , there are three possibilities of dominance relation between them:

- X_1 dominates X_2 ;
- X_1 is dominated by X_2 ;
- X_1 and X_2 are non-dominated to each other.

A non-dominated solution has no solution can be found that dominates it. The set of non-dominated solutions is called the Pareto front.

1.2.3. Fast non-dominated sorting

This approach is proposed by Deb et al. (2002), for each solution i of a set S , two entities are calculated:

1. Domination count n_i , the number of solutions which dominate the solution i ;
2. S_i a set of solutions which solution i dominates.

At the end of this procedure, all solutions which have their domination count as zero be set in the first non-dominated front (F_1). Now, for each of these solutions (each solution i with $n_i = 0$), it visits each member (j) of its set S_i and reduce its domination count (n_j) by one. In this process, if for any member j the domination count becomes zero then it is put in a separate list P . These members belong to the second non-dominated front (F_2). The above procedure can be continued with each member of P and the third front (F_3) is identified. This process continues until all solutions are classified.

1.2.4. Elitist archive and crowding measure

Since Zitzler and Thiele (1999) firstly proposed their strength Pareto evolutionary algorithm (SPEA) with elitist reservation mechanism, numerous researchers have adopted similar elitist reservation concept in the practice (Coello Coello 2006; Wang et al. 2010). In this study, the elitist reservation strategy is also adopted. An external archive is used to keep the best non-dominated solutions that are produced so far by evolutionary algorithm. Initially, this archive is empty. As the evolution progresses, good solutions enter the archive. As a consequence, the size of the true non-dominated becomes huge. The computational complexity of maintaining the archive

increases with the archive size. In addition, considering the use of archive member to direct further search, the archive size also affects the complexity of selection. Hence, the archive size will be kept at its specified maximum size.

To have a good diversity among generated non-dominated solutions in the fixed size external archive, we need to choose a good measure to evaluate the crowding degree around each non-dominated solution. Numerous estimation methods are proposed in the multiple objective evolutionary algorithms. Adaptive hypercube is applied in PAES and MOPSO, the number of divisions (MOPSO) or an appropriate depth parameter is needed selected to control the hypercube. Addition, when a solution converges near the Pareto front, the hypercube is comparatively large. SPEA (Zitzler, Thiele 1999) used the density estimation strategy. This method, however, has large computational complexity. Deb et al. (2002) proposed a crowding distance method to get an estimation of the density of solutions surrounding a particular solution in the population. In the crowding distance measure, only average distance is considered. It may not accurately reflect the crowding degree (Huang et al. 2008; Wang et al. 2010). This study adopted crowding entropy method (Wang et al. 2010) to estimate the density of solutions in external store.

The crowding entropy computation requires sorting the solutions in the archive according to each objective function value in ascending order of value. For each objective function, the boundary solutions (smallest and largest function values) are assigned an infinite crowding entropy value so that which are always selected. All other intermediate solutions are assigned a crowding entropy value according to Eqn (8). This calculation is continued with other objective functions. The overall crowding entropy value is calculated as the sum of individual crowding entropy values corresponding to each objective:

$$CE_i = \sum_{j=1}^k (c_{ij} E_{ij}) / (f_j^{\max} - f_j^{\min}) = -\sum_{j=1}^k [dl_{ij} \log_2(pl_{ij}) + du_{ij} \log_2(pu_{ij})] / (f_j^{\max} - f_j^{\min}); \quad (8)$$

$$E_{ij} = pl_{ij} \log_2(pl_{ij}) + du_{ij} \log_2(pu_{ij}); \quad (9)$$

$$pl_{ij} = dl_{ij} / c_{ij}; \quad (10)$$

$$pu_{ij} = du_{ij} / c_{ij}; \quad (11)$$

$$c_{ij} = dl_{ij} / du_{ij}, \quad (12)$$

where: dl_{ij} are the distances of the i^{th} solution to its lower and upper adjacent solution along j^{th} objective function, respectively. The parameter f_j^{\min} and f_j^{\max} are

the maximum and minimum values of the j^{th} objective function, and k is the number of objective functions.

1.3. Differential evolution

Differential evolution (DE) is a simple population-based, direct-search for solving global optimization problems (Storn, Price 1997; Price *et al.* 2005). The original DE algorithm is described briefly as follows.

Let $S \subset \mathcal{R}^n$ be the search space of the problem under consideration. Then, DE utilizes NP , D -dimensional parameter vectors: $X_{i,G} = \{x_{i,G}^1, x_{i,G}^2, \dots, x_{i,G}^D\}$, $i=1, 2, \dots, NP$ as a population for each generation of the algorithm. At each generation, DE applies two operators, namely mutation and crossover (recombination) to yield one trial vector $U_{i,G+1}$ which will compete with its target vector $X_{i,G}$ in the selection phase. For each target vector $X_{i,G}$, a mutant vector $V_{i,G+1}$ is determined by the following equation:

$$V_{i,G+1} = X_{r1,G} + F(X_{r2,G} - X_{r3,G}), \quad (13)$$

where: $r_1, r_2, r_3 \in \{1, 2, \dots, NP\}$ are randomly selected such that $r_1 \neq r_2 \neq r_3 \neq i$, and F is a scaling factor such that $F \in [0, 1]$.

Following the mutation phase, the crossover operator is applied to increase the diversity. For each mutant vector $V_{i,G+1}$, a trial vector $U_{i,G+1} = \{u_{i,G+1}^1, u_{i,G+1}^2, \dots, u_{i,G+1}^D\}$ is generated, using the following scheme:

$$u_{i,G+1}^j = \begin{cases} v_{i,G+1}^j & \text{if } \text{rand}_j[0,1] \leq CR \text{ or } j = j_{\text{rand}} \\ x_{i,G}^j & \text{otherwise} \end{cases}, \quad (14)$$

$j = 1, 2, \dots, D$

$CR \in [0, 1]$ is user-defined crossover constant; j_{rand} is a randomly chosen index from $\{1, 2, \dots, D\}$, which can ensure that trial vector $V_{i,G+1}$ will differ from its target $X_{i,G}$ by at least one parameter.

To decide whether the trial vector $U_{i,G+1}$ should be a member of the population in next generation, it is compared to the corresponding target vector $X_{i,G}$ using the greedy criterion. The selection operator is expressed as follows:

$$X_{i,G+1} = \begin{cases} U_{i,G+1} & \text{if } f(U_{i,G+1}) < f(X_{i,G}) \\ X_{i,G} & \text{otherwise} \end{cases}. \quad (15)$$

With the memberships of the next generation are selected, the evolutionary cycle of the DE iterates until a stopping condition is satisfied.

1.4. Chaos approach for population initialization

Chaos theory is a scientific theory describing erratic behaviour in certain nonlinear dynamical systems. Chaotic mappings can be considered traveling particles within a limited range occurred in a deterministic nonlinear dynamic system. There is no definite regularity for such a traveling path. Such a movement is very similar to a random process, but extremely sensitive to the initial

condition (Cheng, Huang 2010). Chaotic sequences have been proven easy and fast to generate and store, there is no need for storage for long sequences. Moreover, these sequences are deterministic and reproducible (Bedri Ozer 2010).

The one dimensional logistic map is one of the simplest systems with density of periodic orbits:

$$X_{n+1} = \mu X_n (1 - X_n). \quad (16)$$

In equation above, X_n is the n^{th} chaotic number where n denotes the iteration number. Obviously, X_n under conditions that initial $X_0 \in (0, 1)$ and that $X_0 \notin \{0.0, 0.25, 0.5, 0.75, 1.0\}$. The variation of control parameter μ in Eqn (16) will directly impact the behaviour of X greatly. The domain area of control parameter μ has often been defined as $[0, 4]$. In the experiments $\mu = 4$ has been used.

2. The proposed multiple objective differential evolution algorithm (MODE)

In this section, the proposed optimization algorithm, Chaotic Initialized Multiple Objective Differential Evolution with Adaptive Mutation Strategy (CA-MODE), is described in details. It is noticed that our algorithm is developed based on standard Differential Evolution (Storn, Price 1997; Price *et al.* 2005). In CA-MODE, potential candidate solutions have been taken care right from the initialization and the exploration and exploitation capabilities during various optimization processes are mainly concerned. Besides using chaos to generate potential initial points, we propose adaptive mutation strategy to prevent the search from becoming a purely random search or a purely greedy search. In this way the proposed algorithm can converge faster while maintaining a good diversity. The overall picture of the proposed algorithm is illustrated in Figure 1.

2.1. Chaotic initialization

Before the optimization process, the user provides the value of population size NP , Number of decision variables D , Number of objective functions M , sets mutant constant F , crossover probability constant CR , and initializes the external elitist archive $E = \Phi$ and maximum size of E , specifies the maximum number of generation G_{max} and lower bound (LB), upper bound (UB) of decision variables.

Population initialization is the first and the primary task in any evolutionary algorithm. The population in DE can be guided towards the more promising areas if the initial population can be spread as much as possible over the objective function surface. Hence, incorporating chaotic maps into DE is worth investigating. In CA-MODE, a combined initial population of $2NP$ solutions is generated using uniform random distribution and chaos techniques.

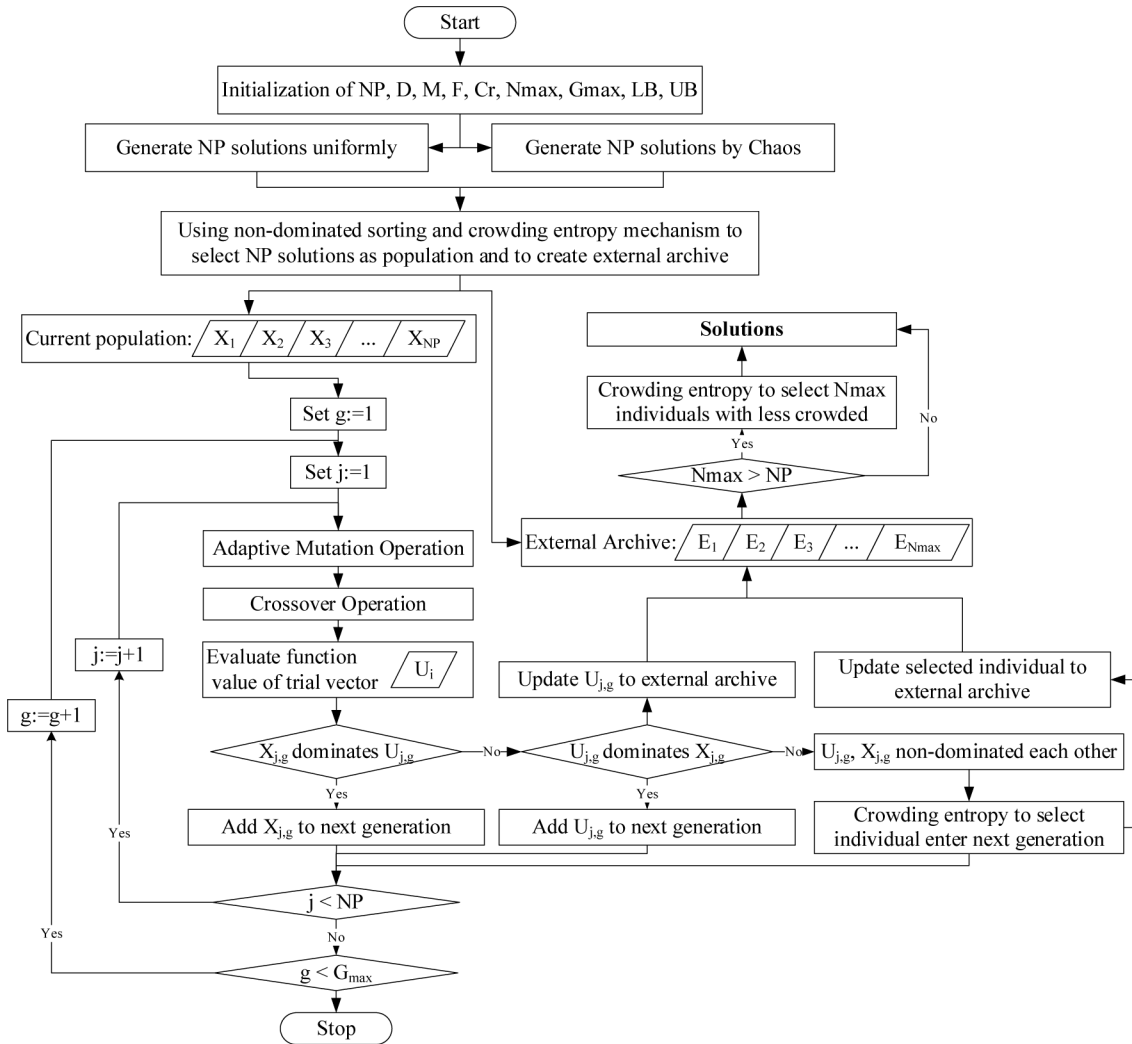


Fig. 1. Flowchart of CA-MODE

Generated a point in D -dimensional space $X = \{x_1, x_2, \dots, x_D\}$, where $x_1, x_2, x_D \in \mathfrak{R}$ and $x_j \in [0, 1]$ is uniform random distribution, and then the chaotic point $X^c = \{x_1^c, x_2^c, \dots, x_D^c\}$ is created as $x_j^c = \mu x_j (1 - x_j)$. So, we can simply generate the first NP individuals as follows:

$$X_{i,j}^{G=0} = LB_j + x_{i,j} * (UB_j - LB_j). \quad (17)$$

Chaos approach is used to generate other NP solutions according to the following equation:

$$X_{i,j}^c = LB_j + x_{i,j}^c * (UB_j - LB_j). \quad (18)$$

2.2. Select population and create elitist archive

During the optimization process size of population is kept as NP (Storn, Price 1997). From combined population, out of which NP best (elite) solutions are selected. It is important to note that in a single objective solution which has ‘highest fitness value’ is the best solution. Nevertheless, in multi-objective domain, to check two solutions dominance approach is used. Thus, in this

research, the NP solutions are selected based on fast non-dominated sorting (Deb et al. 2002) and entropy crowding technique (Wang et al. 2010). The overall picture of this procedure is illustrated in Figure 2.

The solutions belonging to the best non-dominated set (Set F_1) are selected first. If size of F_1 is smaller than NP, the remaining members of population are chosen from subsequent non-dominated fronts in order of their ranking ($F_2, F_3 \dots$). This procedure is continued until no

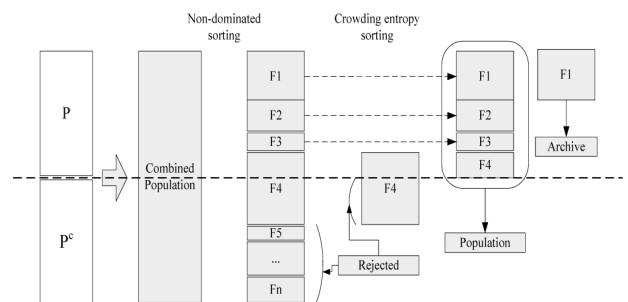


Fig. 2. Select population and create Archive procedure

more sets can be accommodated. Assume that F_k is the last non-dominated set beyond which no other set can be accommodated. In general, number of solutions in all set F_1 to F_k would be larger than NP. To choose exactly NP population members using crowding entropy sorting to select the best solutions, it is necessary to fill all population slots in descending distance order. Archive is the best non-dominated set (F_1).

2.3. Adaptive mutation operation

Once initialized, DE mutates the population to produce a set of mutant vectors. A mutated vector $V_{i,G+1}$ is generated corresponding to the target vector $X_{i,G}$ according to Eqn (13). In the original DE three mutant vectors are selected arbitrarily and the base vector is chosen randomly among 3 vectors (DE/rand/1/bin). This has a good exploratory but it slows down the convergence of DE. In DE/best/1/bin the base vector is always selected as one having the best fitness value. The strategy may provide a fast convergence but it may lead to loss of diversity.

To help CA-MODE in maintaining the exploration and exploitation capabilities, this research proposed three selection mechanisms that are operated based on itself parameter call $ms = 1/3$ (mutant selection) and maximum of generation G_{max} . Three phases of selection are described in Figure 3.

Phase 1: $g \leq ms * G_{max}$ (call random selection)

In Eqn (13), X_{r1}, X_{r2}, X_{r3} are randomly chosen in the population set of NP. At the beginning of the evolutionary process, all the vectors for mutation are randomly selected and the best point of the population may or may not be included in them. Because of its random nature, this strategy helps algorithm in preserving the diversity.

Phase 2: $ms * G_{max} < g \leq 2 * ms * G_{max}$

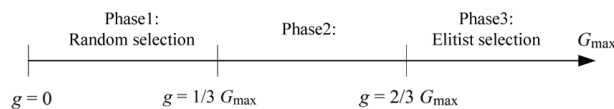


Fig. 3. Adaptive mutation operation

X_{r1} is the base vector is selected from external elitist archive. X_{r2}, X_{r3} are arbitrarily chosen in the population set of NP. In this strategy, the base vector is always selected as the one have the best fitness function value and the other vectors are chosen randomly among population. This technique makes algorithm neither purely greedy nor purely random in nature. It balances between diversity and convergence.

Phase 3: $g > 2 * ms * G_{max}$ (call elitist selection)

In the last phase of optimization process, the algorithm needs to accelerate the convergence. All the vectors X_{r1}, X_{r2}, X_{r3} for mutation are the best points which taken in the external elitist archive.

2.4. Crossover operation

The crossover operation is to diversity the current population by exchanging components of target vector and mutant vector. In this stage, a new vector, named as trial vector, is created according to Eqn (14).

The most important task of multi-objective optimization is the modification of the selection mechanism. The selection operation is based on the concept of Pareto dominance. In this operation, firstly evaluate trial vector $U_{i,G+1}$ then compare with target vector $X_{i,G}$. There may be at most three dominance possibilities between $U_{i,G+1}$ and $X_{i,G}$.

If $X_{i,G}$ dominates $U_{i,G+1}$, $U_{i,G+1}$ is discarded.

If $U_{i,G+1}$ dominates $X_{i,G}$, $X_{i,G+1} = U_{i,G+1}$, and update external archive with update rule.

If $U_{i,G+1}$ and $X_{i,G}$ are non-dominated each other, Crowding entropy is used to select individual to next target vector (the less crowded the better).

2.5. Update elitist archive

The vector which is chosen in selection operation process is called selected vector. If the selected vector is dominated by a member(s) of archive, the selected vector is rejected. If the selected vector dominates some member(s) of the archive, then the dominated members are deleted and selected vector is accepted. If the selected vector is non-dominated with each member in archive, it will enter archive.

The Update rule is illustrated in Figure 4. Figure 4a, in which 3 non-dominated solutions, shown as bold dots,

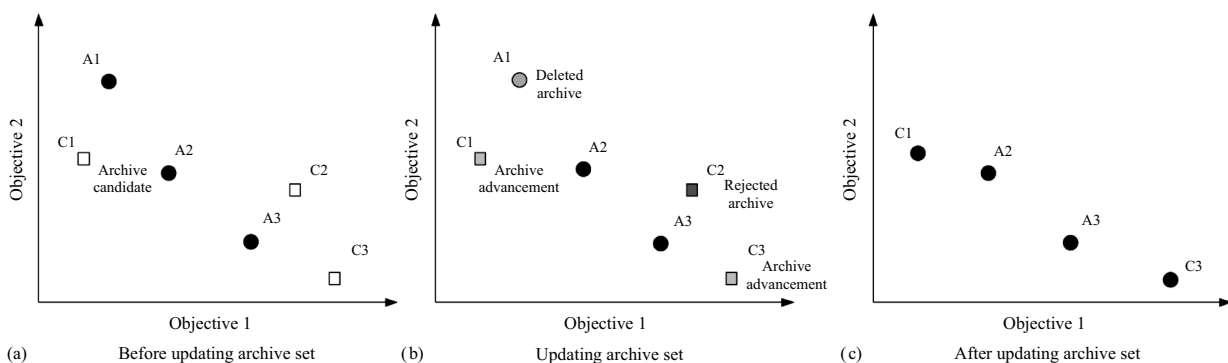


Fig. 4. Illustration of archiving scheme

are stored in the archive: A1, A2, and A3. At the current iteration, three archive candidate solutions are found to be C1 through C3, shown as hollow squares. All the update procedure is showed in Figure 4b, since archive member A1 is dominated by C1, A1 is deleted. C2 is dominated by some archive members, so C2 is rejected. C3 is non-dominated with each member in archive, it will become archive member. Figure 4c displays all archive members in current iteration after updating process.

When the external Archive population reaches its maximum capacity N_{max} , the crowding entropy to select N_{max} individuals with less crowded.

2.6. Stopping conditions

The optimization process terminates when the stopping conditions are met. The user can set type of these conditions. Commonly, maximum generation G_{max} or maximum number of functions evaluations (NFE) can be used as the stopping criterion. When the optimization process terminates, the final set of optimal solutions are readily presented to the user.

3. Multiple objective differential evolution for time-cost-quality trade-off (MODE-TCQ)

This section of the article is dedicated in describing the MODE-TCQ optimization model (Fig. 5). It is noticed that the MODE-TCQ is developed based on the CA-MODE as the searching engine. The objective of this optimization model is to minimize construction cost and duration while maximizing its quality.

In this study, we consider the case that the TCQT problem is accomplished by minimizing the cost, time while maximizing quality of construction projects. The

model requires inputs of project information including activity relationship, activity duration and construction methods for each activity. In addition, the user also needs to provide parameter setting for the search engine, such as maximum number of searching generation (G_{max}), the population size (NP), the size of external elitist archive (N_{max}). With these inputs, the optimizer can carry out calculation process to obtain optimal combination set of construction methods for all activities in a construction project. With all the necessary information provided, the model is capable of operating automatically without any human intervention.

Before the searching process can commence, an initial population of feasible solutions is created using a uniform random generator and chaos. A candidate solution to the TCQT problems can be represented as a vector with D elements as follows:

$$X = [X_{i,1}, X_{i,2}, \dots, X_{i,j}, \dots, X_{i,D}], \quad (19)$$

where D is the number of decision variables of the problem at hand. It is obvious that D is also the number of activities in the project network. The index i denotes the i^{th} individual in the population. The vector $X_{i,j}$ represents one construction method for activity j . The construction method $X_{i,j}$ is an integer number in the range of $[1, M_j]$ ($j = 1$ to D), meaning one position from M_j construction methods. Since original DE operates with real-value variables, a function is employed to convert those activities' construction method options from real values to integer values within the feasible domain:

$$X_{i,j} = Ceil(rand[0,1] \times UB(j)), \quad (20)$$

where $X_{i,j}$ is the option of activity j at the individual i^{th} . $rand [0,1]$ denotes a number between 0 and 1, generated by uniformly distributed random and chaos. $UB(j) = M_j$ is number of construction methods for each activity. $Ceil$ is a function to round a real number to the nearest integer greater than or equal to it.

The search engine (CA-MODE) takes into account the results obtained from scheduling module and searching for an optimal combination of construction methods for each activity. In the research, three contradicting objectives are employed. The formulae of each objective function are described in detail at Section 1.1.

After the searching process terminates, a set of optimal solution, which called Pareto front, is identified. Having the entire Pareto front is of great importance because it can assist planners evaluate the pros and cons of each solution based on qualitative and experience-driven considerations.

4. Case study

In order to demonstrate the effectiveness of the proposed Chaotic Initialized Multiple Objective Differential Evolution with Adaptive Mutation Strategy (CA-MODE) method for the TCQT problem, a numerical case is ana-

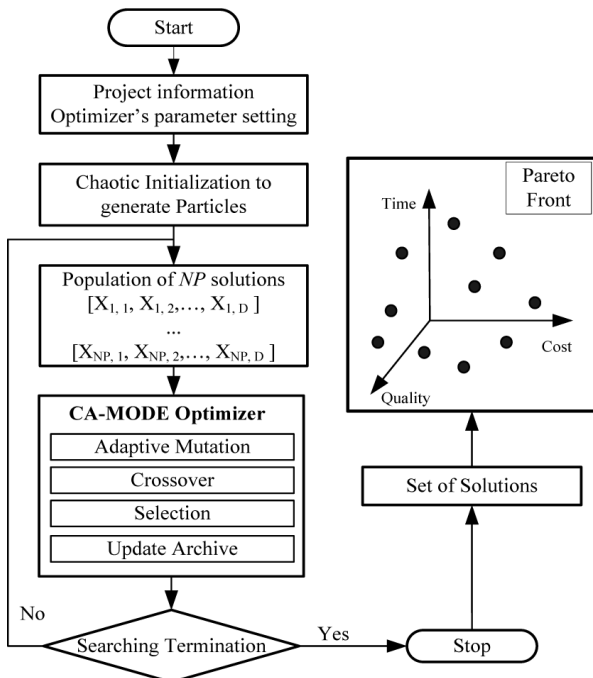


Fig. 5. Flowchart of CA-MODE for the TCQT problem

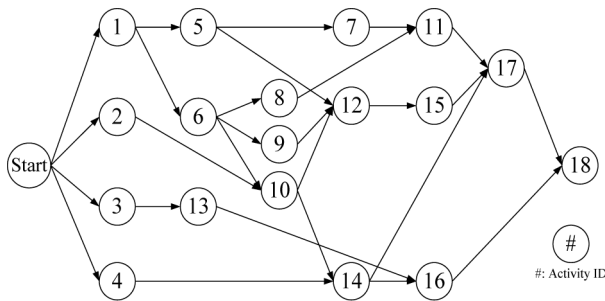


Fig. 6. Network of project

lysed. In addition, the results obtained by CA-MODE method are compared with NSGA-II, MOPSO, MODE and previous research findings. The case is adopted from previous study of an actual construction highway project (El-Rayes, Kandil 2005). This project consists of 18 construction activities, where each has a number of possible resource utilization options that can be used to construct the activity. Figure 6 shows the precedence relationships of the network and Table 1 shows the associated time, cost and quality for the options of activity. From this table, there is an average of 3.4 construction methods for each of 18 activities, which produces billions possible combination for delivering the entire project. Each of these possible combinations leads to a unique impact on project performance. It means decision makers have to face a main challenge of searching in large options to find solutions that establish an optimal and delicate trade-off among construction time, cost and quality. The present optimization model is used to search this large space of possible solutions using the newly developed multi-objective optimization model.

4.1. Optimization result of CAMODE-TCQT

In this section the parameter settings of the proposed CAMODE-TCQT are shown in Table 2 (Storn, Price 1997; Wu *et al.* 2010). To avoid randomness, ten independent optimization runs are conducted and average the external archive contains 100 members. To be concise, Table 3 lists the first 9 non-dominated solutions in the descending order of time, cost, and quality, respectively. These results also include the optimal combination of the construction methods. The distribution of the non-dominated solutions in Pareto optimal front using the proposed CA-MODE is show in Figure 7, which clearly shows the relationships among time, cost and quality. This three dimensional representation of visualize the trade-off among project time, cost and quality in order to support decision makers in evaluating the impact of various resource utilization plans on project performance.

The non-dominated solutions can also establish the trade-off between any two objectives on a two-dimensional plane. Figures 8–10 show the relationship between the time and the cost, the cost and the quality, the time and the quality, respectively. As we can see in the time-cost curve (Fig. 8) for example, the less cost

we spend on project, the longer duration we need to complete project and vice versa. However, Figures 8–10 may not be good representatives of the entire trade-off surface in the three-dimensional space. In fact, the two-dimensional trade-off surface, when projected from three to two dimensions, may lose some non-dominated points because the hidden dimension that makes these points non-dominated.

4.2. Statistical comparison and analysis

To demonstrate the effectiveness of CA-MODE, the performance of proposed model is compared with previous findings (El-Rayes, Kandil 2005), NSGA-II (Deb *et al.* 2002), MOPSO (Yang 2007) and MODE, which have been implemented and applied to TCQT problem with impressive success. In order to be convenient for comparison, all three algorithms used equal number of function evaluations, the population size and the maximum generations are set to 100 and 300, respectively. The size of approximated Pareto optimal set is chosen as 100. The constant mutant and crossover probability factor are chosen at 0.5 and 0.9, respectively. The control parameters of the proposed CA-MODE are the same as the previous. For all experiments, 10 times independent runs are carried out to collect the statistical results.

Table 4 shows the performance results obtained by proposed algorithm and benchmarked algorithms. Observing from this table, all of five solutions found by CA-MODE model are superior to previous findings and benchmarked models. In terms of time, all models can obtain the best time, but CA-MODE yields the higher quality and lower cost (case 1). The proposed algorithm can acquire lowest cost and highest quality. To further investigate the effectiveness of the proposed approach, several quantitative assessments of the performance of a multi-objective optimization algorithm are presented as follows.

Unlike in single-objective optimization, performance measures to multi-objective optimization problems are more complicated. Three issues are normally taken into consideration: (1) convergence to the Pareto optimal set; (2) maintenance of diversity in solutions of the Pareto optimal set; and (3) maximal distribution bound of the Pareto optimal set (Wu *et al.* 2010). Numerous quality indicators have been suggested in literature (Fonseca, Fleming 1995; Deb *et al.* 2002; Zitzler *et al.* 2003; Coello Coello 2006). The quality indicators can be classified into three categories depending on whether they evaluate the closeness to the Pareto front, the diversity in the solutions obtained, or both (Zitzler *et al.* 2003). In this research, three quality indicators evaluating each type of the above are described as the following:

1. C-metric (C): C-metric is quite often used to check the quality of the true Pareto front of optimized problem is not known (Zitzler, Thiele 1999). Let $S_1, S_2 \subseteq S$ be two sets of decision solutions.

Table 1. Options of resource utilization

Activity (i)	Resource option (n)	Duration (days)	Cost (dollars)	Act. weight (%) (wt_i)	QI k = 1		QI k = 2		QI k = 3		Quality
					IW ($wt_{i,k}$)	QP ($Q_{i,k}^n$)	IW ($wt_{i,k}$)	QP ($Q_{i,k}^n$)	IW ($wt_{i,k}$)	QP ($Q_{i,k}^n$)	
1	1	14	2400	3	50	100	30	96	20	98	2.9520
	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	1	15	3000	5	40	98	40	94	20	99	4.8300
	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	1	15	4500	8	70	100	15	97	15	95	7.9040
	2	22	4000			80		82		81	
	3	33	3200			62		60		63	
4	1	12	45000	11	50	99	35	95	15	94	10.6535
	2	16	35000			74		71		76	
	3	20	30000			59		63		64	
5	1	22	20000	10	60	100	20	97	20	99	9.9200
	2	24	17500			93		89		89	
	3	28	15000			77		71		72	
	4	30	10000			61		64		61	
6	1	14	40000	11	50	95	25	95	25	100	10.5875
	2	18	32000			76		74		79	
	3	24	18000			59		62		68	
7	1	9	30000	10	30	97	30	99	40	93	9.6000
	2	15	24000			70		73		71	
	3	18	22000			61		62		67	
8	1	14	220	1	100	95	0	NA	0	NA	0.9500
	2	15	215			83					0.8300
	3	16	200			75					0.7500
	4	21	208			68					0.6800
	5	24	120			61					0.6100
9	1	15	300	1	50	100	50	99	0	NA	0.9950
	2	18	240			97		92			0.9450
	3	20	180			81		88			0.8450
	4	23	150			71		75			0.7300
	5	25	100			63		64			0.6350
10	1	15	450	1	60	94	40	97	0	NA	0.9520
	2	22	400			79		83			0.8060
	3	33	320			63		69			0.6540
11	1	12	450	2	70	96	30	95	0	NA	1.9140
	2	16	350			72		75			1.4580
	3	20	300			61		66			1.2500

Continued of Table 1

Activity (i)	Resource option (n)	Duration (days)	Cost (dollars)	Act. weight (%) (w_{t_i})	QI k = 1		QI k = 2		QI k = 3		Quality
					IW ($w_{t_i,k}$)	QP ($Q_{t_i,k}^n$)	IW ($w_{t_i,k}$)	QP ($Q_{t_i,k}^n$)	IW ($w_{t_i,k}$)	QP ($Q_{t_i,k}^n$)	
12	1	22	2000	3	50	99	35	98	15	95	2.9415
	2	24	1750			89		85		87	2.6190
	3	28	1500			70		71		79	2.1510
	4	30	1000			62		61		63	1.8540
13	1	14	4000	7	40	99	40	96	20	97	6.8180
	2	18	3200			73		71		76	5.0960
	3	24	1800			60		62		63	4.2980
14	1	9	3000	6	80	100	10	95	10	98	5.9580
	2	15	2400			79		82		81	4.7700
	3	18	2200			63		67		66	3.8220
15	1	16	3500	7	70	100	30	98	0	NA	6.9580
16	1	20	3000	3	30	97	30	96	40	98	2.9130
	2	22	2000			89		85		87	2.6100
	3	24	1750			81		79		78	2.3760
	4	28	1500			72		73		74	2.1930
	5	30	1000			67		60		62	1.8870
17	1	14	4000	6	70	98	20	97	10	99	5.8740
	2	18	3200			73		75		72	4.3980
	3	24	1800			62		65		61	3.7500
18	1	9	3000	5	30	98	45	99	25	94	4.8725
	2	15	2400			75		77		71	3.7450
	3	18	2200			63		66		67	3.2675

Notes: QI: Quality indicator; QP: Quality performance; NA: Not available; IW: Indicator weight.

Table 2. CAMODE-TCQT's parameter setting

Input parameters	Notation	Setting
Number of decision variables	D	18
Population size	NP	100
The crossover probability	CR	0.9
Scaling factor	F	0.5
Maximum size of Archive	N_{max}	100
Mutant selection	ms	1/3
Maximum generation	G_{max}	300

C-metric is defined as the mapping between the ordered pair (S_1, S_2) and the interval $[0,1]$:

$$C(S_1, S_2) = \frac{|\{a_2 \in S_2; \exists a_1 \in S_1 : a_1 \leq a_2\}|}{|S_2|} \quad (21)$$

Provided that $C(S_1, S_2) = 1$, all solutions in S_2 are dominated by or equal to solutions in S_1 . If $C(S_1, S_2) = 0$, then none of the solutions in S_2 are covered by S_1 . Both $C(S_1, S_2)$ and $C(S_2, S_1)$ should be checked in the comparison because C-metric is not symmetrical in its arguments (Wang, Singh 2009). Table 5 illustrates comparison results among four algorithms in

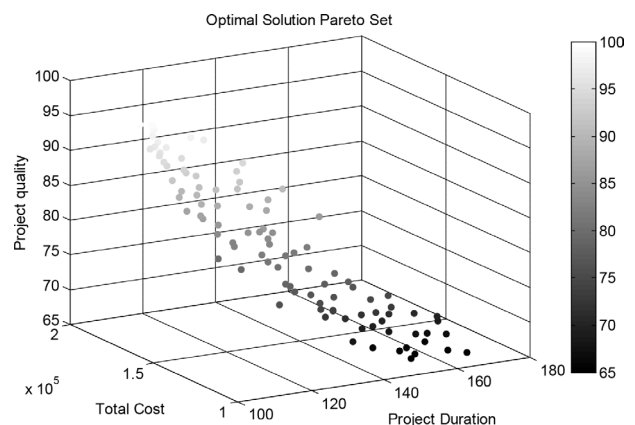


Fig. 7. Time-Cost-Quality trade-off Pareto front using CA-MODE

terms of C-metric, where A1, A2, A3 and A4 indicate CA-MODE, MODE, MOPSO and NSGA-II. It can be seen that MODE, MOPSO and NSGA-II on average have over 42%, 76% and 74% solutions to be dominated by CA-MODE, respectively.

2. Spread (SP): This indicator (Wang *et al.* 2010) is to measure the extent of spread achieved among the non-dominated solutions. Its mathematical definition may be given as:

Table 3. Best non-dominated solutions obtained by CA-MODE-TCQT

Solutions	Partial set	Optimal resource utilization options	Project performance		
			Time (days)	Cost (dollars)	Quality (%)
1	Sorted by Time	[1.2.1.1.2.1.1.2.1.1.1.1.1.1.1.2.1.1]	104	164715	96.17
2		[1.1.1.2.1.1.1.1.1.1.1.1.1.1.1.1.1.1]	104	158820	95.03
3		[1.3.1.1.2.1.1.3.1.1.1.1.1.1.1.5.1.1]	104	163100	95.01
4	Sorted by Cost	[5.5.3.3.4.3.3.5.4.2.3.4.3.3.1.5.3.3]	159	99870	65.24
5		[3.5.2.3.4.3.3.1.1.1.3.1.3.2.1.4.2.1]	120	105570	72.69
6		[1.5.3.3.4.3.3.5.1.1.3.1.3.3.1.5.1.1]	114	105270	71.55
7	Sorted by Quality	[1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.1]	104	168820	97.63
8		[1.1.1.1.1.1.1.2.3.1.1.1.1.1.1.2.1.1]	109	167695	97.06
9		[1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.2.1.1]	104	167820	97.33

Table 4. Result comparison between CAMODE-TCQT and benchmarked algorithms

Model	Case	Optimal Resource Utilization Options	Project Performance		
			Time (days)	Cost (dollars)	Quality (%)
CAMODE-TCQT	1	[1.2.1.1.2.1.1.2.1.1.1.1.1.1.1.2.1.1]	104	164715	96.17
	2	[1.1.1.2.1.1.1.1.1.1.1.1.1.1.1.1.1.1]	104	158820	95.03
	3	[1.5.3.3.4.3.3.5.1.1.3.1.3.3.1.5.1.1]	114	105270	71.55
	4	[5.5.3.3.4.3.3.5.4.2.3.4.3.3.1.5.3.3]	159	99870	65.24
	5	[1.1.1.1.1.1.1.1.1.1.1.1.1.1.1.2.1.1]	104	167820	97.33
Previous findings	1	[1.1.1.1.2.1.1.1.1.1.1.1.1.1.1.1.1.1]	104	166320	95.00
	2	[1.5.3.3.4.3.3.5.1.1.3.1.3.2.1.5.1.1]	114	105470	71.00
	3	[2.3.1.1.2.3.1.1.1.1.1.1.1.1.1.3.1.1]	115	141620	90.00
	4	[2.5.1.3.4.2.3.3.1.1.1.1.3.1.1.5.1.1]	109	121350	77.00
	5	[1.5.1.3.4.3.3.5.1.1.2.1.3.2.1.5.3.1]	124	104620	72.00
NSGA-II-TCQT	1	[1.1.2.3.1.1.1.1.1.1.1.1.1.1.1.1.1.1]	104	153320	92.20
	2	[1.2.2.3.1.1.2.1.1.1.2.1.2.1.1.1.1.1]	104	145820	87.29
	3	[5.3.1.3.4.3.3.2.4.2.1.3.3.2.1.5.3.3]	157	102915	71.56
	4	[5.4.1.3.4.3.3.1.3.1.1.2.3.1.1.4.3.1]	141	104850	74.88
	5	[1.1.2.2.1.1.1.1.1.1.1.1.1.1.1.1.1.1]	104	158320	93.53
MOPSO-TCQT	1	[1.3.1.3.1.1.3.3.1.1.1.1.1.3.1.2.1.1]	104	142815	87.39
	2	[1.4.1.3.1.2.1.2.1.1.1.1.1.1.4.1.1.1]	104	156020	89.52
	3	[5.5.3.3.4.3.3.4.4.1.2.1.3.2.1.4.3.3]	151	101758	68.01
	4	[3.4.3.3.4.3.3.1.2.1.1.4.3.3.1.4.3.2]	143	102160	68.57
	5	[2.2.1.1.1.1.1.2.1.2.1.1.1.1.1.3.1.1]	105	166560	96.05
MODE-TCQT	1	[1.3.1.1.2.1.1.3.1.1.1.1.1.1.1.5.1.1]	104	163100	95.10
	2	[1.5.3.3.4.3.3.5.1.1.3.1.3.3.1.5.1.1]	114	105270	71.55
	3	[1.1.1.2.2.3.1.2.1.1.1.1.1.1.1.2.1.1]	114	133315	90.06
	4	[2.5.1.3.4.2.3.2.1.1.3.1.3.2.1.5.1.1]	109	120615	77.01
	5	[5.5.3.3.4.3.3.5.1.1.1.1.3.2.1.5.1.1]	124	104420	72.08

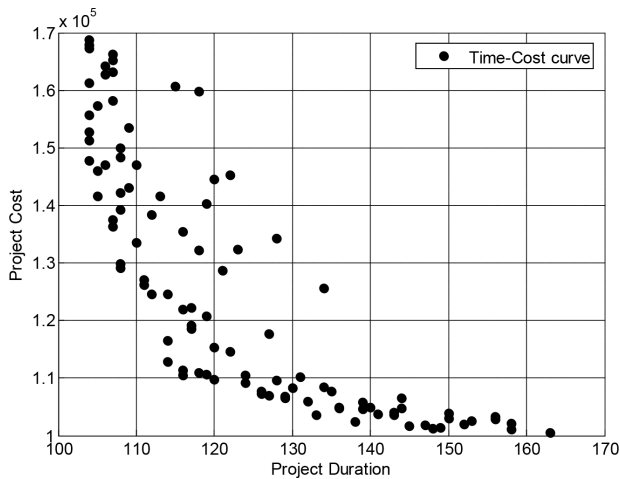


Fig. 8. Time-Cost trade-off analysis

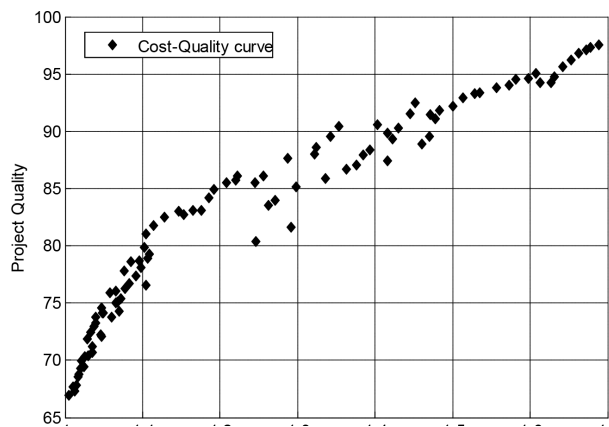


Fig. 9. Cost-Quality trade-off analysis

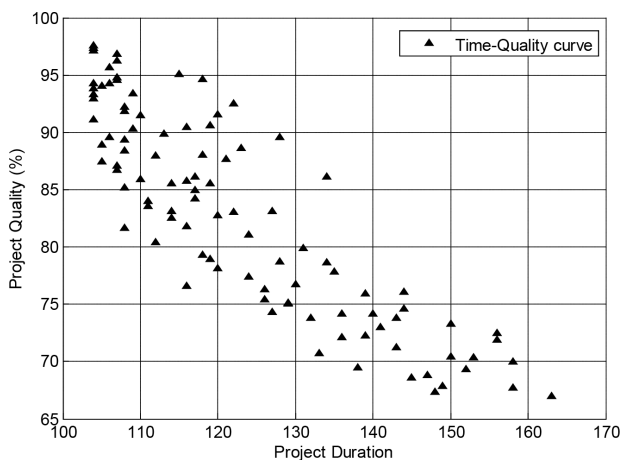


Fig. 10. Time-Quality trade-off analysis

Table 5. Comparison of C-metric for different algorithms

	C(A1, A2)	C(A2, A1)	C(A1, A3)	C(A3, A1)	C(A1, A4)	C(A4, A1)
Best	0.70	0.15	0.88	0.14	0.93	0.11
Wor.	0.28	0.01	0.58	0.00	0.39	0.00
Avg.	0.42	0.08	0.76	0.04	0.74	0.03
Std.	0.15	0.05	0.09	0.04	0.16	0.04

Table 6. Comparison of SP-metric for different algorithms

	CA-MODE	MODE	MOPSO	NSGA-II
Best	0.6931	0.7308	0.7043	0.9761
Worst	0.8461	1.0670	0.9641	1.2699
Avg.	0.7671	0.8688	0.7848	1.1005
Std.	0.0428	0.1010	0.0957	0.0876

Table 7. Comparison of HV-metric for different algorithms

	CA-MODE	MODE	MOPSO	NSGA-II
Best	0.9998	0.9775	0.9832	0.9533
Worst	0.9165	0.9158	0.9000	0.9000
Average	0.9579	0.9481	0.9325	0.9262
Std.	0.0291	0.0242	0.0287	0.0194

$$SP = \frac{\sum_{i=1}^k d(E_i, \Omega) + \sum_{X \in \Omega} |d(X, \Omega) - \bar{d}|}{\sum_{i=1}^k d(E_i, \Omega) + (|\Omega| - k) \bar{d}} \quad (22)$$

where Ω is a set of solutions, (E_1, \dots, E_k) are k extreme solutions in the set of true Pareto-front PF and

$$d(X, \Omega) = \min_{Y \in \Omega, Y \neq X} \|F(X) - F(Y)\|; \quad \bar{d} = \frac{1}{|\Omega|} \sum_{X \in \Omega} d(X, \Omega).$$

A value of zero for this metric indicates all members of the Pareto optimal set are equidistantly spaced. A smaller value of SP indicates a better distribution and diversity of the non-dominated solutions. Table 6 shows that the comparison of the spread metric for different algorithms. It can be seen that the average performance of CA-MODE is superior to the other algorithms.

3. Hyper-volume (HV): This indicator calculates the volume (in the objective space) covered by members of a non-dominated set of solutions Ω for problem where all objectives are to be minimized (Zitzler *et al.* 2003; Wu *et al.* 2010). Mathematically, for each solution $X_i \in \Omega$; a hypercube v_i is constructed with a reference point W and the solution X_i as the diagonal corners of the hypercube. The reference point can be found simply by constructing a vector of worst objective function values. Thereafter, a union of all hypercubes is found and its HV is calculated:

$$HV = \bigcup_{i=1}^{|\Omega|} v_i \quad (23)$$

Algorithms with larger HV values are desirable. The HV value of a set of solutions is normalized by a references

set of Pareto optimal solution with same reference point. After normalization, the HV values are confined in range [0,1]. Table 7 shows the comparison results among different algorithms in term of HV . From Table 7 we can see that the proposed model obtains the largest HV values, which means that CA-MODE has better convergence and diversity performance than the other algorithms.

Conclusions

This paper presents a novel multi-objective optimization model, namely CA-MODE, for solving Construction Project Time-Cost-Quality Trade-off problem. The new optimization model utilizes chaos to scatter the individuals over the searching place; this technique can significantly enhance the diversity of the initial population and generate the potential candidate right from the very beginning of the algorithm. The elitist reservation strategy is also adopted. An external archive is used to store the Pareto optimality found so far in the whole evolution process. Moreover, an adaptive mutation operation is introduced to balance diversity and convergence of proposed algorithm. The experiment results revealed that the proposed CA-MODE approach is efficient for solving multi-objective TCQT problems where multi-objective Pareto optimal solutions can be found in one simulation run. Compared with other multi-objective evolutionary algorithms MODE, MOPSO, NSGA-II and previous findings, the CA-MODE has better diversity characteristics, and yields better compromise solutions and higher degree of satisfaction.

The Pareto front generated by the proposed algorithm which is useful information for decision makers to further determine the optimal design under the consideration of project time, cost and quality trade-offs. It shows no difference if no references on time, cost and quality for the decision makers are selected. If the decision-makers have certain preferences on time, cost and quality, they can quickly make decision according to the sorted specific factors from approach.

The proposed model CA-MODE has the vast application potential to solve different multi-objective optimization problems in construction management, since it can be easily modified to deal with these problems. To name a few, examples include the trade-off among time, cost and environmental; performance, cost and reliability in engineering design.

References

- Babu, A. J. G.; Suresh, N. 1996. Project management with time, cost, and quality considerations, *European Journal of Operational Research* 88(2): 320–327. [http://dx.doi.org/10.1016/0377-2217\(94\)00202-9](http://dx.doi.org/10.1016/0377-2217(94)00202-9)
- Bedri Ozer, A. 2010. CIDE: chaotically initialized differential evolution, *Expert Systems with Applications* 37(6): 4632–4641. <http://dx.doi.org/10.1016/j.eswa.2009.12.045>
- Burns, S. A.; Liu, L.; Feng, C.-W. 1996. The LP/IP hybrid method for construction time-cost trade-off analysis, *Construction Management and Economics* 14(3): 265–276. <http://dx.doi.org/10.1080/014461996373511>
- Cheng, M.-Y.; Huang, K.-Y. 2010. Genetic algorithm-based chaos clustering approach for nonlinear optimization, *Journal of Marine Science and Technology* 18(3): 435–441.
- Coello Coello, C. A. 2006. Evolutionary multi-objective optimization: a historical view of the field, *Computational Intelligence Magazine, IEEE* 1(1): 28–36. <http://dx.doi.org/10.1109/MCI.2006.1597059>
- Das, S.; Suganthan, P. N. 2011. Differential evolution: a survey of the state-of-the-art, *IEEE Transactions on Evolutionary Computation* 15(1): 4–31. <http://dx.doi.org/10.1109/TEVC.2010.2059031>
- De, P.; James Dunne, E.; Ghosh, J. B.; Wells, C. E. 1995. The discrete time-cost tradeoff problem revisited, *European Journal of Operational Research* 81(2): 225–238. [http://dx.doi.org/10.1016/0377-2217\(94\)00187-H](http://dx.doi.org/10.1016/0377-2217(94)00187-H)
- Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Transactions on Evolutionary Computation* 6(2): 182–197. <http://dx.doi.org/10.1109/4235.996017>
- El-Rayes, K.; Moselhi, O. 2001. Optimizing resource utilization for repetitive construction projects, *Journal of Construction Engineering and Management* 127(1): 18–27. [http://dx.doi.org/10.1061/\(ASCE\)0733-9364\(2001\)127:1\(18\)](http://dx.doi.org/10.1061/(ASCE)0733-9364(2001)127:1(18))
- El-Rayes, K.; Kandil, A. 2005. Time-cost-quality trade-off analysis for highway construction, *Journal of Construction Engineering and Management* 131(4): 477–486. [http://dx.doi.org/10.1061/\(ASCE\)0733-9364\(2005\)131:4\(477\)](http://dx.doi.org/10.1061/(ASCE)0733-9364(2005)131:4(477))
- Elmaghraby, S. E. 1993. Resource allocation via dynamic programming in activity networks, *European Journal of Operational Research* 64(2): 199–215. [http://dx.doi.org/10.1016/0377-2217\(93\)90177-O](http://dx.doi.org/10.1016/0377-2217(93)90177-O)
- Feng, C.; Liu, L.; Burns, S. 1997. Using genetic algorithms to solve construction time-cost trade-off problems, *Journal of Computing in Civil Engineering* 11(3): 184–189. [http://dx.doi.org/10.1061/\(ASCE\)0887-3801\(1997\)11:3\(184\)](http://dx.doi.org/10.1061/(ASCE)0887-3801(1997)11:3(184))
- Fonseca, C. M.; Fleming, P. J. 1995. An overview of evolutionary algorithms in multiobjective optimization, *Evolutionary Computation* 3(1): 1–16. <http://dx.doi.org/10.1162/evco.1995.3.1.1>
- Hegazy, T. 1999. Optimization of construction time-cost trade-off analysis using genetic algorithms, *Canadian Journal of Civil Engineering* 26(6): 685–697. <http://dx.doi.org/10.1139/199-031>
- Huang, V. L.; Suganthan, P. N.; Qin, A. K.; Baskar, S. 2008. *Multiobjective differential evolution with external archive and harmonic distance-based diversity measure*. School of Electrical and Electronic Engineering Nanyang, Technological University Technical Report. 25 p.
- Khang, D. B.; Myint, Y. M. 1999. Time, cost and quality trade-off in project management: a case study, *International Journal of Project Management* 17(4): 249–256. [http://dx.doi.org/10.1016/S0263-7863\(98\)00043-X](http://dx.doi.org/10.1016/S0263-7863(98)00043-X)
- Li, H.; Cao, J.; Love, P. 1999. Using machine learning and GA to solve time-cost trade-off problems, *Journal of Construction Engineering and Management* 125(5): 347–353. [http://dx.doi.org/10.1061/\(ASCE\)0733-9364\(1999\)125:5\(347\)](http://dx.doi.org/10.1061/(ASCE)0733-9364(1999)125:5(347))
- Price, K. V.; Storn, R. M.; Lampinen, J. A. 2005. *Differential evolution. A practical approach to global optimization*. Berlin Heidelberg: Springer-Verlag. 538 p.
- Reddy, M.; Kumar, D. 2007. Multiobjective differential evolution with application to reservoir system optimization, *Journal of Computing in Civil Engineering* 21(2): 136–146. [http://dx.doi.org/10.1061/\(ASCE\)0887-3801\(2007\)21:2\(136\)](http://dx.doi.org/10.1061/(ASCE)0887-3801(2007)21:2(136))
- Siemens, N. 1971. A simple CPM time-cost tradeoff algorithm, *Management Science* 18(3B): 354–363.
- Storn, R. M.; Price, K. 1997. Differential evolution – a simple and efficient heuristic for global optimization

- tion over continuous spaces, *Journal of Global Optimization* 11: 341–359.
<http://dx.doi.org/10.1023/A:1008202821328>
- Wang, L.; Singh, C. 2009. Reserve-constrained multiarea environmental/economic dispatch based on particle swarm optimization with local search, *Engineering Applications of Artificial Intelligence* 22(2): 298–307.
<http://dx.doi.org/10.1016/j.engappai.2008.07.007>
- Wang, Y.-N.; Wu, L.-H.; Yuan, X.-F. 2010. Multi-objective self-adaptive differential evolution with elitist archive and crowding entropy-based diversity measure, *Soft Computing – A Fusion of Foundations, Methodologies and Applications* 14(3): 193–209.
- Wu, L. H.; Wang, Y. N.; Yuan, X. F.; Zhou, S. W. 2010. Environmental/economic power dispatch problem using multi-objective differential evolution algorithm, *Electric Power Systems Research* 80(9): 1171–1181.
<http://dx.doi.org/10.1016/j.epsr.2010.03.010>
- Xuexia, Z.; Weirong, C.; Suganthan, P. N. 2012. Optimal multi-objective reactive power dispatch considering static voltage stability based on dynamic multi-group self-adaptive differential evolution algorithm, in *The 2nd International Conference on Intelligent System Design and Engineering Application (ISDEA)*, 6–7 January 2012, Sanya, Hainan, 1448–1456.
- Yang, I. 2007. Using elitist particle swarm optimization to facilitate bicriterion time-cost trade-off analysis, *Journal of Construction Engineering and Management* 133(7): 498–505.
[http://dx.doi.org/10.1061/\(ASCE\)0733-9364\(2007\)133:7\(498\)](http://dx.doi.org/10.1061/(ASCE)0733-9364(2007)133:7(498))
- Zahraie, B.; Tavakolan, M. 2009. Stochastic time-cost-resource utilization optimization using nondominated sorting genetic algorithm and discrete fuzzy sets, *Journal of Construction Engineering and Management* 135(11): 1162–1171.
[http://dx.doi.org/10.1061/\(ASCE\)CO.1943-7862.0000092](http://dx.doi.org/10.1061/(ASCE)CO.1943-7862.0000092)
- Zhang, H.; Xing, F. 2010. Fuzzy-multi-objective particle swarm optimization for time–cost–quality tradeoff in construction, *Automation in Construction* 19(8): 1067–1075.
<http://dx.doi.org/10.1016/j.autcon.2010.07.014>
- Zhou, A.; Qu, B.-Y.; Li, H.; Zhao, S.-Z.; Suganthan, P. N.; Zhang, Q. 2011. Multiobjective evolutionary algorithms: a survey of the state of the art, *Swarm and Evolutionary Computation* 1(1): 32–49.
<http://dx.doi.org/10.1016/j.swevo.2011.03.001>
- Zitzler, E.; Thiele, L. 1999. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach, *IEEE Transactions on Evolutionary Computation* 3(4): 257–271.
<http://dx.doi.org/10.1109/4235.797969>
- Zitzler, E.; Thiele, L.; Laumanns, M.; Fonseca, C. M.; Fonseca, V. G. D. 2003. Performance assessment of multiobjective optimizers: an analysis and review, *Transactions on Evolutionary Computation* 7(2): 117–132.
<http://dx.doi.org/10.1109/TEVC.2003.810758>

Min-Yuan CHENG. Distinguished Professor of Department of Civil and Construction Engineering at the National Taiwan University of Science and Technology. An Editorial Board member of *Automation in Construction* Journal. His research interests include construction business process re-engineering, geographic information system, construction automation and e-business for construction management, project management information system, artificial intelligence applications in construction management.

Duc-Hoc TRAN. PhD candidate, Department of Civil and Construction Engineering National Taiwan University of Science and Technology. Research interests: computational intelligence and applications of evolutionary algorithm in construction management.

Minh-Tu CAO. PhD candidate of Department of Civil and Construction Engineering at the National Taiwan University of Science and Technology. He received Master's degree in construction management (2012). His research interests include applications of artificial intelligence in construction management, construction economics, and decision-making support systems.