Canadian Journal of Civil Engineering

An Evolutionary Stochastic Discrete Time-cost Trade-off Method

Journal:	<i>Canadian Journal of Civil Engineering</i>
Manuscript ID	cjce-2018-0352.R1
Manuscript Type:	Article
Date Submitted by the Author:	24-Oct-2018
Complete List of Authors:	Hussein, Bahaa; Concordia University, Building Engineering Moselhi, Osama; Concordia University, Department of Building, Civil & Environmental Engineering
Keyword:	computational methods < type of paper to review, theoretical analysis < type of paper to review, constr. management < Construction
Is the invited manuscript for consideration in a Special Issue? :	Not applicable (regular submission)

SCHOLARONE™
Manuscripts

An Evolutionary Stochastic Discrete Time-cost Trade-off Method

Bahaa Hussein, Osama Moselhi

Department of Building, Civil and Environmental Engineering, Concordia University,
Montréal, Canada

E-mail addresses: b_alsaidi@yahoo.com (Bahaa Hussein);

osama.moselhi@concordia.ca (Osama Moselhi);

Correspondent Author: Bahaa Hussein

E-mail address: b_alsaidi@yahoo.com

Address: Sir George Williams Campus, 1455 De Maisonneuve Blvd. W.

Montreal, Quebec, Canada, H3G 1M8.

Telephone Number: +97366710151

Abstract:

This study introduces a newly developed method for optimized time-cost trade-off under uncertainty. It identifies the optimal execution mode for each project activity that results in minimizing the overall project cost and/or duration while satisfying a specified joint confidence level of both time and cost. The method uses an evolutionary-based algorithm along with a design generator of experiments and blocking techniques. The developed method accounts for managerial flexibility towards the selection of execution modes. This accommodates experience-based judgement of project managers in this process. Hence, the second fold of the developed method is a completely randomized experiment module that depicts the main effect of changing an activity mode on the project total cost and overall duration. The method provides the decision maker a guideline for making well-informed implementation strategies. The results obtained demonstrate benefits and accuracy of the developed method and its applicability for large scale projects.

Keywords: Evolutionary optimization methods; Discrete time-cost trade-off; Uncertainty analysis; Joint confidence level.

1 Introduction

The diversity of available means, resources, materials, technologies and methods for executing an activity provides alternative implementation modes that can be assigned to individual activities in construction projects. In other words, an activity may be executed using alternative materials or with different forms of resources. For example, methods and materials used in earth backfilling activities, dewatering methods for deep excavations. Selection amongst those available modes requires the project manager's decision. The choice of activity's mode may have a different duration for that activity and a different direct and indirect costs. Time-cost trade-off problem (TCTP) analysis on construction projects are often used to explore available alternative modes in an attempt to find the optimal overall project duration and/or total cost. Solving for this problem can be quite cumbersome especially on large and complex projects. Solving the TCTP have been intensively researched since 1960s. However, most of these methods are deterministic and consider a single cost and duration values for the activities involved. As well, these methods generally assume a linear relationship between time and cost and cannot adequately solve discrete time-cost trade-off problem (DTCTP) since the relationship between discrete modes may not be modeled. Solving the DTCTP have been categorized as non-deterministic polynomial-time hard (NP-Hard) (De et al. 1995). In real world, the alternative modes, especially those that are new to the construction industry or new technologies, are not proven and come with higher uncertainty estimate of their time and cost. Therefore, some modes for a single activity can be riskier than others and have a probability distribution function with a larger range. When the overlaps among the modes are significant the problem becomes hard to solve using deterministic

methods (Zheng and Kumaraswamy 2005). When incorporating the uncertainty of each mode into the equation, the difficulty of solving the DTCTP becomes exponentially larger. This is due to now becoming a stochastic problem where each mode attribute of time and cost are represented by a probability rather than a deterministic value.

There exist many techniques for the TCTP. Those can be categorized into three areas namely, mathematical programming, heuristics and simulation. Examples of mathematical techniques that have used linear programming were introduced by Kelly (1961). Patterson and Huber (1974) and Sakellariopoulos and Chassiakos (2004) used the integer programming technique to tackle discrete relationships, however, integer programming requires an extensive computational effort and fail to be applied on large project networks. Elmaghraby (1993) and De et al. (1995) introduced dynamic programming techniques for special class projects that can be decomposed into parallel or a series of sub-projects.

Heuristic methods have dominated much of the research efforts because of their ability to find a good solution within a reasonable computational effort. However, such methods lack mathematical rigor and finding the optimal solution is not guaranteed. Examples of heuristic techniques include the structural stiffness method introduced by Moselhi (1993), a branch-and-bound approach introduced by Demeulemesster et al (1996), and an optimization method using the maximal flow theory introduced by Liu et al (1995). Some of the more recent researches that have implemented heuristic methods in TCT and decision-making problems are Chiu and Chiu (2005), Vanhoucke and Debels (2005), Hauser (2014) and, Pendharkar (2015).

Solving for the stochastic network has been proven to be a hard problem as the problem would include an extensive amount of computations for numerical integration from a complete enumeration of all possible solutions. Therefore, Simulation techniques have become more popular in recent researches; however, most of those studies have been focused on either the project duration or costs while fewer studies have been conducted to optimize the project in a multi-objective approach for simultaneous cost and schedule optimization. Furthermore, fewer of those researches have tackled the consideration of uncertainties in TCTP. Zheng and Kumaraswamy (2005), Leu and Yang (1998), Moselhi and Rofigari (2013), Moselhi and Alshibani (2013), and Kalhor et al. (2011) proposed a fuzzy method for a stochastic TCT problem. Fuzzy set theory dominated those researches due to its capacity to produce the results in fewer runs. However, Monte Carlo simulation techniques can provide more statistical information exploring the uncertainties of activity durations and costs.

Meta-heuristic optimization algorithms have gained popularity amongst recent researches. The methods typically begin by defining a function for a set of independent variables to obtain a global minimum or maximum for the function output. The function is solved by assigning initial values for the variables then evolve to iterating those variables to determine variance solutions from the search space. The iterative process is performed until a termination criterion is reached. Depending on the function complexity and the total size of the search space, the meta-heuristic approach may only visit some of the possible solutions, therefore, there is no guarantee that these methods can find the optimal solution. Examples of the most popular meta-heuristic approaches are Genetic Algorithms (GA), Simulated Anneal (SA) and, Ant Colony Optimization algorithm

(ACO). Examples of such techniques were introduced by Ng and Zhang (2008) and, Afshar et al. (2009). Examples of recent research including stochastic analysis are those by Eshtehardian et al. (2009) and (Zahraie and Tavakolan 2009) using fuzzy set theory and genetic algorithms, Gutjahr et al. (2000) using a stochastic model based on a branch-and bound approach, Aghaie and Mokhtari (2009) and Kalhor et al. (2011) applying a modified ant colony optimization and Monte Carlo simulation, Yang et al. (2013) using the particle swarm optimization approach and, Ke (2014) using a stochastic simulation hybridized model with genetic algorithms.

The objective of this study is to provide project managers and decision makers a holistic, integrated cost and time analysis tool that can answer to enquiries such as, what are the optimal modes of construction that result in the multi-objective solution for schedule compression, cost reduction or the optimal joint cost and schedule compression that meets a targeted confidence level. The second objective is to find the main effect of interchanging modes and their relationship to the objective function. Answers to those objectives will increase the project manager's probability of making correct decisions that achieve successful project delivery on budget and on time. The method is built on a Monte-Carlo simulation of the project to estimate the joint confidence level of both time and cost. Optimal solutions are obtained by performing an evolutionary full factorial design of experiment matrix using a blocking technique and a complete randomized design of experiments. The results are represented in main effect and interaction plots to facilitate needed rational support for making well-informed execution decisions.

2 The Problem

Time–cost trade-off problem is not new. It focusses on selecting optimal implementation modes for the individual project activities so as to ultimately minimize the overall project time and/or its total cost. However, large-scale construction projects often include numerous activities, each of which comes with different possible implementation modes yielding different costs and durations. Knowing that uncertainty exists in those costs and durations the problem then becomes almost impossible to stochastically evaluate all possible combinations of modes within a reasonable short time with a reasonable cost of computations. A full factorial design of experiments is the matrix of all combinations of possible modes also known as the Cartesian product matrix, where each combination can be in resemblance to a chromosome. For example, a small project modeled by only 15 activities each having 5 discrete modes, the number of possible combination for a full factorial design of experiment matrix of modes to solve the problem is more than 30 billion combinations. The number of chromosomes increase exponentially with the increase in the number of activities in the project network and/or with the increase in the number of alternative delivery modes at each activity. Therefore, blocking and randomization techniques are used in the developed method to narrow the search space. Numerical examples are used to demonstrate the benefits and accuracy of the developed method and to highlight its superior capabilities, particularly for large project networks.

3 Evolutionary stochastic discrete time- cost trade -off method

The following sections present the developed object-oriented programming for the Evolutionary Stochastic Discrete Time-Cost Trade-Off Method (ESDTCT).

The developed method is composed of two modules as shown in Figure 1. The first identifies the supreme solution representing the combination of modes that yields the optimal cost and/or schedule minimization for a specified joint confidence level of both time and cost. This combination is in resemblance to a chromosome where the gens are the modes of the different activities. While the second module is a random stochastic search that depicts the main effect of changing a mode within a chromosome on the total cost and overall duration of the project for a specified joint confidence level of both time and cost. The following sections provide the background for the set of processes used simultaneously in setting the computational operations for the developed method.

[Fig. 1 about here.]

The developed method adopts biomedical terms amongst others such as “Genes”, “Evolution”, “Chromosomes”. Those terms have been interchangeably used in GA and are further defined here to put those terms in context of the developed method. Figure 2 shows a graphical representation of those terms.

- Gene (Activity): is an element position of a chromosome. Where this element represents an individual activity within the project network.
- Mode: is a variant form of a gene; it represents the resources that are available for use to execute an individual activity.
- Supreme mode: is the best mode assigned to an activity that results in the optimal solution of the problem. Where the term “best” is defined by the optimality definition for the objective function.

- Chromosome: is one such solution to the given problem. It represents the combination of a string of modes for each activity in the network. Accordingly, it depicts one possible solution in the solution space.
- Supreme chromosome: is the chromosome that results in the most optimal solution of the problem.
- Population: is all the possible chromosomes available for solving the problem.
- Generation: is a sample of the population that represent a group of chromosomes sharing specific characters.
- Evolution: is the gradual crossover of supreme modes to the next generation. The evolutionary technique is used to evolve the supreme modes surfacing from previous generations.
- Blocking: is a design of experiments (DOE) technique. The technique is used to focus the analysis on a set of factors (or primary activities) having more relevant influence on the objective in a particular generation. Those factors assume all of their possible values, while all the other factors in the design are kept constant and called here as nuisance factors.
- Primary activity: is an element position of a chromosome being investigated for inclusion of its supreme mode in a particular generation. A set of primary activities are identified at each generation using the immediate reverse dominator tree (IRDT).
- Immediate reverse dominator tree (IRDT): is the set of activities that immediately precedes primary activities from a previous generation. The finish activity is the root of the reversed tree. An illustration example of the IRDT is shown in Figure 3 where

a total of two IRDT generations exist to walk backwards through the entire network activities.

[Fig. 2 about here.]

[Fig. 3 about here.]

The Joint Cost and Schedule Confidence Level (JCL) of both cost and schedule developed by NASA (Coonce 2009) is used in the developed method to estimate the joint confidence level of a project overall cost and time. The method integrates the analysis of a project cost, schedule, and uncertainty. The objective here is to estimate the project's cost and time for a selected probability confidence level. For example, a 50% confidence represents the probability for mutually estimating 50% confidence in cost and 50% confidence in schedule. This is unlike estimating independently the 50% cost confidence or estimating the 50% schedule confidence. NASA Cost Estimating Handbook (Version 4.0, 2015) recommends the implementation of JCL, in fact, NASA JCL policy (NPR 7120.5E), mandates projects to perform a JCL with a recommended 70% probability for estimating both cost and schedule (Comstock et al. 2015).

The methodology for developing a JCL starts with building a probabilistic cost-loaded schedule and systematically integrating cost, schedule, and uncertainty. The method facilitates the establishment of expectations and probabilities of meeting those expectations. It also offers a holistic view to achieve cost and schedule goals as well as contingency estimation for cost and schedule under a specified confidence level.

A Pareto frontier curve is developed by joining all the possible combinations of cost and schedule solutions that satisfy a selected confidence level. Figure 4 shows a typical

scatter plot for all possible solutions resulting from the Monte-Carlo simulation of the cost loaded project schedule iterated for the possible values represented by the uncertainty profiles of the individual activities costs and durations, the chart also shows a 50% JCL Pareto frontier curve.

[Fig. 4 about here.]

3.1 Defining Optimal Solution.

The developed ESDTCT method objective function requires reducing the JCL analysis to a single solution. This single solution is determined based on the ultimate objective set forth by the analyst as follows:

1. Cost reduction: this is selected when the objective of the analysis is set to solve for the lowest cost solution falling on the frontier curve.
2. Schedule compression: this is selected when the objective of the analysis is set to solve for the lowest project duration solution falling on the frontier curve.
3. Joint costs and schedule compression: this is selected when the objective of the analysis is set to solve for the optimal solution on the frontier curve while concurrently compressing the project cost and schedule. This value is defined as the vertex solution on the frontier curve. The calculation method for finding the vertex is further detailed in the following sections. Figure 4 shows an illustration of the three objective solutions.

A calculation procedure is developed to determine the reduced vertex solution in the case of joint costs and schedule compression as follows:

The cost and duration values are of different units of measurement and magnitudes; therefore, it is necessary to normalize the values so that each is scaled to values in the same range. The solutions found on the Pareto frontier are normalized to the range [0,1]. To do this, the nadir and ideal objective vectors can be used to normalize the objective values. The nadir objective vector was recognized and used by the multiple criteria decision-making researchers since the early 1970s. The nadir objective vector is a point constructed from the intersection of the maximum cost and schedule values from all pareto frontier solutions, while the ideal objective vector is a point constructed from the intersection of the minimum cost and schedule values from all pareto frontier solutions. The pareto frontier cost and schedule values are then normalized to the range [0,1] using the following equations:

$$T_i^{norm} = (T_i - T^{ideal}) / (T^{nadir} - T_i) \quad 1$$

Where T_i^{norm} is the normalized time value for solution $i \in \text{pareto frontier}$, T^{ideal} and T^{nadir} are the minimum and maximum time value respectively among the pareto frontier solutions.

Similarly, the cost values are normalized using the following equation:

$$C_i^{norm} = (C_i - C^{ideal}) / (C^{nadir} - C_i) \quad 2$$

The reduced solution of the JCL is the vertex point of the normalized pareto frontier curve. The vertex point is defined as the solution providing the optimal balance between time and cost compression and is determined as the point having the shortest vector length connecting to the ideal vector point using Pythagorean Theorem:

$$l_i = \sqrt{(T_i^{norm})^2 + (C_i^{norm})^2} \quad 3$$

Where l_i is the solution response vector length for solution $i \in \text{pareto frontier}$. Then the optimal solution i_{optimal} is the vertex point solution that has the least response vector length. Figure 5 shows an illustration for the nadir and ideal vector points for a selected pareto frontier and Figure 6 shows an illustration of the normalized pareto frontier.

[Fig. 5 about here.]

[Fig. 6 about here.]

Throughout the thesis, a total of 100 Monte Carlo simulation runs is chosen to represent the solution space at each individual experiment run. This number is selected for the purpose of minimizing the amount of computations and is found to provide good accuracy of results as demonstrated in the test examples. For small problems, the number of simulation runs can be adjusted to a higher number and conversely the tolerance factor can be tighter. The number of simulation runs have a direct relationship with the total execution run time. Furthermore, a tolerance band factor is applied to the specified JCL to allow for the selection of a larger number of solutions that better represent the pareto frontier. Throughout the study, a tolerance band factor (t) of $\pm 2\%$ is applied. For example, if the specified JCL is 50%, then the selection of the pareto frontier solutions are those that have a JCL falling on the band between 48% to 52%.

3.2 Computational Procedure

The ESDTCT method has two distinct modules, the first is performing a full factorial design of experiments with a blocking technique designated here as ESDTCT^{Exp}. The second module performs a random search of the solution space designated here as ESDTCT^{Rand}. Each of those modules provides answers and insights to the solution space

for the stochastic time-cost trade-off problem. The computational procedure and outputs from those modules are explained in the following sections. The developed modules are implemented in spreadsheet-based application using Google Sheets which is one of the core components of Google cloud applications using a programming language called Google Apps Script (GAS). GAS is a JavaScript that is executed, not in the browser but remotely on the Google cloud. Google BigQuery SQL is used to facilitate the calculation procedure. The application can be executed using any modern browser (preferably Chrome and Firefox), an internet connection running on Windows, Mac OS X or Linux. Instead of running on the client computer, it gets executed in the Google Cloud, therefore the speed of executing the code is not limited by the client computer specifications, rather by the capacity of google servers, which in turn has undergone significant development since a major upgrade in March 2014. Figure 7 shows the developed method network configuration diagram and the integration between the Google cloud apps.

[Fig. 7 about here.]

Evolutionary experiment generation module – ESDTCT^{Exp}

This module is developed to find the supreme chromosome representing the combination of supreme modes that result in the best overall project cost and/or time minimization for a specified joint confidence level. The method intelligently reduces the number of experiments that are required to search for the supreme chromosome amongst the total number of experiments in a complete enumeration of the total solution space. This is achieved by breaking the computation steps into several generations, where each generation is a full factorial DOE matrix with a blocking technique that allows a full

enumeration of all possible combination of modes for primary activities while all other activities in the project network are blocked to a static mode. The primary activities in a generation are selected using the immediate reverse dominator tree (IRDT). The initial generation starts with the completion activity and all its IRDT activities as primary activity and runs backwards in the project network logic to select the next generation of primary activities. By doing so, the number of experiments is reduced to the Cartesian product of the primary activities. The methodology behind this process is to experiment on the mode interchangeability for activities immediately driving the completion activity to determine their supreme modes that result in an optimum objective value. Those supreme modes are memorized and carried forward as static modes in successive generations. A pseudocode for the $\text{ESDTCT}^{\text{Exp}}$ procedure is given in Appendix A.

Random experiments module $\text{ESDTCT}^{\text{Rand}}$

This module is developed to explore the solution space through completely randomized DOE sample from the total solution space. The objective here is to predict, approximately, the impact of mode changes on the overall project cost and time for a specified joint confidence level of time and cost. This provides the decision maker an approximate indication of how the project cost and duration will react to alternate modes that may not be the optimal modes but selected based on the project manager's judgement. A typical main effect plot is illustrated in Figure 8. A pseudocode for the $\text{ESDTCT}^{\text{Rand}}$ procedure is given in Appendix B.

This procedure is run until a user-defined time or cost threshold is reached for the simulation. This procedure is concurrently computed with the experiments generation module procedure.

The schematic diagram in Figure 9 summarizes the ESDTCT modules procedures developed in this study.

[Fig. 8 about here.]

[Fig. 9 about here.]

4 Numerical Example 1: Testing on a small size project under uncertain environment.

To demonstrate the application of the developed method, a simple 18 activities network example is used. The example was originally developed by Feng et al. (1997) and Hegazy (1999) who solved for the TCTP in GA-based application under crisp cost and time values, i.e. deterministic solution. The same example was later extended to account for uncertainty by Eshtehardian et al. (2009) and solved using a multi-objective genetic algorithm (MOGA) using fuzzy time and cost values. Kalhor et al. (2011) also solved the same example using a fuzzy non-dominated archiving ant colony approach (NA-ACO) and confirmed the same results obtained by the aforementioned MOGA. The network configuration of the example is shown in Figure 10. For each activity, there are two to five alternative modes making a total number of 5,9 Billion possible mode alternatives. Each mode has a time and associated cost of which their uncertainties are expressed by a triangular function, defined by the optimistic, most likely and pessimistic estimates as presented in Table 1. The overhead cost is identical to that of Eshtehardian et al. (2009), i.e. triangular probability function with \$185 optimistic, \$200 most likely, and \$235 pessimistic values. The indirect cost rate for the activities was not considered by

Eshtehardian and therefore has been set to zero to allow for proper comparison of the results. The objective of the problem is to identify the supreme chromosome solution that results in schedule compression which satisfies a 50% joint cost and schedule confidence level. The performance criteria are set as that the optimal chromosome result should be no worse than aforementioned optimal values using the MOGA and the NA-ACO approaches.

[Fig. 10 about here.]

[Table. 1 about here.]

The ESDTCT^{Exp} procedure took 5 generations to calculate for the supreme mode of all the activities and accordingly arrive at the supreme chromosome. The overall number of experimental chromosome runs is 844 as compared to the total size of the solution space of 5,9 Billion possible chromosomes. The example was solved in under 3 minutes. The selection of primary activities at each generation is shown in Figure 11. The summary results of each generation are included in Table 2. Due to the extensive size of data, the details about results from each generation are not presented for which a cumulatively 85 thousand simulation iterations were performed.

[Fig. 11 about here.]

[Table. 2 about here.]

It is noted that the supreme chromosome generated by the developed ESDTCT^{Exp} method is different from that reported by Kalhor et al (2011) using the NA-ACO method referenced in Figure 12, where activities 1, 8, 14, 15 and 17 have resulted in a different supreme mode. To compare the two results a hypothesis is made here that the supreme

chromosome resulting from $ESDTCT^{Exp}$ is better than that produced by the MOGA and NA-ACO methods. To test the hypothesis, Monte Carlo simulation is performed to calculate for the JCL pareto frontier for the optimal chromosome reported from the MOGA and NA-ACO methods. Figure 12 shows the difference in those results where the $ESDTCT^{Exp}$ supreme chromosome resulted in a solution with a slightly different cost but a significantly lower overall project duration. Therefore, it can be concluded here that the performance of the developed $ESDTCT^{Exp}$ is superior to that of the MOGA and NA-ACO methods.

The effect of decision maker's appetite for risk acceptance is represented by his/her selection of a joint confidence level (JCL). The higher the JCL, the more confidence the project is in meeting the resultant cost and schedule values. The effect of diverse selection of JCL is tested. Results of using a JCL value of 20%, 50% and 80% are listed in Table 3. The analysis shows that solving the problem for the objective of joint cost and schedule compression resulted in different supreme chromosome combination for different JCL values as noticed in differing modes for activities 2,3,8 and 11.

Another parametric study is also made to test the effect of decision maker's objective settings for the optimization problem. The problem is solved for the three objectives of cost reduction, schedule compression and joint cost-schedule compression. The analysis results are listed in Table 3. It can be noticed that different objectives produced different supreme chromosome combinations. For example, the supreme chromosome at the specified 50% JCL is almost entirely different for each optimization objective. In other words, different supreme chromosome solutions are found more fitting as a result of the objective settings and the specified JCL.

[Fig. 12 about here.]

[Fig. 13 about here.]

[Table. 3 about here.]

To test the developed ESDTCT^{Rand} module, the example is solved to obtain the main effect plot that represents the most influential activities and variables (modes) for the total project cost and time at the selected 50% joint confidence level for the objective of joint overall time and cost compression. A random search of the total solution space is performed, and a total 75 million iterations were made for a 750 thousand random selection of chromosomes in under 4 minutes. This random selection represents 0.01% of the total solution space. The main effect plot for the example is shown in Figure 13.

[Fig. 14 about here.]

By visual analysis of the plot, it can be noticed for example that there are insignificant differences between choosing any of the 5 modes associated to activities 8. The reason for this insignificance differences could be related to any or a combination of the following three items:

- The activity in hand has a relatively small cost value assigned to the modes of this activity as compared to the other project activities or as compared to the overall project cost,
- Another reason can be due to the non-criticality of this activity that does not affect in significance the overall project duration,
- A third reason may be concluded is the significant overlap of the probability distributions among the different modes cost and duration estimates. In other

words, there is insignificant differences in time and cost distributions between the activity modes.

Conversely, activity 4 is observed to have a wide range influence on the objective response vector where the small response vector value indicates a more optimal solution resulting in an expected lower cost and duration of the project. similarly, the interpretations of this influence can be explained by the following items:

- The activity has a relatively significant cost value over the entire project,
- A small overlap or no overlap in some cost and duration estimates among the different modes.
- The activity is on the critical path and variations in durations are highly correlated to the increased project duration and hence the overhead cost.

While there is a more significant reduction in cost and duration when choosing mode 3 for the said activity, it is worth noting here that the main effect plot only explored a small sample of the total solution space and can only provide a rough indication of the effects. For the example in hand, the optimal modes found using the ESDTCT^{Exp} module are generally matching to those modes showing the least response vector visualized in the main effect plot in Figure 13. However, this rough indication from the main effect plots may be more accurate for small projects and less accurate for large projects. Therefore, once the final selection of the desired modes is made, it is recommended to perform a final simulation run to validate the impact on the objectives of total cost and time.

5 Numerical Example 2: Testing on a large size project under uncertain environment

This example is set to test the performance and accuracy of the developed method on a large size project composed of 63-activities each having between 3 to 5 different modes as a result the problem has a total solution space of 1.373 quattuordecillion possible chromosome combinations making the problem impossible to perform exhaustive enumeration and hence in need for an optimization algorithm. This example is adopted from Bettemir (2009). In his PhD thesis the problem was solved in 8 different meta-heuristic algorithms all under certain cost and time values to find the optimum cost reduction solution. The network configuration of the example is shown in Figure 14 and the cost and duration values for each mode is listed as the most likely values in Table 4 and Table 5. The constant daily overhead cost introduced by Bettemir was \$2,300. As to compare the results, the developed $\text{ESDTCT}^{\text{Exp}}$ module is converted to a certain environment by parsing the minimum and maximum values for time and cost as the same as that of most likely. The results are listed in Table 6 along with the results from the variance meta-heuristic algorithms obtained by Bettemir. The supreme chromosome cost and duration values firmly confirm to aforementioned optimal lowest cost using the Particle Swarm Optimization and the Genetic Algorithm Simulated Annealing optimization methods. The $\text{ESDTCT}^{\text{Exp}}$ method was able find the optimal solution by analyzing only 677,649 experiments of the total solution space, that is only $4.94 \times 10^{-35} \%$ which is extremely low and efficient. This analysis confirms the efficiency and accuracy of the results under certain environment in comparison to other optimization algorithms from literature.

[Fig. 15 about here.]

[Table. 4 about here.]

[Table. 5 about here.]

[Table. 6 about here.]

Moreover, this example is extended to test under uncertain environment. The input data for uncertainty in cost and activity duration values is generated and listed Table 4 and Table 5. To the best of our knowledge, this problem exceeds the largest and hardest test cases in literature solved for the DTCTP with uncertainty in both cost and schedule. The objective is set to solved for the optimum cost reduction with a 50% JCL. The uncertainty in the overhead daily cost is (\$2070, \$2300, \$2760) for the optimistic, most likely and pessimistic values respectively. The triangular PDF is used to represent the uncertainty profile for all cost and duration values. The ESDTCT^{Exp} module run time took 11 minutes to solve for the 11 generations and arrive at the supreme chromosome solution. The ESDTCT^{Exp} module results are listed at each generation and for the final supreme chromosome in Table 7.

[Table. 7 about here.]

As data depicts, 20 of the activities had resulted in a different supreme mode within the supreme chromosome structure when compared between the supreme chromosome in certain and uncertainty environments. Those differences indicated that different modes of an activity are more fit to supremacy when influenced by the uncertainty parameters.

The main effect plot is generated from the developed ESDTCT^{Rand} method to allow predicting the effect of mode selection on the objective cost reduction solution at 50% JCL. The main effect plot from a total of 3 million random experiments was computed in

parallel time to the $ESDTCT^{Exp}$ module and shown in Figure 15. The plot provides approximate correlation for the effect of the activity mode interchange.

[Fig. 15 about here.]

6 Conclusion

The developed method presented in this study is designed to solve discrete time–cost trade-off problem under uncertainty. One objective of the method is to select the optimal chromosome that represents the combination of activity modes (assigned resources) while minimizing project time and cost, while concurrently maintaining targeted joint confidence level of time and cost. Another objective of the method is to identify in a main effect plot the most significant modes that influence the ultimate results of time and cost of the project. Monte-Carlo simulation is embedded into the modeling structure of the developed method to account for uncertainties in both time and cost. Two case studies taken from literature are solved to show the applicability of the developed method. The identified optimal solution (chromosome) outperformed those reported by others using various optimization methods. The outcomes prove that, depending on the shapes, range and overlaps of the probability distribution functions of each mode, the method generates different optimal chromosome solutions for different joint confidence levels. Only a small percentage of the solution space was searched to obtain the optimal solutions. This is achieved by decomposing the project network into blocks of activities and limiting the higher-level interactions between the different modes. Utilizing the power of the computational engine of Google’s big-query, the execution time to solve the large size

problem was 11 minutes using the free BigQuery quota limitations¹, the execution time is not linearly proportioned to the size of the model since big queries are run concurrently on the server. Since the run time is reasonably short, it is safe to say that the developed method can be applied for solving larger problems while maintaining the same accuracy. Future work of the authors focusing on adding a knowledge-based component to narrow the search space.

7 Limitations

Although the developed method can handle any project size including a 10,000-activity schedule, it has limitations to the problem complexity. This is mainly determined by the number of relationship logic between the network activities. The size of a Cartesian product matrix for the full factorial experiment enumeration for primary activities in a single generation is governed by the number of primary activities and their respective number of available modes. The maximum size that can be processed is restricted by the Google BigQuery limitation of 2000 maximum bound for concurrent slots per project for on-demand pricing. This limitation however is the current state for the free quota and can be scalable upon request from Google by adding instances of virtual machines to the Google compute engine. With the free quota limitations, we were able to process 8 primary tasks with 5 modes each or 15 tasks with 2 to 5 modes each at any generation. This equates to approximately 400,000 experimental runs at each generation. This was achieved by breaking the Cartesian matrix into the Google maximum limit of 50 concurrent “Interactive” queries. More Google “Batch” queries can be deployed however will increase the execution run time. For more details on Google BigQuery limitations

¹ Source: <https://cloud.google.com/bigquery/quotas> (April 2018)

and details on Batch and Interactive queries, the reader is referenced to Google quotas and limits (Maharana et al. 2015).

Another limitation of the developed method is ignoring the higher-level interactions between the mode allocations in different activities. In example, the selection of mode A in activity 1 is not correlated to the selection of mode B in activity 2. This assumption was necessary to minimize the matrix size of the full factorial design of experiments runs. Such interactions will be further studied in future researches. However, the higher-level interactions between the modes within each activity is fully investigated which is more relevant to the developed method.

References

- Afshar, A., Ziaraty, A. K., Kaveh, A., & Sharifi, F. (2009). Nondominated archiving multicolony ant algorithm in time–cost trade-off optimization. *Journal of Construction Engineering and Management*, 135(7), 668-674.
- Aghaie, A., & Mokhtari, H. (2009). Ant colony optimization algorithm for stochastic project crashing problem in PERT networks using MC simulation. *The International Journal of Advanced Manufacturing Technology*, 45(11-12), 1051.
- Bettemir, Ö. H. (2009). “Optimization of time-cost-resource trade-off problems in project scheduling using meta-heuristic algorithms.” Ph.D. thesis, Middle East Technical Univ., Ankara, Turkey.
- Chiu, Y. S. P., & Chiu, S. W. (2005). Incorporating expedited time and cost of the end product into the product structure diagram. *International Journal of Machine Tools and Manufacture*, 45(7-8), 987-991.
- Comstock, D. A., Van Wychen, K., & Zimmerman, M. B. (2015). Why Don't They Just Give Us Money? Project Cost Estimating and Cost Reporting.

- Coonce, Thomas. "Joint Confidence Level – Yes We Are", NASA Headquarters Cost Analysis Division, 2009 NASA Cost Workshop, April 28th, 2009.
- De, P., Dunne, E. J., Ghosh, J. B., & Wells, C. E. (1995). The discrete time-cost tradeoff problem revisited. *European Journal of Operational Research*, 81(2), 225-238.
- Demeulemeester, E. L., Herroelen, W. S., & Elmaghraby, S. E. (1996). Optimal procedures for the discrete time/cost trade-off problem in project networks. *European Journal of Operational Research*, 88(1), 50-68.
- Elmaghraby, S. E. (1993). Resource allocation via dynamic programming in activity networks. *European Journal of Operational Research*, 64(2), 199-215.
- Eshtehardian, E., Afshar, A., & Abbasnia, R. (2009). Fuzzy-based MOGA approach to stochastic time-cost trade-off problem. *Automation in Construction*, 18(5), 692-701.
- Feng, C. W., Liu, L., & Burns, S. A. (1997). Using genetic algorithms to solve construction time-cost trade-off problems. *Journal of computing in civil engineering*, 11(3), 184-189.
- Gutjahr, W. J., Strauss, C., & Wagner, E. (2000). A stochastic branch-and-bound approach to activity crashing in project management. *INFORMS Journal on Computing*, 12(2), 125-135.
- Hauser, J. R. (2014). Consideration-set heuristics. *Journal of Business Research*, 67(8), 1688-1699.
- Hegazy, T. (1999). Optimization of construction time-cost trade-off analysis using genetic algorithms. *Canadian Journal of Civil Engineering*, 26(6), 685-697.
- Kalhor, E., Khanzadi, M., Eshtehardian, E., & Afshar, A. (2011). Stochastic time-cost optimization using non-dominated archiving ant colony approach. *Automation in Construction*, 20(8), 1193-1203.
- Ke, H. (2014). A genetic algorithm-based optimizing approach for project time-cost trade-off with uncertain measure. *Journal of Uncertainty Analysis and Applications*, 2(1), 8.

- Kelley Jr, J. E. (1961). Critical-path planning and scheduling: Mathematical basis. *Operations research*, 9(3), 296-320.
- Liu, L., Burns, S. A., & Feng, C. W. (1995). Construction time-cost trade-off analysis using LP/IP hybrid method. *Journal of construction engineering and management*, 121(4), 446-454.
- Moselhi, O. (1993). Schedule compression using the direct stiffness method. *Canadian Journal of Civil Engineering*, 20(1), 65-72.
- Moselhi, O. and Rofigari, N., (2013), "Schedule Compression: Multi-objective Methodology" *Journal of Construction Innovation*, Vol.13, No. 4, pp.374-393.
- Moselhi, O., & Alshibani, A. (2013). Schedule compression using fuzzy set theory and contractors judgment. *Journal of Information Technology in Construction (ITcon)*, 18(4), 64-75.
- NASA Cost Estimating Handbook. Version 4.0. Washington, D.C.: NASA. 2015
- Ng, S. T., & Zhang, Y. (2008). Optimizing construction time and cost using ant colony optimization approach. *Journal of construction engineering and management*, 134(9), 721-728.
- Patterson, J. H., & Huber, W. D. (1974). A horizon-varying, zero-one approach to project scheduling. *Management Science*, 20(6), 990-998.
- Pendharkar, P. C. (2015). Cost minimizing target setting heuristics for making inefficient decision-making units efficient. *International Journal of Production Economics*, 162, 1-12.
- Sakellariopoulos, S., & Chassiakos, A. P. (2004). Project time-cost analysis under generalised precedence relations. *Advances in Engineering Software*, 35(10-11), 715-724.
- Vanhoucke, M., & Debel, D. (2005). The discrete time/cost trade-off problem under various assumptions exact and heuristic procedures. *Universiteit Gent, FEEB*.
- Yang, I. T., Lin, Y. C., & Lee, H. Y. (2013). Use of support vector regression to improve computational efficiency of stochastic time-cost trade-off. *Journal of Construction Engineering and Management*, 140(1), 04013036.

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

Draft

List of figures

Figure 1: Developed ESDTCT method modules.

Figure 2: Definition and structure of terms

Figure 3: Immediate reverse dominator tree.

Figure 4: Typical JCL scatter plot and pareto frontier curve showing the different optimization objectives.

Figure 5: Nadir and Ideal vector points for the Pareto frontier.

Figure 6: Normalized Pareto frontier.

Figure 7: ESDTCT cloud network diagram.

Figure 8: Typical Main Effect Plot.

Figure 9: Schematic diagram of the proposed ESDTCT method.

Figure 10: Example 1: Network configuration.

Figure 11: Example 1: Network configuration showing primary activities at each generation in the ESDTCT method.

Figure 12: Example 1: ESDTCT supreme chromosome pareto frontier results verses NA-ACO method optimal chromosome reported by (Kalhor et al 2011).

Figure 13: Example 1: Main effect plot for response vectors.

Figure 14: Example 2: Network configuration showing primary activities at each generation in the ESDTCT method.

Figure 15: Example 2 Main effect plot for response vectors under uncertainty.

.

List of Appendices

Appendix A: Pseudocode for the ESDTCT^{Exp} Module

Appendix B: Pseudocode for the ESDTCT^{Rand} Module

Table 1 Details of example 1 (adapted from Eshtehardian et al. 2009)

	Mode Duration in days (Optimistic, Most Likely, Pessimistic) Direct cost \$ (Optimistic, Most Likely, Pessimistic)			Triangular Probability Distribution Function selected for both time and cost	
Activity	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5
1	(10,14,19) (2100,2400,2890)	(11,15,20) (1900,2150,2560)	(13,16,20) (1720,1900,2280)	(16,21,24) (1250,1500,2000)	(19,24,31) (985,1200,1750)
2	(12,15,21) (2870,3000,3420)	(16,18,23) (2185,2400,2850)	(18,20,25) (1650,1800,2255)	(20,23,28) (1300,1500,1950)	(19,25,30) (900,1000,1190)
3	(11,15,23) (4250,4500,4990)	(17,22,29) (3850,4000,4460)	(29,33,40) (2985,3200,3560)		
4	(10,12,16) (42050,45000,48800)	(13,16,21) (38500,35000,39000)	(17,20,28) (28500,30000,33550)		
5	(19,22,25) (18500,20000,22550)	(22,24,27) (16000,17500,19950)	(24,28,33) (14150,15000,17050)	(29,30,34) (8500,10000,12600)	
6	(12,14,17) (38500,40000,42860)	(17,18,21) (29800,32000,34550)	(21,24,29) (16550,18000,21000)		
7	(8,9,10) (28500,30000,33670)	(11,15,19) (21670,24000,28560)	(16,18,23) (20000,22000,23560)		
8	(11,14,16) (185,220,282)	(13,15,19) (182,215,255)	(13,16,21) (182,200,245)	(17,21,25) (175,208,234)	(21,24,29) (110,120,132)
9	(11,15,20) (290,300,313)	(16,18,23) (212,240,288)	(18,20,22) (165,180,225)	(20,23,28) (125,150,196)	(21,25,28) (85,100,124)
10	(13,15,19) (420,450,492)	(21,22,27) (385,400,485)	(30,33,39) (290,320,356)		
11	(10,12,16) (410,450,510)	(14,16,20) (313,350,395)	(16,20,26) (284,300,352)		
12	(18,22,29) (1850,2000,2450)	(19,24,30) (1565,1750,2050)	(20,28,36) (1325,1500,1880)	(21,30,45) (915,1000,1350)	
13	(12,14,18) (3650,4000,4540)	(16,18,20) (2970,3200,3385)	(23,24,26) (1595,1800,2160)		
14	(7,9,11) (2580,3000,3685)	(13,15,19) (2200,2400,2880)	(16,18,23) (2080,2200,2850)		
15	(10,12,15) (4385,4500,4850)	(13,16,18) (3200,3500,3750)			
16	(18,20,23) (2650,3000,3850)	(19,22,26) (1850,2000,2480)	(20,24,30) (1340,1750,2240)	(22,28,34) (1250,1500,1950)	(23,30,39) (860,1000,1320)
17	(12,14,18) (3750,4000,4670)	(15,18,22) (3000,3200,3530)	(22,24,29) (1650,1800,2140)		
18	(8,9,12) (2850,3000,3575)	(11,15,20) (2030,2400,2950)	(14,18,22) (1950,2200,2660)		
Overhead cost \$ (185, 200, 235)					

Table 2 ESDTCT Optimal chromosome results of example 1 at each generation (for a JCL = 50%)

Activity ID >>				1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Num of available modes				5	5	3	3	4	3	3	5	5	3	3	4	3	3	2	5	3	3
Generation	Num of Experimental runs	Total duration	Total cost	<< Chromosome >>																	
1	45	102	\$191,053	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	V	V	V
2	54	102	\$188,518	1	1	1	1	1	1	1	1	1	V	1	V	V	V	5	1	1	
3	540	103	\$165,896	1	1	V	V	1	1	V	V	1	1	3	V	3	2	1	5	1	1
4	180	115	\$138,555	1	1	3	3	V	V	3	4	V	V	V	1	3	2	1	5	1	1
5	25	118	\$134,795	V	V	3	3	4	V	3	4	1	1	3	1	3	2	1	5	1	1
Final result		118	\$134,795	2	5	3	3	4	3	3	4	1	1	3	1	3	2	1	5	1	1
$\Sigma = 844$ runs																					
				<div><div></div> = Activity locked at supreme mode</div> <div><div>V</div> = Activity allowed to vary for its modes</div> <div><div>1</div> = Activity locked at deterministic Most Likely value of mode 1</div>																	

Table 3 Example 1 ESDTCT optimal chromosome for different JCL and different optimization objectives.

Activity ID >>				1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Num of available modes >>				5	5	3	3	4	3	3	5	5	3	3	4	3	3	2	5	3	3
Optimization objective	Selected joint cost and schedule confidence level %	Total duration	Total cost	<< Chromosome >>																	
Schedule compression	20%	100	\$198,356	1	3	1	1	1	1	1	5	1	1	3	1	2	3	1	3	1	1
	50%	104	\$181,280	1	4	1	1	3	1	3	1	1	1	1	1	2	1	1	5	1	1
	80%	107	\$174,272	1	3	1	3	2	1	2	1	1	1	3	1	2	3	1	3	1	1
Cost compression	20%	125	\$132,310	2	5	3	3	4	3	3	5	2	1	3	2	3	3	2	5	1	1
	50%	128	\$133,786	3	5	3	3	4	3	3	2	1	1	3	1	3	3	2	5	2	1
	80%	129	\$135,690	2	5	3	3	4	3	3	3	1	1	3	1	3	3	2	5	2	1
Joint cost and schedule compression	20%	111	\$134,177	1	5	2	3	4	3	3	4	1	1	1	1	3	2	1	5	1	1
	50%	118	\$134,795	1	5	3	3	4	3	3	4	1	1	3	1	3	2	1	5	1	1
	80%	118	\$138,458	1	4	3	3	4	3	3	1	1	1	1	1	3	2	1	5	1	1

Table 4 Details of example 2 (activities 1 to 34)

Activity	Mode Duration in days (a,m,b) Direct cost \$(a,m,b)		Triangular Probability Distribution Function selected for both time and cost		
	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5
1	(13,14,17) (2813,3750,4200)	(9,12,15) (3825,4250,5525)	(9,10,13) (4860,5400,6372)	(6,9,11) (5625,6250,8625)	
2	(17,21,26) (10125,11250,12825)	(13,18,24) (13320,14800,17612)	(13,17,20) (14580,16200,18792)	(13,15,20) (17685,19650,26135)	
3	(20,24,30) (20205,22450,28736)	(20,22,30) (22410,24900,34860)	(17,19,24) (25155,27950,35217)	(15,17,23) (28485,31650,34815)	
4	(16,19,22) (16020,17800,22962)	(15,17,20) (17460,19400,21340)	(11,15,18) (19440,21600,27864)		
5	(24,28,33) (28062,31180,37728)	(22,26,36) (30780,34200,40356)	(20,23,31) (34425,38250,44370)	(17,21,26) (37260,41400,55062)	
6	(33,44,55) (48834,54260,61856)	(30,42,58) (52605,58450,66633)	(33,38,43) (56903,63225,85354)	(28,35,44) (61335,68150,86551)	
7	(33,39,43) (42840,47600,56644)	(30,36,46) (45675,50750,56840)	(29,33,36) (49320,54800,71240)	(27,30,40) (53775,59750,81260)	
8	(36,52,65) (55926,62140,83268)	(37,47,53) (62730,69700,92004)	(37,44,51) (65340,72600,101640)	(28,39,48) (73575,81750,94830)	
9	(45,63,83) (65475,72750,97485)	(45,59,69) (71505,79450,90573)	(48,55,63) (77625,86250,108675)	(41,51,62) (82350,91500,115290)	(40,49,68) (89550,99500,125370)
10	(51,57,80) (59850,66500,91105)	(43,53,59) (63225,70250,82193)	(38,50,59) (68220,75800,103088)	(35,46,56) (72675,80750,112243)	(36,41,54) (77805,86450,111521)
11	(55,63,85) (74790,83100,108030)	(44,59,66) (80505,89450,108235)	(39,55,63) (88020,97800,134964)	(39,50,58) (93825,104250,115718)	(34,45,50) (101160,112400,157360)
12	(53,68,80) (67950,75500,83050)	(45,62,73) (73800,82000,100040)	(45,58,68) (78750,87500,120750)	(39,53,66) (82620,91800,119340)	(42,49,64) (86895,96550,106205)
13	(32,40,47) (30825,34250,40758)	(30,37,46) (34650,38500,44275)	(26,33,44) (39555,43950,61530)	(27,31,36) (43875,48750,56550)	
14	(29,33,37) (47475,52750,59080)	(22,30,40) (52605,58450,76570)	(20,27,31) (57060,63400,84956)	(18,25,34) (59625,66250,88775)	
15	(35,47,59) (34326,38140,44242)	(28,40,48) (37350,41500,56440)	(31,35,44) (42885,47650,53368)	(26,32,44) (48690,54100,61674)	
16	(56,75,96) (85140,94600,116358)	(58,70,88) (91125,101250,123525)	(53,66,75) (101475,112750,129663)	(44,61,82) (112050,124500,160605)	(43,57,68) (119565,132850,152778)
17	(44,60,70) (70605,78450,108261)	(43,55,74) (76050,84500,100555)	(39,49,58) (82125,91250,122275)	(39,47,54) (85176,94640,104104)	
18	(73,81,109) (114435,127150,170381)	(58,73,86) (128925,143250,180495)	(47,66,81) (139140,154600,177790)	(53,61,82) (145710,161900,192661)	
19	(28,36,46) (74250,82500,95700)	(26,34,46) (85320,94800,104280)	(24,30,39) (91530,101700,126108)		
20	(29,41,50) (43515,48350,57537)	(30,37,50) (47925,53250,68693)	(31,34,45) (53505,59450,70746)	(24,32,37) (60120,66800,79492)	
21	(54,64,70) (76725,85250,108268)	(46,60,81) (83340,92600,108342)	(43,57,75) (98820,99800,130738)	(45,53,70) (96750,107500,123625)	(44,49,69) (102375,113750,126263)
22	(50,58,79) (66825,74250,95040)	(39,53,67) (71190,79100,99666)	(42,50,67) (78030,86700,120513)	(40,47,64) (82350,91500,111630)	(33,42,58) (87660,97400,130516)
23	(34,43,52) (59805,66450,79740)	(34,41,45) (62820,69800,88646)	(30,37,42) (68220,75800,105362)	(23,33,45) (73260,81400,100122)	(26,30,35) (79605,88450,118523)
24	(55,66,75) (65250,72500,80475)	(50,62,77) (70650,78500,91060)	(45,58,78) (75330,83700,97092)	(44,53,66) (80415,89350,122410)	(43,49,56) (86760,96400,124356)
25	(41,54,59) (59985,66650,85979)	(37,50,67) (63090,70100,81316)	(39,47,54) (67320,74800,86768)	(36,43,49) (71550,79500,90630)	(32,40,53) (78120,86800,104160)
26	(71,84,117) (84150,93500,105655)	(64,79,109) (92250,102500,112750)	(63,73,86) (100125,111250,145738)	(52,68,95) (107775,119750,167650)	(54,62,75) (115650,128500,165765)
27	(59,67,93) (70650,78500,91845)	(47,60,67) (77805,86450,118437)	(41,57,74) (80190,89100,116721)	(50,56,74) (82350,91500,113460)	(43,53,59) (85275,94750,131703)
28	(51,66,85) (76500,85000,119000)	(47,63,83) (80775,89750,123855)	(53,60,73) (83250,92500,125800)	(44,58,67) (87120,96800,127776)	(38,54,65) (90450,100500,111555)
29	(53,76,90) (83430,92700,127926)	(55,71,79) (88650,98500,137900)	(48,67,92) (94140,104600,116106)	(56,64,70) (98910,109900,121989)	(52,60,79) (104040,115600,147968)
30	(24,34,44) (24750,27500,30800)	(25,32,41) (26820,29800,39038)	(26,29,39) (28575,31750,43498)	(23,27,33) (30420,33800,41574)	(19,26,31) (32580,36200,42716)
31	(74,96,132) (130500,145000,189950)	(74,89,117) (139320,154800,185760)	(70,83,110) (151785,168650,219245)	(55,77,89) (161550,179500,233350)	(55,72,80) (170190,189100,219356)
32	(37,43,52) (38835,43150,56527)	(33,40,51) (43470,48300,61824)	(26,37,44) (46305,51450,57110)	(29,35,44) (49140,54600,69342)	(27,33,46) (55305,61450,78042)
33	(37,52,59) (55125,61250,71050)	(44,49,60) (57915,64350,74646)	(33,44,59) (61875,68750,83188)	(33,41,54) (67050,74500,96850)	(29,38,46) (71550,79500,88245)
34	(54,74,99) (80325,89250,124058)	(51,71,94) (84420,93800,105994)	(57,66,74) (89775,99750,123690)	(51,62,76) (94590,105100,147140)	(48,57,71) (102825,114250,155380)

Table 5 Details of example 2 (activities 35 to 63)

Activity	Mode Duration in days (a,m,b) Direct cost \$(a,m,b)		Triangular Probability Distribution Function selected for both time and cost		
	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5
35	(104,138,157) (164700,183000,203130)	(102,126,160) (181350,201500,227695)	(83,115,150) (214200,238000,271320)	(87,103,123) (255375,283750,380225)	(73,98,113) (267750,297500,330225)
36	(41,54,63) (42750,47500,58425)	(44,49,68) (45675,50750,70035)	(38,42,53) (51120,56800,76112)	(29,38,42) (56475,62750,84713)	(27,33,37) (61425,68250,94868)
37	(29,34,46) (20250,22500,28575)	(23,32,36) (21690,24100,27956)	(21,29,39) (24075,26750,33438)	(21,27,33) (26820,29800,33972)	(22,24,30) (28440,31600,43924)
38	(36,51,68) (55125,61250,85138)	(41,47,63) (59220,65800,86856)	(33,44,49) (64125,71250,88350)	(30,41,53) (68850,76500,104040)	(32,38,48) (72360,80400,105324)
39	(54,67,75) (73035,81150,108741)	(43,61,79) (78840,87600,106872)	(40,57,79) (82890,92100,113283)	(37,52,60) (87705,97450,131558)	(43,49,61) (92520,102800,117192)
40	(34,41,55) (40725,45250,55205)	(32,39,53) (43560,48400,58080)	(27,36,46) (46080,51200,57856)	(27,33,46) (49230,54700,65093)	(23,31,35) (52380,58200,79734)
41	(26,37,43) (15750,17500,23450)	(26,31,42) (19080,21200,26288)	(22,27,30) (24165,26850,34368)	(18,23,30) (29070,32300,40375)	
42	(37,44,61) (32760,36400,49504)	(36,41,54) (35775,39750,50880)	(32,38,46) (38520,42800,55212)	(29,32,39) (43470,48300,53130)	(24,30,41) (45225,50250,60300)
43	(54,75,103) (60120,66800,90180)	(62,69,84) (64080,71200,90424)	(52,63,75) (68760,76400,105432)	(48,59,72) (73170,81300,111381)	(41,54,64) (77580,86200,120680)
44	(65,82,93) (92475,102750,136658)	(68,76,104) (98550,109500,148920)	(56,70,79) (114300,127000,173990)	(57,66,89) (123120,136800,184680)	(55,63,81) (131400,146000,167900)
45	(53,59,70) (76275,84750,105938)	(45,55,73) (82260,91400,117906)	(38,51,67) (91170,101300,123586)	(42,47,65) (113850,126500,174570)	(34,43,60) (128475,142750,161308)
46	(59,66,92) (84825,94250,112158)	(55,63,77) (89550,99500,117410)	(51,59,71) (97425,108250,125570)	(46,55,77) (106650,118500,159975)	(43,50,65) (122400,136000,156400)
47	(44,54,70) (66150,73500,97020)	(39,51,64) (70650,78500,91060)	(37,47,52) (75240,83600,94468)	(33,44,50) (79830,88700,116197)	(32,41,50) (84060,93400,110212)
48	(30,41,56) (33075,36750,46305)	(33,39,48) (35820,39800,43780)	(31,37,51) (39420,43800,60882)	(27,34,41) (43650,48500,66930)	(25,31,38) (48555,53950,73372)
49	(133,173,202) (240750,267500,358450)	(129,159,202) (260730,289700,393992)	(125,147,168) (280800,312000,374400)	(120,138,163) (317250,352500,412425)	(90,121,146) (357975,397750,505143)
50	(90,101,118) (43020,47800,53058)	(63,74,86) (55170,61300,74786)	(44,63,77) (69120,76800,89856)	(37,49,57) (82350,91500,128100)	
51	(71,83,109) (76140,84600,94752)	(55,77,93) (84285,93650,113317)	(60,72,81) (88650,98500,136915)	(52,65,88) (94140,104600,128658)	(43,61,84) (101880,113200,155084)
52	(28,31,42) (20835,23150,26623)	(20,28,33) (24840,27600,37812)	(23,26,35) (26820,29800,35760)	(18,24,31) (29475,32750,38973)	(17,21,26) (31680,35200,48224)
53	(34,39,46) (28350,31500,34965)	(26,36,50) (30825,34250,44525)	(29,33,45) (34020,37800,51786)	(21,29,35) (37125,41250,45788)	(19,26,32) (40140,44600,57980)
54	(17,23,30) (14850,16500,22110)	(17,22,25) (16020,17800,23496)	(15,21,29) (17775,19750,24885)	(16,20,25) (19080,21200,28832)	(16,18,21) (21870,24300,27945)
55	(23,29,33) (21060,23400,26676)	(20,27,36) (22725,25250,32825)	(21,26,30) (24210,26900,33356)	(21,24,31) (26460,29400,37632)	(15,22,26) (29250,32500,42900)
56	(29,38,42) (37125,41250,53625)	(31,35,48) (40185,44650,60724)	(29,33,37) (43020,47800,64052)	(23,31,36) (46260,51400,65278)	(22,29,34) (49905,55450,69867)
57	(29,41,47) (34020,37800,51030)	(33,38,45) (37125,41250,55688)	(25,35,46) (41040,45600,50616)	(28,32,44) (44775,49750,69153)	(23,30,40) (48060,53400,58740)
58	(19,24,30) (11250,12500,16375)	(16,22,28) (12240,13600,17816)	(18,20,27) (13725,15250,16928)	(13,18,25) (15120,16800,18648)	(13,16,19) (17505,19450,24702)
59	(23,27,37) (31140,34600,43596)	(17,24,27) (33750,37500,49500)	(19,22,29) (37125,41250,53213)	(16,19,23) (42075,46750,52360)	(14,17,20) (45675,50750,69020)
60	(27,31,39) (25650,28500,31920)	(20,29,33) (27450,30500,36905)	(23,27,31) (29925,33250,41563)	(18,25,35) (34200,38000,42560)	(16,21,28) (39420,43800,56064)
61	(23,29,34) (20250,22500,31050)	(20,27,38) (22275,24750,29453)	(20,25,29) (24525,27250,34335)	(16,22,26) (26820,29800,33674)	(16,20,23) (30150,33500,40535)
62	(21,25,30) (34875,38750,44563)	(20,23,28) (37080,41200,45320)	(18,21,25) (40275,44750,52805)	(14,19,26) (44820,49800,67230)	(12,17,23) (45990,51100,70518)
63	(24,27,33) (8550,9500,12160)	(18,26,35) (8730,9700,12610)	(19,25,31) (9090,10100,13938)	(17,24,29) (9720,10800,14364)	(19,22,29) (11430,12700,17145)
Overhead cost \$ (\$2070, \$2300, \$2760)					

Table 6 Example 2 comparison of results from different meta-heuristic optimization methods.

Time and cost optimal solution using various meta-heuristic optimization methods (reported by Bettemir 2009).		Cost	Duration
GMSA	Genetic Memetic Algorithm Simulated Annealing.	\$ 5,421,120	630
HGAQSA	Hybrid Genetic Algorithm with Quantum Simulated Annealing.	\$ 5,421,120	630
GASAVNS	Genetic algorithm with Simulated Annealing and Variable Neighborhood Search.	\$ 5,421,120	630
ACO	Any Colony Optimization	\$ 5,492,210	635
EMS	Electromagnetic Scatter Search.	\$ 5,532,920	622
PSO	Particle Swarm Optimization.	\$ 5,421,320	633
GASA	Genetic Algorithm Simulated Annealing	\$ 5,421,620	633
GA	Genetic Algorithm	\$ 5,690,790	623
ESDTCT	developed method	\$ 5,421,320	633
ESDTCT	Supreme Chromosome : [1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,2,1,1,2,1,1,1,1,1,1,1,1,1,1,2,1,1,1,4,1,1,1,1,1,1,1,3, 1,1,1,1,5,1,1,5,1,1,1,5,1,5] for activities 1 to 63		

Table 7 Example 6 ESDTCT^{Exp} supreme chromosome results at each generation under uncertain environment.

[illegible]

Canadian Journal of Civil Engineering

ESDTCT Method

Project Network

IRDT (Generation 1)

IRDT (Generation 2)

<https://mc06.manuscriptcentral.com/cjce-pubs>

Highlighted are primary activities in a generation

<https://mc06.manuscriptcentral.com/cjce-pubs>
Shaded area are the selection of primary activities at each generation

Activity ID >>>>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
ESDTCT Supreme Chromosome	1	5	3	3	4	3	3	4	1	1	3	1	3	2	1	5	1	1
NA-ACO Optimal Chromosome *	3	5	3	3	4	3	3	5	1	1	3	1	3	3	2	5	3	1

Differences <https://mc06.manuscriptcentral.com/cjce-pubs>

* Obtained by Kalhor et al. (2011)

Main Effect Plot for solution response vector

Data Means

Marks the modes of the supreme chromosome resulting from ESDTCT^{Exp} module

Shaded area are the selection of primary activities at each generation

Appendix A. Pseudocode for the ESDTCT^{Exp} Module

```

PROGRAM ESDTCT_Exp:
Gen = 0 // Initialize generation
WHILE not all project activities are "Supreme mode activities" DO
  IF Gen = 0 then
    Loop for network logic array
      Tag "primary activities" as those selected from the Immediate
      reverse dominator tree for the completion milestone.
      Tag the balance of network activities as "Nuisance activities"
    End Loop
  ELSE IF
    Gen += 1
    Loop for network logic array
      Tag "primary activities" in (Gen-1 to Gen 0) as "Supreme mode
      activities"
      Tag "primary activities" as those selected from the Immediate
      reverse dominator tree for all primary activities in (Gen-1)
      Tag the balance of network activities as "Nuisance activities"
    End Loop
  END IF
  Build a generation of Chromosomes (design of experiments DOE matrix) using a
  multi level Cartesian product matrix with the following Blocking technique:
  > Vary "primary activities" for all their modes,
  > lock "Supreme mode activities" to the identified supreme modes and
  > lock "Nuisance activities" at their most likely duration and cost of
  their respective mode 1.
  // (Note: the resulting matrix is a design of experiment where each run
  represents a chromosome of modes with a focus on analyzing the effect of
  primary activities.)

  Partition the DOE matrix into groups of 5000 chromosomes. // (Note: 5000 is
  selected to reduce the data flow to below 5 Mb at each call for the JavaScript
  function. This is made to respect the free-quota limit applied by BigQuery.)

  Initiate Google BigTable "Output" to collect the output of each chromosome

  Run concurrent Google BigQuery calls for each partition.

  Run Monte-Carlo analysis using the JCL simulation technique for full
  enumeration of the partition chromosomes as follows:

  FOR experiment run 1 to 5000 //enumeration of the Chromosomes at each
  partition
    FOR Simulation runs = 1 to 100 //(note 100 is a user selected
    number)
      Select random values for the activities durations and cost
      values according to the assigned probability distribution
      function (PDF)
      Where P is the random number [0,1] , a is the optimistic
      value , m is the most likely value and, b is the
      pessimistic value.
      CASE Triangular PDF // For Triangular probability
      distribution function
      TriDistInv(P , a , m , b ) {
        var d ,      var x
        d = b - a
        if (d != 0){x = (m - a) / d} else {x = b;}
        if (P <= x){return a + ((Math.sqrt(P * x)) *
        d)} else {
          return b - ((Math.sqrt((1 - P) * (1 - x))) *
          d);} }
    }
  }

```

```

CASE Beta-PERT PDF // For Beta-PERT probability
                    distribution function
BetaPERTDistInv(P , a , m , b ) {
// Note : The Beta-PERT distribution is a special
case of the Beta Distribution function "BETA"
available in Java Script and specified by the
parameters alpha, beta Where:
    var alpha = (4*m + b - 5*a)/(b - a)
    var beta = (5*b - a - 4*m)/(b - a)
    return Math.BETA.INV(P, alpha, beta,
                        a, b); }

CALL CPM function to solve for the network activities start
and finish dates using the forward pass of the CPM
technique. Then the Expected value E() is:

E(Total duration) = E(completion date for the Finish
                    milestone) - project start date + 1 day
E(Total overhead cost) = E(overhead cost rate) x
                        E(total project duration).
E(Total indirect cost) = SUM for activity i = 1 to N(
                        E(indirect cost rate(i)) x
                        E(duration(i))
E(Total project cost) = SUM for activity i = 1 to N (
                        E(direct cost(i)), E(indirect
                        cost(i))
                        + E(overhead cost values)

NEXT Simulation run

Identify the pareto frontier for the selected JCL % (all simulation run
solutions that fall on the specified JCL% ± Tolerance factor

Append the obtained pareto frontier solutions characteristics resulting
from the current experiment run to the BigTable "Output", including the
chromosome encoding of the modes, the total cost and duration of the
solution.

NEXT experiment run

Identify the Nadar and Ideal points amongst the solutions
collected in the BigTable "Output" and normalize the cost and
schedule values of each solution to the range [0,1]

Analyze the "Output" table to Identify the single optimal solution
based on the optimization objective as follows:
CASE: optimization is set for Cost Reduction: THEN
    optimal solution is the one having the lowest total
    cost value
CASE: optimization is set for Schedule Compression: THEN
    optimal solution is the one having the lowest total
    duration value
CASE: optimization is set for Joint Cost and Schedule
Compression: THEN
    optimal solution is the one having the minimum
    distance to the Nadar point (Response vector).
Use the chromosome mode encoding as input for the next generation
END WHILE //Next generation

The final chromosome resulting from all generations is the Supreme chromosome that
provides the most optimum solution to the Trade-Off analysis.

END.

```

Appendix B. Pseudocode for the ESDTCT^{Rand} Module

PROGRAM ESDTCT_Rand:

Initiate Google BigTable "Random_Output" to collect the output of each chromosome

WHILE User defined termination condition is not satisfied DO

Run concurrent Google BigQuery calls, each for 50000 random chromosomes.

// (Note: 50,000 is selected to reduce the data flow to below 5 Mb at each call for the JavaScript function. This is made to respect the free-quota limit applied by BigQuery.)

Run Monte-Carlo analysis using the JCL simulation technique for each of the chromosomes as follows:

FOR experiment run 1 to 50000 //enumeration of the Chromosomes at each partition

FOR Simulation runs = 1 to 100 //(note 100 is a user selected number)

Select random values for the activities durations and cost values according to the assigned probability distribution function (PDF) Where P is the random number [0,1] , a is the optimistic value , m is the most likely value and, b is the pessimistic value.

CASE Triangular PDF // For Triangular probability distribution function

```
TriDistInv(P , a , m , b ) {
    var d ,      var x
    d = b - a
    if (d != 0){x = (m - a) / d} else {x = b;}
    if (P <= x){return a + ((Math.sqrt(P * x)) * d)}
else {
    return b - ((Math.sqrt((1 - P) * (1 - x))) * d);} }
```

CASE Beta-PERT PDF // For Beta-PERT probability distribution function

```
BetaPERTDistInv(P , a , m , b ) {
// Note : The Beta-PERT distribution is a special case of
the Beta Distribution function "BETA" available in Java
Script and specified by the parameters alpha, beta Where:
    var alpha = (4*m + b - 5*a)/(b - a)
    var beta = (5*b - a - 4*m)/(b - a)
    return Math.BETA.INV(P, alpha, beta,
        a, b); }
```

CALL CPM function to solve for the network activities start and finish dates using the forward pass of the CPM technique. Then the Expected value E() is:

$E(\text{Total duration}) = E(\text{completion date for the Finish milestone}) - \text{project start date} + 1 \text{ day}$

$E(\text{Total overhead cost}) = E(\text{overhead cost rate}) \times E(\text{total project duration}).$

$E(\text{Total indirect cost}) = \text{SUM for activity } i = 1 \text{ to } N(E(\text{indirect cost rate}(i)) \times E(\text{duration}(i)))$

$E(\text{Total project cost}) = \text{SUM for activity } i = 1 \text{ to } N (E(\text{direct cost}(i)), E(\text{indirect cost}(i)) + E(\text{overhead cost values}))$

Identify the pareto frontier for the selected JCL % (all simulation run solutions that fall on the specified JCL% $\pm$ Tolerance factor

Identify the Nadar and Ideal points amongst the solutions resulting from the simulation run that fall on the selected pareto frontier curve and normalize the cost and schedule values of each solution to the range [0,1]

Analyze the selection of solutions to Identify the single optimal solution based on the optimization objective as follows:

```

CASE: optimization is set for Cost Reduction: THEN
    optimal solution is the one having the lowest total
    cost value
CASE: optimization is set for Schedule Compression: THEN
    optimal solution is the one having the lowest total
    duration value
CASE: optimization is set for Joint Cost and Schedule
    Compression: THEN
    optimal solution is the one having the minimum
    distance to the Nadar point (Response vector). // use
    equation 1 to 3.

```

Append the obtained optimal solution characteristics resulting from the current simulation run to the BigTable "Random_Output", including the chromosome encoding of the modes, the total cost and duration of the solution.

```

        NEXT Simulation run
    NEXT experiment run
END WHILE //Next concurrent BigQuery

// Generate the main effect plot
//loop over the BigTable "Random-Output" to obtain the average of the Response for
    each mode for each activity.
FOR Activity i = 1 to K // K = total number of activities in the project network.
    FOR Mode j = 1 to M // M = total number of modes assigned to the activity.
        CASE: optimization is set for Cost Reduction: THEN
            Response (i,j) = Average (Total cost (i,j))
        CASE: optimization is set for Schedule Compression: THEN
            Response (i,j) = Average (Total Duration (i,j))
        CASE: optimization is set for Joint Cost and Schedule
            Compression: THEN
            Response (i,j) = Average (Response vector (i,j))
    NEXT Mode
Next Activity
Plot the Main effect charts
END.

```