

A DISCRETE-TIME APPROXIMATION TECHNIQUE FOR THE TIME-COST TRADE-OFF IN PERT NETWORKS

AMIR AZARON¹, MASATOSHI SAKAWA², REZA
TAVAKKOLI-MOGHADDAM³ AND NIMA SAFAEI⁴

Abstract. We develop a discrete-time approximation technique dealing with the time-cost trade-off problem in PERT networks. It is assumed that the activity durations are independent random variables with generalized Erlang distributions, in which the mean duration of each activity is a non-increasing function of the amount of resource allocated to it. It is also assumed that the amount of resource allocated to each activity is controllable. Then, we construct an optimal control problem with three conflicting objective functions. Solving this optimal control problem, optimally, is impossible. Therefore, a discrete-time approximation technique is applied to solve the original multi-objective optimal control problem, using goal attainment method. To show the advantages of the proposed technique, we also develop a Simulated Annealing (SA) algorithm to solve the problem, and compare the discrete-time approximation results against the SA and also the genetic algorithm results.

Keywords. Project management, multiple objective programming, optimal control, markov processes, simulated annealing.

Mathematics Subject Classification. 90C29, 90C35, 90C59.

Received December 25, 2004. Accepted September 8, 2006.

¹ Department of Industrial Engineering, Dalhousie University, Halifax, Nova Scotia, B3J 2X4 Canada; a.azaron@dal.ca

² Department of Artificial Complex Systems Engineering, Graduate School of Engineering, Hiroshima University, Kagamiyama 1-4-1, Higashi-Hiroshima, Hiroshima, 739-8527 Japan; sakawa@msl.sys.hiroshima-u.ac.jp

³ Department of Industrial Engineering, Faculty of Engineering, University of Tehran, Tehran, Iran; tavakoli@ut.ac.ir

⁴ Department of Industrial Engineering, Iran University of Science and Technology, Tehran, Iran; nimasafaei@iust.ac.ir

© EDP Sciences, ROADEF, SMAI 2007

INTRODUCTION

The most important method to schedule a project assuming deterministic durations is the well-known CPM – Critical Path Method. However, most durations have random natures and therefore, PERT was proposed to determine the distribution of the total duration, T . This method is based on the substitution of the network by the CPAD – critical path assuming that each activity has a fixed duration equal to its mean (critical path using average durations). The mean and the variance of the CPAD are given by the sum of the means and of the variances of its activities, respectively, and therefore these results considered the mean and the variance of the total duration of the network.

Unfortunately, this is an optimistic assumption as the real mean, $E(T)$, is greater than or equal to such estimate. Thus, many authors have studied:

1. Analytical approximations of the cumulative distribution function of T , $F(T)$. Charnes *et al.* [5] developed a chance-constrained programming approach to PERT problems. They assume exponential activity durations. Martin [27] provided a systematic way of analyzing PERT networks through series-parallel reductions. Fatemi Ghomi and Hashemin [12] generalized the Gaussian quadrature formula to compute $F(T)$. Fatemi Ghomi and Rabbani [13] presented a structural mechanism, which changes the structure of network to a series-parallel network, to estimate $F(T)$. Schmit and Grossmann [35] developed a new technique for computing the exact overall duration of a project, when activity durations use a probability density function which combines piecewise polynomial segments and Dirac delta functions, defined over a finite interval. Pontrandolfo [30] provided an approximate estimate of the project duration by deriving the equations that put into relation the duration of the project and those of every possible PERT-path. Kulkarni and Adlakha [24] developed a continuous-time Markov process approach to PERT problems.

2. Upper or lower bounds of $F(T)$. Elmaghraby [9] provided lower bounds for the true, expected project completion time. Fulkerson [17], Clingen [7], Robillard [31] and Perry and Creig [29] have done the similar works.

3. Monte-Carlo simulation to estimate $F(t)$. Several authors have used conditional sampling to achieve variance reduction (see Burt and Garman [4], Garman [18] and Sigal *et al.* [34]). Fishman [15] achieved further variance reduction by using a combination of quasirandom points and conditional sampling to estimate the distribution and mean of project completion time.

In CPM networks, activity duration is viewed either as a function of cost or a function of resources committed to it. The well-known time-cost trade-off problem (TCTP) in CPM network takes the former view. In the TCTP, the objective is to determine the duration of each activity in order to achieve the minimum total direct and indirect costs of the project. Studies on TCTP have been done using various kinds of cost functions such as linear (Fulkerson [16] and Kelly [21]), discrete (Demeulemeester *et al.* [8]), convex (Lamberson and Hocking [25] and Berman [3]) and concave (Falk and Horowitz [11]).

When the cost functions are arbitrary (still non-increasing), a dynamic programming (DP) approach was suggested by Robinson [32]. Another powerful approach for solving this problem on dynamic programming was presented by Elmaghraby [10]. Tavares [36] has presented a general model based on the decomposition of the project into a sequence of stages, in that the optimal solution can be easily computed for each practical problem as it is shown for a real case study. Golenko-Ginzburg and Gonik [19] developed a heuristic method for the stochastic extension of the CPM TCTP model, which takes into account the chance element of time, yet ignores the possible difference between the actual cost and the pre-given budget. Laslo [26] developed some formulations of TCTP models that represent different assumptions of the effect of the changing performance speed on the frequency distribution parameters of the activity duration, as well as the effect of the random activity duration on the activity cost.

Weglarz [37] studied this problem using optimal control theory and assuming that the processing speed of each activity at time t is a continuous, non-decreasing function of the amount of resource allocated to the activity at that instant of time. This means that also time is here considered as a continuous variable. Unfortunately, it seems that this approach is not applicable to networks with a reasonable size (>10).

Recently, some researchers have adopted computational optimization techniques such as genetic algorithms and simulated annealing to solve TCTP. Feng *et al.* [14], Chau *et al.* [6] and Azaron *et al.* [1] proposed models using genetic algorithms and the Pareto front approach to solve construction time-cost trade-off problems.

The mentioned TCTP models mainly focus on deterministic situations. PERT does not take into account the time-cost trade-off, analytically. Therefore, combining the aforementioned concepts to develop an optimal or near-optimal method dealing with the time-cost trade-off problem under uncertainty would be beneficial to scheduling engineers in forecasting a more realistic project completion time and cost.

In this paper, we initially extend the method developed by Kulkarni and Adlakha [24] to analytically obtain the distribution function of project completion time in Markov PERT networks with generalized Erlang distributions of activity durations. This is done through solving a system of linear differential equations, which is obtained from a relevant continuous-time Markov process. Then, we construct an optimal control problem with three conflicting objective functions and develop a discrete-time approximation technique to solve this problem. Finally, we develop a simulated annealing algorithm and compare the discrete-time approximation results against the simulated annealing and also the genetic algorithm results, proposed by Azaron *et al.* [1].

It is assumed that the amount of resource allocated to each activity is controllable, where the mean duration of each activity is a non-increasing function of this control variable. The direct cost of each activity is also assumed to be a non-decreasing function of the amount of resource allocated to it.

The problem is formulated as a multi-objective optimal control problem, where the objective functions are the project direct cost (to be minimized), the mean of

the project completion time (min) and its variance (min). Then, goal attainment technique is used to solve this multi-objective problem and to obtain the optimal resources allocated to the activities.

Solving this optimal control problem, optimally, is impossible. Therefore, we either need an approximation technique or a heuristic algorithm to solve the problem. First, we do the discretization of time and convert the optimal control problem into an equivalent nonlinear optimization problem. Then, we develop a simulated annealing algorithm to solve the problem. Finally, we compare the discrete-time approximation results against the simulated annealing and also the genetic algorithm results, which are two powerful heuristic algorithms.

The remainder of this paper is organized in the following way. In Section 1, an analytical method for obtaining the distribution function of project completion time in Markov PERT networks is presented. In Section 2, we present the multi-objective optimal control problem. In Section 3, we explain about the simulated annealing algorithm to solve the problem. In Section 4, we solve some numerical examples and compare the discrete-time approximation results against the simulated annealing and the genetic algorithm results. Finally we draw some conclusions of the paper in Section 5.

1. PROJECT COMPLETION TIME IN MARKOV PERT NETWORKS

A project is represented as an Activity-on-Arc (AoA) graph, where an activity begins as soon as all its predecessor activities have finished. In this section, we present an analytical method to obtain the distribution function of project completion time in Markov PERT networks, or in fact the distribution function of the longest path length from the source to the sink node of a directed acyclic stochastic network, in which the arc lengths or activity durations are mutually independent random variables with generalized Erlang distributions. To do this, we extend the technique of Kulkarni and Adlakha [24].

Approximating general distributions by phase-type (*PH*) distributions has significant applications in the analysis of stochastic systems. A popular approach to analyzing stochastic networks involving a general distribution of service time G is to approximate G by a *PH* distribution. A *PH* distribution is a very general mixture of exponential distributions. It has been shown that matching three moments is sufficient for accurate modeling of many systems. Most existing algorithms for fitting a general distribution G to a *PH* distribution, restrict their attention to a subset of *PH* distributions. This chosen subset is the class of generalized Erlang distributions, refer to Neuts [28] for more details.

Now, let $G = (V, A)$ be a PERT network, where the source and sink nodes are denoted by s and t , respectively. Let $V = \{v_1, v_2, \dots, v_m\}$ represent the set of nodes, and $A = \{a_1, a_2, \dots, a_n\}$ represent the set of activities of the PERT network. Duration of activity $a \in A$ (Ta) exhibits a generalized Erlang distribution

of order n_a and the infinitesimal generator matrix G_a as:

$$G_a = \begin{pmatrix} -\lambda_{a1} & \lambda_{a1} & 0 & \cdots & 0 & 0 \\ 0 & -\lambda_{a2} & \lambda_{a2} & \cdots & 0 & 0 \\ \vdots & & & & & \vdots \\ 0 & 0 & 0 & \cdots & -\lambda_{an_a} & \lambda_{an_a} \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix}.$$

In this case, T_a would be the time until absorption in the absorbing state. An Erlang distribution of order n_a is a generalized Erlang distribution with $\lambda_{a1} = \lambda_{a2} = \dots = \lambda_{an_a}$. When $n_a = 1$, the underlying distribution becomes exponential with the parameter λ_{a1} .

First, we transform the original PERT network into a new one, in which all activity durations have exponential distributions. For constructing this network, we use the idea that if the duration of activity a is distributed according to a generalized Erlang distribution of order n_a and the infinitesimal generator matrix G_a , it can be decomposed to n_a exponential serial arcs with the parameters $\lambda_{a1}, \lambda_{a2}, \dots, \lambda_{an_a}$. Then, we substitute each generalized Erlang activity with n_a serial exponential activities with the parameters $\lambda_{a1}, \lambda_{a2}, \dots, \lambda_{an_a}$.

Now, Let $G' = (V', A')$ be the transformed network, in which V' and A' represent the sets of its nodes and activities, respectively. The duration of each activity $a \in A'$ in the transformed network is exponential with parameter λ_a . For $a \in A'$, let $\alpha(a)$ be the starting node of arc a , and $\beta(a)$ be the ending node of arc a .

Definition 1. Let $I(v)$ and $O(v)$ be the sets of arcs ending and starting at node v , respectively, which are defined as follows:

$$I(v) = \{a \in A' : \beta(a) = v\} \quad (v \in V') \quad (1)$$

$$O(v) = \{a \in A' : \alpha(a) = v\} \quad (v \in V'). \quad (2)$$

Definition 2. If $X \subset V'$, such that $s \in X$ and $t \in \bar{X} = V' - X$, then an (s, t) cut is defined as:

$$(X, \bar{X}) = \{a \in A' : \alpha(a) \in X, \beta(a) \in \bar{X}\}. \quad (3)$$

An (s, t) cut (X, \bar{X}) is called an uniformly directed cut (UDC), if (\bar{X}, X) is empty.

Definition 3. Let $D = E \cup F$ be a uniformly directed cut (UDC) of a network. Then, it is called an admissible 2-partition, if $E \cap F = \phi$ and $I(\beta(a)) \not\subset F$ for any $a \in F$.

Definition 4. We assume that the project modeled by network G starts at time zero and ends at a random time T . During the project execution and at time t , each activity can be in one of the active, dormant or idle states, which are defined as follows:

Active: an activity is active at time t , if it is being executed at time t .

Dormant: an activity is dormant at time t , if it has finished but there is at least one unfinished activity in $I(\beta(a))$. If an activity is dormant at time t , then its successor activities in $O(\beta(a))$ cannot begin.

Idle: an activity is idle at time t , if it is neither active nor dormant at time t .

The sets of active and dormant activities are denoted by $Y(t)$ and $Z(t)$, respectively, and $X(t) = (Y(t), Z(t))$.

Let S denote the set of all admissible 2-partition cuts of the network, and $\bar{S} = S \cup \{(\phi, \phi)\}$. Note that $X(t) = (\phi, \phi)$ implies that $Y(t) = \phi$ and $Z(t) = \phi$, *i.e.* all activities are idle at time t and hence the project is completed by time t .

It is proven that $\{X(t), t \geq 0\}$ is a continuous-time Markov process with the state space \bar{S} (refer to Kulkarni and Adlakha [24] for details). The infinitesimal generator matrix of this process is denoted by $Q = [q\{(E, F), (E', F')\}]$, (E, F) and $(E', F') \in \bar{S}$ where

$$q\{(E, F), (E', F')\} = \begin{cases} \lambda_a & \text{if } a \in E, I(\beta(a)) \not\subset F \cup \{a\}, \\ & E' = E - \{a\}, F' = F \cup \{a\}; \end{cases} \quad (4)$$

$$\begin{cases} \lambda_a & \text{if } a \in E, I(\beta(a)) \subset F \cup \{a\}, \\ & E' = (E - \{a\}) \cup O(\beta(a)), \\ & F' = F - I(\beta(a)); \end{cases} \quad (5)$$

$$\begin{cases} \sum_{a \in E} \lambda_a & \text{if } E' = E, F' = F; \end{cases} \quad (6)$$

$$\begin{cases} 0 & \text{otherwise.} \end{cases} \quad (7)$$

$\{X(t), t \geq 0\}$ is a finite-state absorbing continuous-time Markov process. Since $q\{(\phi, \phi), (\phi, \phi)\} = 0$ this state would be an absorbing one and obviously the other states are transient. Furthermore, the states in \bar{S} are numbered such that the Q matrix is an upper triangular matrix. We assume that the states are numbered $1, 2, \dots, N = |\bar{S}|$. State 1 is the initial state, namely $(O(S), \phi)$ and state N is the absorbing state, namely (ϕ, ϕ) .

Let T represent the length of the longest path in the network, or the project completion time. Clearly, $T = \min\{t > 0 : X(t) = N/X(0) = 1\}$. Thus, T is the time until $\{X(t), t \geq 0\}$ gets absorbed in the final state starting from state 1 .

Kolmogorov backward or forward equations can be applied to compute $F(t) = P\{T \leq t\}$ or the distribution function of project completion time in the Markov PERT network. Using the backward algorithm, we define:

$$P_i(t) = P\{X(t) = N/X(0) = i\} \quad i = 1, 2, \dots, N \quad (8)$$

Therefore, $F(t) = P_1(t)$.

The system of differential equations for the vector $P(t) = [P_1(t), P_2(t), \dots, P_N(t)]^T$ is given by

$$\begin{aligned} P'(t) &= QP(t) \\ P(0) &= [0, 0, \dots, 1]^T. \end{aligned} \quad (9)$$

2. MULTI-OBJECTIVE OPTIMAL CONTROL PROBLEM

In this section, we develop a multi-objective model to optimally control the resources allocated to the activities in a PERT network whose activity durations exhibit generalized Erlang distributions, where the mean duration of each activity is a non-increasing function and the direct cost of each activity is a non-decreasing function of the amount of resource allocated to it. We may decrease the project direct cost by decreasing the amount of resources allocated to the activities. However, clearly it causes the mean project completion time to be increased, because these objectives are in conflict with each other. Consequently, an appropriate trade-off between the total direct costs and the mean project completion time is required. The variance of the project completion time should also be considered in the model, because when we only focus on the mean time, the resource quantities may be non-optimal if the project completion time substantially varies because of randomness.

Therefore, we face a multi-objective stochastic programming problem. The objective functions are the project direct cost (to be minimized), the mean of project completion time (min) and the variance of project completion time (min).

The direct cost of activity $a \in A$ is assumed to be a non-decreasing function $d_a(x_a)$ of the amount of resource x_a allocated to it. Therefore, the project direct cost (PDC) would be equal to $PDC = \sum_{a \in A} d_a(x_a)$.

The mean duration of activity $a \in A$, which is equal to $\sum_{j=1}^{n_a} \frac{1}{\lambda_{aj}}$, is assumed to be a non-increasing function $g_a(x_a)$ of the amount of resource x_a allocated to it. Let U_a represent the amount of resource available to be allocated to the activity a , and L_a represent the minimum amount of resource required to achieve the activity a .

The mean and the variance of project completion time are given by

$$E(T) = \int_0^\infty t P_1'(t) dt \quad (10)$$

$$Var(T) = \int_0^\infty t^2 P_1'(t) dt - \left[\int_0^\infty t P_1'(t) dt \right]^2 \quad (11)$$

where $P_1'(t)$ is the density function of project completion time.

The infinitesimal generator matrix Q would be a function of the vector $\lambda = [\lambda_{aj}, a \in A, j = 1, 2, \dots, n_a]^T$ in the optimal control problem, where n_a is the order

of the generalized Erlang distribution. Therefore, the nonlinear dynamic model is:

$$\begin{aligned} P'(t) &= Q(\lambda)P(t) \\ P_i(0) &= 0 \quad i = 1, 2, \dots, N-1. \\ P_N(t) &= 1 \end{aligned} \quad (12)$$

Accordingly, the appropriate multi-objective optimal control problem is:

$$\begin{aligned} \text{Min } f_1(x, \lambda) &= \sum_{a \in A} d_a x_a \\ \text{Min } f_2(x, \lambda) &= \int_0^\infty t P'_i(t) dt \\ \text{Min } f_3(x, \lambda) &= \int_0^\infty t^2 P'_1(t) dt - [\int_0^\infty t P'_1(t) dt]^2 \end{aligned}$$

s.t.

$$\begin{aligned} P'(t) &= Q(\lambda)P(t) \\ P_i(0) &= 0 \quad i = 1, 2, \dots, N-1 \\ P_N(t) &= 1 \\ g_a(x_a) &= \sum_{j=1}^{n_a} \frac{1}{\lambda_{aj}} \quad a \in A \\ x_a &\leq U_a \quad a \in A \\ x_a &\geq L_a \quad a \in A \\ \lambda_{aj} &\geq 0 \quad a \in A, j = 1, 2, \dots, n_a. \end{aligned} \quad (13)$$

A possible approach to solving (13) to optimality is to use the maximum principle, see Sethi and Thompson [33] for details. It is proved that finding the optimal solution of the above optimal control problem is impossible (refer to Azaron *et al.* [1] for details). Relatively, few optimal control problems can be solved optimally. So, we need an approximation or a heuristic technique to solve this problem.

First, we use an approximate technique. In this technique, we do the discretization of time and convert the optimal control problem (13) into an equivalent nonlinear programming problem. In other words, we transform the differential equations into the equivalent difference equations as well as transform the integral terms into equivalent summation terms. To follow this approach, the time interval is divided into K equal portions with the length of Δt . If Δt is sufficiently small, it can be assumed that $P(t)$ varies only in times $0, \Delta t, \dots, (K-1)\Delta t$. Since each $P_i(k)$, for $i = 1, 2, \dots, N-1, k = 1, 2, \dots, K$, is a distribution function, then the following constraints should also be considered in the final discrete-time problem.

$$P_i(K) \leq 1 \quad i = 1, 2, \dots, N-1, k = 1, 2, \dots, K. \quad (14)$$

Theoretically, when K approaches to infinity and Δt approaches to zero, the optimal results of the original problem will be obtained, but in this case the computational time also approaches to infinity, which is not practical in reality. Practically, we should select a finite value for K . The accuracy of the discrete-time approximation model is guaranteed by using a small value for Δt and a large value for K . An accurate solution should also possess the following property: $P_1(K) \geq 1 - \epsilon$, in which ϵ is a positive value approaching zero.

We also develop a simulated annealing approach to solve the problem. Finally, we compare the discrete-time approximation technique against the simulated annealing technique and the genetic algorithm to show the advantages of the discrete-time approximation technique.

2.1. GOAL ATTAINMENT METHOD

The goal attainment method requires setting up a goal and weight, b_j and c_j ($c_i \geq 0$), $j = 1, 2, 3$, for the project direct cost, the mean of the project completion time and its variance, respectively. The c_j relate the relative under-attainment of the b_j . For under-attainment of the goals, a smaller c_j is associated with the more important objectives. c_j , $j = 1, 2, 3$, are generally normalized so that $\sum_{i=1}^3 c_i = 1$.

Considering $P(k\Delta t)$ or the k th value of P as $P(k)$, the appropriate goal attainment formulation of the discrete-time approximation of the time-cost trade-off problem leads to:

Min z

s.t.

$$\sum_{a \in A} d_a(x_a) - c_1 z \leq b_1$$

$$\sum_{k=0}^{K-1} k\Delta t(P_1(k+1) - P_1(k)) - c_2 z \leq b_2$$

$$\sum_{k=0}^{K-1} (k\Delta t)^2(P_1(k+1) - P_1(k)) - [\sum_{k=0}^{K-1} k\Delta t(P_1(k+1) - P_1(k))]^2 - c_3 z \leq b_3$$

$$P(k+1) = P(k) + Q(\lambda)P(k)\Delta t \quad k = 0, 1, \dots, K-1$$

$$P_i(0) = 0 \quad i = 1, 2, \dots, N-1$$

$$P_N(k) = 1 \quad k = 0, 1, \dots, K$$

$$P_i(k) \leq 1 \quad i = 1, 2, \dots, N-1, k = 1, 2, \dots, K$$

$$g_a(x_a) = \sum_{j=1}^{n_a} \frac{1}{\lambda_{aj}} \quad a \in A$$

$$\begin{aligned}
x_a &\leq U_a & a &\in A \\
x_a &\geq L_a & a &\in A \\
\lambda, z &\geq 0. & & (15)
\end{aligned}$$

Lemma 1. If x^* is Pareto-optimal, then there exists a c, b pair such that x^* is an optimal solution to the optimization problem (15).

The optimal solution using this formulation is fairly sensitive to b and c . Depending upon the values for b , it is possible that c does not appreciably influence the optimal solution. Instead, the optimal solution can be determined by the nearest Pareto-optimal solution from b . This might require that c be varied parametrically to generate a set of Pareto-optimal solutions.

Solving the goal attainment formulation (15) leads to the approximated objective function value $z(z_{Approx})$. For computing the exact value of (z_{Exact}) , in order to obtain the accuracy of the discrete-time approximation technique, we should do the following approach. After solving the optimization problem (15) and obtaining λ^* , we compute $P_1(t)$ from equation (9). Then, the exact mean and the variance of the project completion time are computed from (10) and (11), respectively. Finally, z_{Exact} is given by

$$z_{Exact} = Max \left\{ \frac{PDC - b_1}{c_1}, \frac{E(T) - b_2}{c_2}, \frac{Var(T) - b_3}{c_3} \right\}. \quad (16)$$

3. SIMULATED ANNEALING ALGORITHM

3.1. INTRODUCTION

Simulated annealing (SA) is a stochastic relaxation technique that has its origin in statistical mechanics (refer to Kirkpatrick *et al.* [22] and Koulamas *et al.* [23]). The SA methodology draws its analogy from the annealing process of solids. In the annealing process, a solid is heated to a high temperature and gradually cooled to low it to crystallize. As the heating process allows the atoms to move randomly, if the cooling is done too rapidly, it gives the atoms enough time to align themselves in order to reach a minimum energy state that named stability or equipment. This analogy can be used in combinatorial optimization, in which the state of solid corresponding to the feasible solution, the energy at each state corresponding to the improvement in the objective function and the minimum energy state will be the optimal solution.

SA uses a stochastic approach to direct the search. It allows the search to proceed to a neighboring state even if the move causes the value of the objective function becomes worse. SA guides the original local search method in the following way. If a move to a neighbor X' in the neighborhood $N(X)$ decreases the objective function value, or leaves it unchanged, then the move is always accepted.

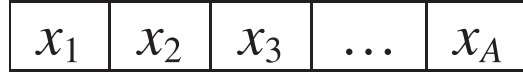


FIGURE 1. Solution representation.

```

SUB InitTemp()
DO
  Generate two solutions  $X_1$  and  $X_2$  at
  random
LOOP UNTIL ( $Z_{APP}(X_1) \neq Z_{APP}(X_2)$ )
  Set  $T_0 = -1 Z_{APP}(X_1) - Z_{APP}(X_2) / \ln(0.9)$ 
END SUB

```

FIGURE 2. Generation of initial temperature subroutine-pseudo code.

More precisely, the solution X' is accepted as the new current solution if $\Delta \leq 0$ where $\Delta = C(X') - C(X)$ and $C(X)$ is the value of objective function. Moves, which increase the objective function value, are accepted with a probability of $e^{(-\Delta/T)}$ to allow the search to escape a local optimum; where T is a parameter named the temperature. The value of T varies from a relatively large value to a small value close to zero. These values are controlled by a cooling schedule that specifies the initial and incremental temperature values at each stage of the algorithm.

3.2. SA IMPLEMENTATION

3.2.1. Solution representation

As shown in Figure 1, the solution is represented by an array with A entries that A is the number of activities in PERT network. The value of i th entry is the amount of resource allocated to activity i where $L_i \leq x_i \leq U_i$.

3.2.2. Initial temperature

The initial temperature is generated according to a heuristic procedure, which is shown in Figure 2. This procedure ensures that in primary iterations of SA algorithm, the probability of accepting the non-improver solutions is about 90 percent. Because, according to the **InitTemp()** presented in Figure 2, we have $e^{\frac{-\Delta Z_{APP}}{T_r}} \approx 0.9$.

3.2.3. Neighborhood search

For exploitation of solution space and generating the neighbor solutions, we use an arithmetic mutation, in which the value of allocated resource to a randomly

```

SUB MUTATION( $X^n = (x_1^n, x_2^n, \dots, x_a^n, \dots, x_A^n)$ )
  Select activity  $a$  at random
  choose  $u$  randomly at interval (0,1)
  choose  $\lambda$  randomly at interval (0,1)
  If  $u > 0.5$  Then
     $y_a^n = \lambda x_a^n + (1-\lambda)U_a$ 
  Else
     $y_a^n = \lambda L_a + (1-\lambda)x_a^n$ 
  End If
   $X^n = (x_1^n, x_2^n, \dots, y_a^n, \dots, x_A^n)$ 
END SUB

```

FIGURE 3. Generation of neighborhood solution subroutine-pseudo code.

selected activity is changed base on the linear combination of the current value of allocated resource and its upper or lower bound. The related procedure is shown in Figure 3.

3.2.4. Temperature updating rule

We consider a well-known temperature updating rule introduced by Gutzmann [20]. According to the Gutzmann rule, the rate of decreasing temperature at each iteration is as below:

$$\alpha = \sqrt[A]{\left(\frac{T_f}{T_0}\right)^{1/MTT}}$$

where A is the number of activities, T_0 is the initial temperature and T_f is the final temperature. MTT is the maximum number of consecutive temperature trails that is a stoppage rule.

3.2.5. SA algorithm

The principle advantage of simulated annealing algorithm is escaping of local optimums by acceptance of no improver solutions with respect to the certain probability in each temperature. The SA algorithm has two inside and outside loops. The inside loop controls the achievement to equilibrium in the current temperature and outside loop controls the rate of temperature decrease. The SA parameters are as follows:

EL (Epoch Length): Number of accepted solutions in each temperature for achieving to equipment.

MTT: Maximum number of consecutive temperature trails.

```

r = 0, T = T0, Xbest = ∅, EL=100, MTT=20
Generate X0
Xbest = X0
Call InitTemp()
Ti = T0/2
α = √[(Tf/T0)1/MTT]; A= number of activities (Gutzmann, 1987)
Do (Outside loop)
  n = 0
  Do (Inside loop)
    Xnew = MUTATION(Xn)
    ΔZAPP = ZAPP(Xnew) - ZAPP(Xn)
    If ΔZAPP < 0 Then
      n = n + 1 and Xn = Xnew
      If ZAPP(Xn) - ZAPP(Xbest) < 0 Then Xbest = Xn
    Else
      Generate y → U(0,1) Randomly
      Set z = e $\frac{-\Delta Z_{APP}}{T_r}$ 
      If y < z Then n = n + 1 and Xn = Xnew
    End if
  Loop While( n < EL)
  r = r + 1
  Tr = α × Tr-1
Loop While (r < MTT and Tr > Tf)
Print Xbest

```

FIGURE 4. SA algorithm pseudo code.

T_0 : Initial temperature.

T_f : Final temperature.

α : Decreasing rate of the temperature (cooling schedule).

X : Typical solution.

$Z_{APP}(X)$: The fitness of solution X with respect to the approximate value of z .

n : Counter for number of accepted solutions in each temperature.

r : Counter for number of consecutive temperature trails, where T_r is equal to temperature in iteration r .

The steps of complete algorithm are shown in Figure 4. Each problem is executed ten times and the best obtained solution is reported.

4. NUMERICAL EXAMPLES

To investigate the performance of the proposed method for the time-cost trade-off problem in PERT networks, we consider 3 typical very small, small and medium-scaled cases with different configurations. Cases I, II and III are shown in Figures 5, 6 and 7, respectively. Tables 1, 2 and 3 show the characteristics of the

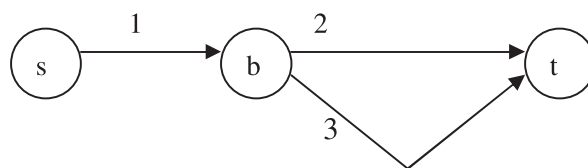


FIGURE 5. Case I.

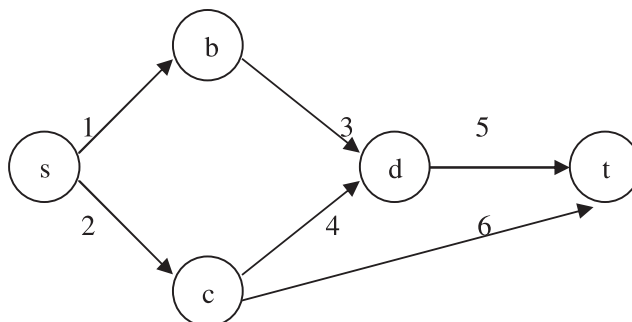


FIGURE 6. Case II.

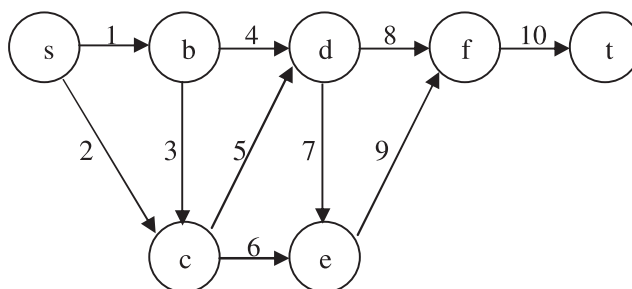


FIGURE 7. Case III.

TABLE 1. Characteristics of the activities in Case I.

A	Distribution	Parameters	$d_a(x_a)$	$g_a(x_a)$	L_a	U_a
1	Exponential	λ_1	$3x_1^2 + 2$	$24 - 5x_1$	1	4
2	Exponential	λ_2	$2x_2 + 1$	$20 - 3x_2$	1	6
3	Generalized Erlang	$(\lambda_{31}, \lambda_{32})$	x_3	$15 - 2x_3$	1	6

activities in Cases I, II and III, respectively. The cost unit is in thousand dollars and the time unit is in months. The objective is to obtain the optimal allocated resource quantities, using the discrete-time approximation technique.

TABLE 2. Characteristics of the exponential activities in Case II.

A	$d_a(x_a)$	$g_a(x_a)$	L_a	U_a
1	$2x_1$	$0.7 - 0.1x_1$	1	5
2	$3x_2 + 1$	$1.5 - 0.2x_2$	1	6
3	$x_3 + 2$	$1 - 0.1x_3$	1	9
4	x_4	$1.5 - 0.3x_4$	1	4
5	$3x_5 + 4$	$0.9 - 0.1x_5$	1	5
6	$x_6 + 3$	$1.1 - 0.1x_6$	1	6

TABLE 3. Characteristics of the exponential activities in Case III.

a	$d_a(x_a)$	$g_a(x_a)$	L_a	U_a
1	$2x_1$	$0.7 - 0.1x_1$	1.5	3
2	$3x_2 + 1$	$1.5 - 0.2x_2$	1.5	3
3	$x_3 + 2$	$1 - 0.1x_3$	1.5	3
4	x_4	$1.5 - 0.3x_4$	1.5	3
5	$3x_5 + 4$	$1.3 - 0.2x_5$	1.5	3
6	$x_6 + 3$	$1.1 - 0.1x_6$	1.5	3
7	$2x_7 + 5$	$1.5 - 0.2x_7$	1.5	3
8	$4x_8 + 1$	$1 - 0.2x_8$	1.5	3
9	$5x_9 + 2$	$0.9 - 0.1x_9$	1.5	3
10	$2x_{10} + 3$	$2 - 0.4x_{10}$	1.5	3

The sizes of the state spaces in Cases I, II and III are equal to 7, 17 and 25, respectively. Then, three factorial experiments according to the following three sets of c :

$$c1 : (c1 = 0.909, c2 = 0.0455, c3 = 0.0455)$$

$$c2 : (c1 = 0.7693, c2 = 0.0769, c3 = 0.1538),$$

$$c3 : (c1 = 0.8929, c2 = 0.0178, c3 = 0.0893),$$

and also the following two sets of b in the three indicated cases:

$$\text{Case I: } b1 : (b1 = 25, b2 = 8, b3 = 25), b2 : (b1 = 40, b2 = 7, b3 = 20),$$

$$\text{Case II: } b1 : (b1 = 40, b2 = 1.5, b3 = 0.7), b2 : (b1 = 35, b2 = 1.6, b3 = 0.8),$$

$$\text{Case III: } b1 : (b1 = 65, b2 = 5, b3 = 3.5), b2 : (b1 = 70, b2 = 4.5, b3 = 3),$$

are designed to obtain a set of Pareto-optimal solutions in each case. For example, using the first set of c leads to the following considerations: one month deviation from the mean project completion time is as important as its variance and 20 times as important as one thousand dollars deviation from the project direct cost, respectively.

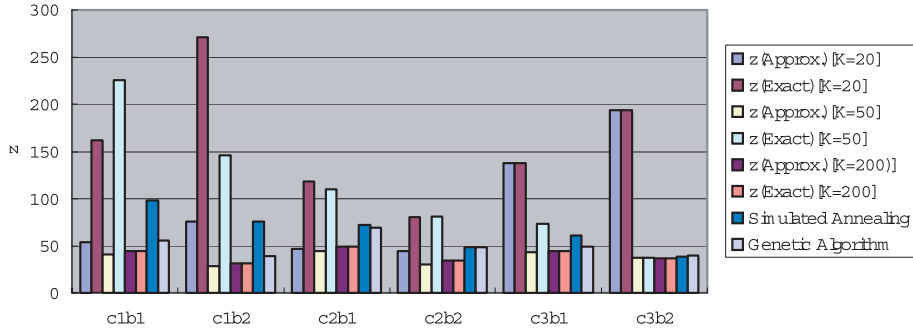


FIGURE 8. Objective function values of Case I.

We consider the following levels of K ($K = 20$, $K = 50$ and $K = 200$) in our computational experiments. $P_1(K)$ should be greater than 0.99 in all experiments. To satisfy this property, we consider the following combinations of K and Δt in Cases I, II and III, respectively: $(K = 20, \Delta t = 2)$, $(K = 20, \Delta t = 0.25)$ and $(K = 20, \Delta t = 0.5)$, if $K = 20$, $(K = 50, \Delta t = 0.8)$, $(K = 50, \Delta t = 0.1)$ and $(K = 50, \Delta t = 0.2)$, if $K = 50$, and finally $(K = 200, \Delta t = 0.2)$, $(K = 200, \Delta t = 0.025)$ and $(K = 200, \Delta t = 0.05)$, if $K = 200$. LINGO 6 is used to compute the approximated objective function values ($Z_{Approx.}$) and the related computational times, according to the goal attainment formulation (15) for the six indicated combinations of b and c in each case. The exact objective function values (Z_{Exact}) are also computed from equation (16).

We also solve the problem using both the simulated annealing and the genetic algorithm, proposed by Azaron *et al.* [1], and compare the discrete-time approximation results against the simulated annealing and the genetic algorithm results.

Figures 8, 9 and 10 show the approximated and the exact objective function values in the discrete-time approximation technique for the six combinations of b and c in Cases I, II and III, respectively, considering $K = 20$, 50 and 200, and also the objective function values using the simulated annealing and the genetic algorithm.

Figures 11, 12 and 13 show the related computational times (sec.) on a PC Pentium IV 2.1 GHz for the six combinations of b and c in Cases I, II and III, respectively, using the three mentioned techniques.

According to Figures 8, 9 and 10, in almost all combinations of the three indicated cases, the approximated and the exact objective function values in the discrete-time approximation technique are decreased, when we increase K , or the accuracy of the discrete-time approximation method is increased. The differences between $Z_{Approx.}$ and Z_{Exact} are also decreased, when K is increased. Moreover, in all cases, the objective function values obtained using the discrete-time approximation technique with $K = 200$, which seem to be optimal or very near to optimal solutions, are less than the values obtained by the simulated annealing and

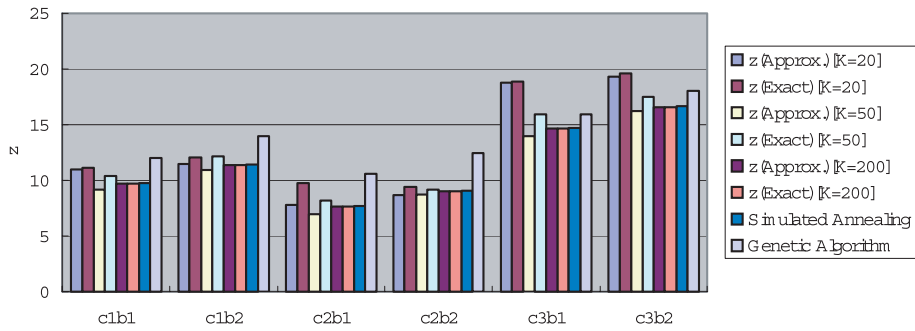


FIGURE 9. Objective function values of Case II.

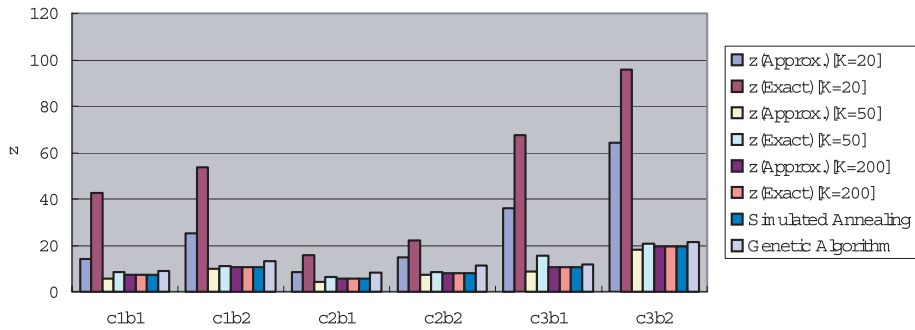


FIGURE 10. Objective function values of Case III.

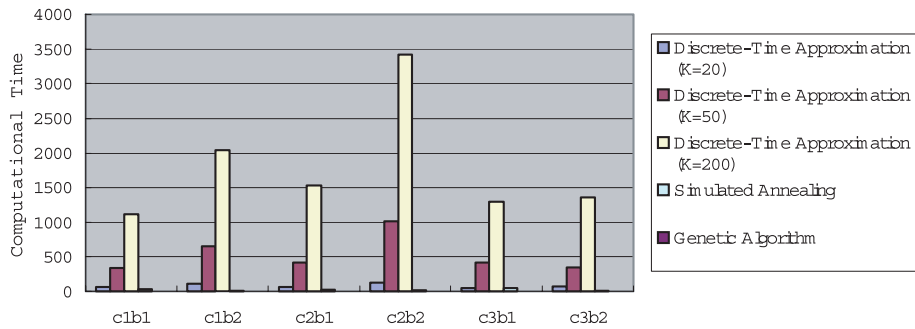


FIGURE 11. Computational times of Case I.

the genetic algorithm, especially when there is any activity with non-exponential duration in the PERT network such as Case I.

According to Figures 11, 12 and 13, computational time grows with K . Computational times are also strongly dependent on the network size and also the number

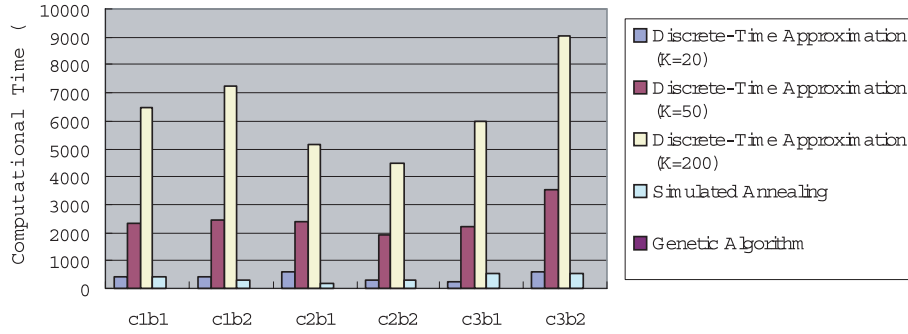


FIGURE 12. Computational times of Case II.

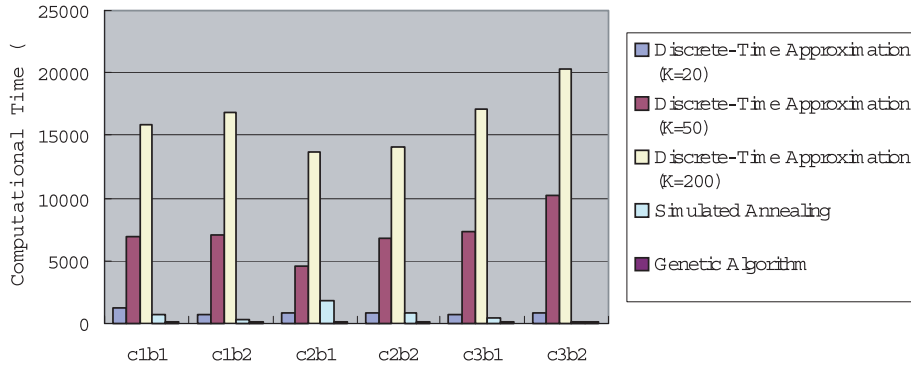


FIGURE 13. Computational times of Case III.

of activities with non-exponential durations in the network. Moreover, comparing the computational times in the discrete-time approximation technique against the simulated annealing and the genetic algorithm shows that the discrete-time approximation technique is not computationally efficient in large-sized problems, especially when using a large value for K . The computational times using the genetic algorithm are much less than the other methods, especially in Cases I and II.

It is totally concluded that if we take some more time or increase the value of K in the discrete-time approximation technique, the quality of results would be much better than the simulated annealing and the genetic algorithm as two heuristic alternatives for solving the problem.

5. CONCLUSIONS

In this paper, we developed a discrete-time approximation technique dealing with the time-cost trade-off problem in PERT networks with generalized Erlang distributions of activity durations.

The multi-objective nonlinear optimization problem was cast into a discrete-time approximation, and the latter model was solved using the goal attainment technique. This method has fewer variables to work with, so it will be computationally faster, and therefore is a good method to solve our problem, which is very complicated to solve even with the presented discrete-time approximation technique.

We solved three different typical cases with different configurations. It was found that in all cases, the approximated and the exact objective function values are decreased, when K is increased. Moreover, the differences between $Z_{Approx.}$ and Z_{Exact} are decreased, when K is increased. We also concluded that the computational time is strongly dependent on the network size and grows with K .

When K approaches to infinity and Δt goes to zero, the approximated discrete project completion time distribution approaches to the original continuous distribution. In this case, the optimal solution of the discrete-time problem also approaches to the optimal solution of the original continuous-time problem, accordingly. An optimal difference scheme is the minimum value for Δt , in which $P_1(K\Delta t) \geq 1 - \epsilon$.

The discrete-time approximation technique is not computationally efficient especially for solving large scale problems. As indicated, the computational times using the simulated annealing and the genetic algorithm are significantly lower than the discrete-time approximation technique, but the quality of the discrete-time approximation results is better than those heuristic algorithms, when K is increased.

As mentioned, solving the original optimal control problem, optimally, is impossible. Therefore, we should have either used the discrete-time approximation technique or a heuristic approach to solve the problem. The efficiency of any heuristic approach for solving large-sized problems can be tested by comparing its results against the discrete-time approximation results in small and medium-sized problems. For example, according to our experiments, the quality of results in the simulated annealing is mainly better than the genetic algorithm, while the computational times using the genetic algorithm are less than the simulated annealing.

As explained, each general distribution can be approximated by a phase-type distribution. Therefore, our model can be easily extended to the general PERT networks, where general activity durations are allowed. In general PERT networks, the non-exponential distributions are approximated by the appropriate generalized Erlang distributions, first, (by matching the first three moments) and then our proposed model is applied to obtain the optimal allocated resource quantities.

Another multi-objective technique like the surrogate worth trade-off method, method of satisfactory goals or STEM could also be applied to solve our multi-objective problem (see Azaron *et al.* [2], which is dealing with the surrogate worth trade-off method).

Acknowledgements. This research is partially supported by the Killam Trusts, Dalhousie University.

REFERENCES

- [1] A. Azaron, C. Perkgoz and M. Sakawa, A Genetic Algorithm Approach for the Time-Cost Trade-Off in PERT Networks. *Appl. Math. Comput.* **168** (2005) 1317–1339.
- [2] A. Azaron, H. Katagiri, M. Sakawa, K. Kato and A. Memariani, A Multi-objective Resource Allocation Problem in PERT Networks. *Eur. J. Oper. Res.* **172** (2006) 838–854.
- [3] E. Berman, Resource Allocation in a PERT Network Under Continuous Activity Time-cost Function. *Management Sci.* **10** (1964) 734–745.
- [4] J. Burt and M. Garman, Conditional Monte Carlo: A Simulation Technique for Stochastic Network Analysis. *Management Sci.* **18** (1977) 207–217.
- [5] A. Charnes, W. Cooper and G. Thompson, Critical Path Analysis via Chance Constrained and Stochastic Programming. *Oper. Res.* **12** (1964) 460–470.
- [6] D. Chau, W. Chan and K. Govindan, A Time-Cost Trade-Off Model with Resource Consideration Using Genetic Algorithm. *Civil Engineering Systems* **14** (1997) 291–311.
- [7] C. Clingen, A Modification of Fulkerson’s Algorithm for Expected Duration of a PERT Project when Activities Have Continuous d. f. *Oper. Res.* **12** (1964) 629–632.
- [8] E. Demeulemeester, W. Herroelen and S. Elmaghraby, *Optimal Procedures for the Discrete Time-Cost Trade-Off Problem in Project Networks*. Research Report, Department of Applied Economics, Katholieke Universiteit Leuven, Leuven, Belgium (1993).
- [9] S. Elmaghraby, On the Expected Duration of PERT Type Networks. *Management Sci.* **13** (1967) 299–306.
- [10] S. Elmaghraby, Resource Allocation via Dynamic Programming in Activity Networks. *Eur. J. Oper. Res.* **64** (1993) 199–245.
- [11] J. Falk and J. Horowitz, Critical Path Problem with Concave Cost Curves. *Management Sci.* **19** (1972) 446–455.
- [12] S. Fatemi Ghomi and S. Hashemin, A New Analytical Algorithm and Generation of Gaussian Quadrature Formula for Stochastic Network. *Eur. J. Oper. Res.* **114** (1999) 610–625.
- [13] S. Fatemi Ghomi and M. Rabbani, A New Structural Mechanism for Reducibility of Stochastic PERT Networks. *Eur. J. Oper. Res.* **145** (2003) 394–402.
- [14] C. Feng, L. Liu, and S. Burns, Using Genetic Algorithms to Solve Construction Time-Cost Trade-Off Problems. *J. Construction Engineering and Management, ASCE* **11** (1997) 184–189.
- [15] G. Fishman, Estimating Critical Path and Arc Probabilities in Stochastic Activity Networks. *Naval Res. Logistic Quarterly* **32** (1985) 249–261.
- [16] D. Fulkerson, A Network Flow Computation for Project Cost Curves. *Management Sci.* **7** (1961) 167–178.
- [17] D. Fulkerson, Expected Critical Path Lengths in PERT Networks. *Oper. Res.* **10** (1962) 808–817.
- [18] M. Garman, More on Conditioned Sampling in the Simulation of Stochastic Networks. *Management Sci.* **19** (1972) 90–95.
- [19] D. Golenko-Ginzburg and A. Gonik, A Heuristic for Network Project Scheduling with Random Activity Durations Depending on the Resource Reallocation. *Inter. J. Production Economics* **55** (1998) 149–162.
- [20] K. Gutzmann, Combinatorial Optimization Using a Continuous State Boltzmann Machine. *Proceedings of IEEE First Int. Conf. Neural Networks*, Vol. III, San Diego, CA (1987) 721–728.
- [21] J. Kelly, Critical Path Planning and Scheduling: Mathematical Basis. *Oper. Res.* **9** (1961) 296–320.
- [22] S. Kirkpatrick, J. Gelatt and M. Vecchi, Optimization by Simulated Annealing. *Science* **220** (1983) 671–680.
- [23] C. Koulamas, S. Antony and R. Jaen, A Survey of Simulated Annealing Applications to Operations Research Problems. *Omega* **22** (1994) 41–56.

- [24] V. Kulkarni and V. Adlakha, Markov and Markov-Regenerative PERT Networks. *Oper. Res.* **34** (1986) 769–781.
- [25] L. Lamberson and R. Hocking, Optimum Time Compression in Project Scheduling. *Management Sci.* **16** (1970) 597–606.
- [26] Z. Laslo, Activity Time-Cost Tradeoffs under Time and Cost Chance Constraints. *Comput. Industrial Engineering* **44** (2003) 365–384.
- [27] J. Martin, Distribution of the Time Through a Directed Acyclic Network. *Oper. Res.* **13** (1965) 46–66.
- [28] M. Neuts, *Matrix-geometric Solutions in Stochastic Models*. Johns Hopkins University Press, Baltimore, MD (1981).
- [29] C. Perry, I. Creig, Estimating the Mean and Variance of Subjective Distributions in PERT and Decision Analysis. *Management Sci.* **21** (1975) 1477–1480.
- [30] P. Pontrandolfo, Project Duration in Stochastic Networks by the PERT-Path Technique. *Inter. J. Project Management* **18** (2000) 215–222.
- [31] P. Robillard, Expected Completion Time in PERT Networks. *Oper. Res.* **24**, (1976) 177–182.
- [32] D. Robinson, A Dynamic Programming Solution to Cost-Time Trade-Off for CPM. *Management Sci.* **22** (1965) 158–166.
- [33] S. Sethi, and G. Thompson, *Optimal Control Theory*. Martinus Nijhoff Publishing, Boston (1981).
- [34] C. Sigal, A. Pritsker and J. Solberg, The Use of Cutsets in Monte-Carlo Analysis of Stochastic Networks. *Mathematical Simulation* **21** (1979) 376–384.
- [35] C. Schmit, and I. Grossmann, The Exact Overall Time Distribution of a Project with Uncertain Task Durations. *Eur. J. Oper. Res.* **126** (2000) 614–636.
- [36] L. Tavares, Optimal Resource Profiles for Program Scheduling. *Eur. J. Oper. Res.* **29** (1987) 83–90.
- [37] J. Weglarz, Project Scheduling with Continuously Divisible Doubly Constrained Resources. *Management Sci.* **27** (1981) 1040–1053.