Resource allocation in multi-server dynamic PERT networks using multi-objective programming and Markov process

S. Yaghoubi¹, S. Noori¹ and M. Bagherpour¹*

¹Department of Industrial Engineering, Iran University of Science and Technology, Tehran, Iran
E-mail: bagherpour@iust.ac.ir

Abstract

In this research, both resource allocation and reactive resource allocation problems in multi-server dynamic PERT networks are analytically modeled, where new projects are expected to arrive according to a Poisson process, and activity durations are also known as independent random variables with exponential distributions. Such system is represented as a queuing network, where multi servers at each service station are allocated, and also each activity of a project is operated at a devoted service station with only one server located at a node of the network based on First Come First Serve (FCFS) policy. In order to propose a novel approach for modeling of multi-server dynamic PERT network, initially the network of queues is transformed into a stochastic network. Then, a differential equations system is organized to solve and obtain approximate completion time distribution for any particular project by applying an appropriate finite-state continuous-time Markov model. Finally, a multi-objective model including four conflicted objectives is presented to optimally control the resources allocated to the service stations in a multi-server dynamic PERT network, and the goal attainment method is further employed to solve a discrete-time approximation of the primary multi-objective problem.

Keywords: Project management; Markov processes; multiple objective programming; reactive resource allocation

1. Introduction

Some organizations are project-oriented based and operate their activities depending on projects. In such situations, the organizations may carry out the multi project concurrently, whereas, Payne [1] revealed that up to 90% organizations execute the projects in a multi-project environment. Commonly, the limited resources are shared and competed among multiple projects for achieving their own goals.

Therefore, multi-project management system is a vital approach in project scheduling and management, whereas traditional project scheduling has been concerned mostly with single project optimization. Multi-project resource constrained scheduling problem (MPRCSP) is the main topic of most investigations on multi-project scheduling considering static and deterministic environments. Pritsker et al. [2] by using a zero-one integer programming approach and Wiest [3] by presenting an heuristic model, analyzed the MPRCSP. Then, Kurtulus and Davis [4] and Kurtulus and Narula [5] by applying priority rules and defining measures such as the rate of utilization of each resource type and the peak of total resource requirements, studied the MPRCSP.

Also, multi criteria and multi objective modeling is then used in MPRCSP. For example, a 0–1 goal programming is applied by Chen [6] in multi-project resource-constrained scheduling for the maintenance of mineral processes, and a lexicographically two criteria is presented by Lova et al. [7]. Recently, scheduling rules in the static MPRCSP environment were presented by Kanagasahapathi et al. [8] considering performance measures involving mean tardiness and the maximum tardiness of projects involved.

Moreover, heuristic & meta-heuristic algorithms for analyzing MPRCSP were applied [9-14]. Recently, MPRCSP was extended by considering transfer times and its relevant costs by Kruger and Scholl [15].

In the literature, MPRCSP was mostly analyzed on static and deterministic environments and a few investigations have been focused on multi-project scheduling under uncertainty and dynamic conditions. A simulation model for multi-project resource allocation with stochastic activity, as a multi-channel queuing, was presented by Fatemi-Ghomi and Ashjari [16]. Also, a nonlinear mixed-integer programming model for optimizing the multi project resource allocation was proposed by Nozick et al. [17], whereas changing resource allocations affects the probability distribution of activity duration. An event-driven approach was
represented by Kao et al. [18], and also, using Critical Chain Project Management (CCPM) approach, the uncertainty in multi project system was studied by Byali and Kannan [19]. MPRCSP is commonly analysed by either connecting them together into a large single project by the addition of dummy start and end activities or considering the projects as independent and linking them by using an objective function which contains each project individually (probably with appropriate weigh factors) and the corresponding resource constraints.

In many organizations, not only are the activity durations uncertain, but also, new projects dynamically arrive to the project based organizations over the time horizon. Clearly, in this condition, project scheduling procedure would be more difficult and more complex than before. This problem, considered in project-oriented organizations, was studied by Adler et al. [20] by applying simulation. In this investigation, the organization was presented as a “stochastic processing network” with a collection of service stations (work stations) or resources, where one or more identical “servers” for serving projects under a pre-specified discipline, has been settled at each station. The represented organization can be expressed as a queuing network (dynamic PERT network), where each activity is getting the required services, queuing up for access to a resource, or waiting to join a predecessor activity. Such problem is attractive for organizations with similar projects, for example, maintenance projects in which a typical project will be repeated.

Also, the concept of CONWIP (constant work-in-process) is employed by Anavi-Isakow and Golany [21] in dynamic PERT network for controlling projects using simulation study. Authors presented two control mechanisms: CONPIP (Constant Number of Projects In Process) that limits the number of projects, and CONTIP (Constant Time of projects In Process), that restricted the total processing time of all active projects. A risk element was considered in dynamic PERT network by Li and Wang [22] and a multi-objective risk-time-cost trade-off problem was proposed based on general project risk element transmission theory.

Through resource allocation problem in dynamic PERT network, two commonly used approaches exist. The first approach was propounded by Cohen et al. [23, 24], where the resources may work in parallel, i.e., the number of servers and resources allocated in every service station are equal (e.g., electrical work station with electricians, mechanical work station with mechanics, etc.) and the amount of resources available to be allocated to all service stations is constant. They presented near optimal resource allocated to the entities that perform the projects in CONPIP system by using Cross Entropy (CE) based on simulation. We denote this approach as “resources as servers” and in this article, based on this approach a multi-objective model will be proposed.

In the second approach, investigated by Azaron and Tavakkoli-Moghaddam [25], the number of servers in every service station is fixed and resources allocated affect the mean of service times.

Authors presented an analytical multi-objective model for the resource allocation problem in a dynamic PERT network and assumed the activity durations are exponentially distributed random variables, the new projects are generated according to a Poisson process, the number of servers in every service station is either one or infinity and the capacity of the system is infinite. Recently, Yaghoubi et al. [26] modeled the resource allocation problem in dynamic PERT networks, where the capacity of system is finite and projects are generated according to a Poisson process. We denote this approach as “resources affecting servers” and in this article, based on this approach a multi-objective model is proposed.

In both approaches, the uncertainty is considered in the entrance of projects and also in the duration of service stations, whereas other uncertainty such as project network disruption may happen. In this research, for avoiding project network disruption, “reactive resource allocation” is suggested. Along with the project execution, a project may be disposed by considerable unforeseen disruptions, therefore, reactive scheduling (rescheduling), with revising or re-optimizing the initial baseline schedule, aims to adjust the baseline schedule and consequently, overcome the disruptions.

Firstly, reactive scheduling was propounded in manufacturing environments and then it was applied through project scheduling approaches. Comprehensive investigations about reactive scheduling in manufacturing environments have been studied [27-32]. Vieira et al. [29], based on wide variety of experimental and practical investigations, introduced a framework of strategies, policies and methods for reactive scheduling and Aytug et al. [30] by defining different types of uncertainties, proposed a review of rescheduling based approaches. Herroelen and Leus [31] represented the basic aspects for scheduling under uncertain conditions: reactive scheduling, stochastic project scheduling, fuzzy project scheduling, robust (proactive) scheduling and sensitivity analysis. Also, Van de Vonder et al. [32], based on practical design, analysed several predictive-reactive resource-constrained project scheduling procedures.

Various approaches exist in the literature of the reactive scheduling problems. A simple and initial approach is a right shift rule that is removed ahead
in time of all the affected activities [33]. Full rescheduling, the other approach, considers the remainder of activities that are to be completed. The other important approach is minimum perturbation strategy which applies the exact and suboptimal method for minimizing the difference between the revised schedule and the primary schedule [34-36]. Recently, Liu and Shih [37], based on a primary schedule and actual progress, studied resource-constrained construction rescheduling and suggested a new rescheduling optimization model using constraint programming, also, Novas and Heming [38] introduced the repair-based reactive scheduling of industrial batch plants.

Reviewing the above mentioned researches indicated no closely related work was found to analytically analyze multi-server dynamic PERT networks. As the main contribution of this research, we develop a novel approach for the resource allocation problem (or time-cost trade off problem) and also reactive resource allocation problem in multi-server dynamic PERT networks by means of multi-objective programming and Markov process.

Through this investigation, we consider a multiple environment and concurrent projects including new projects, containing all the activities that arrive at the system according to an independent Poisson process. It is also assumed each activity of the projects is performed at a devoted service station located at a node of the network with FCFS policy. It is further assumed different servers are allocated in each service station, while the services processing times (or activity durations) are followed as independently random variables with exponential distributions.

For modeling a multi-server dynamic PERT network, firstly, for obtaining the states of a system, the network of queues is transformed into a stochastic network. Then, a system of differential equations is organized to solve and obtain the approximate completion time distribution for any particular project by creating an appropriate finite-state continuous-time Markov model. Finally, a multi-objective model with four conflicted objectives is presented to optimally control the resources allocated to service stations in a multi-server dynamic PERT networks by means of multi-objective programming and Markov process.

The flow chart of our proposed method for multi-server dynamic PERT network is extensively explained:

**Step 1.** Compute the density function of the sojourn time (waiting time plus activity duration) in each service station. (see appendix A)

**Step 1.1.** If $m_s = 1$, then the queueing system would be an $M/M/1$ queue, and the density function of time spent at the service station $a (w_s(t))$ would be exponentially expressed with parameter $\mu_a - \lambda$, therefore, $w_s(t)$ is calculated as follows:

$$w_s(t) = (\mu_a - \lambda) e^{-(\mu_a - \lambda)t} \quad t > 0, \text{ if } m_s = 1 \quad (1)$$
Step 1.2. If $m_a = \infty$, then the queueing system is $M / M / \infty$, and the density function of time spent in the service station $a$ would be exponentially expressed with parameter $\mu_a$, therefore, $w_a(t)$ is calculated as follows:

$$w_a(t) = \mu_a e^{-\mu_a t} \quad t > 0, \text{ if } m_a = \infty$$  \hspace{1cm} (2)

Step 1.3. If $1 < m_a < \infty$, then the queueing system is $M / M / m_a$, and the density function of time spent in the service station $a$ would be approximately two series exponential with parameters $\left(\frac{m_a \mu_a - \lambda}{\rho_a}\right)$ and $\left(\frac{m_a \mu_a}{\rho_a m_a - 1}\right)$, where $\rho_a = \frac{\lambda}{m_a \mu_a}$. Therefore, $w_a(t)$ is approximately calculated as follows:

$$w_a(t) \approx \left(\frac{m_a \mu_a}{m_a - 1}\right) \left(\frac{m_a \mu_a - \lambda}{\rho_a}\right) e^{-\frac{m_a \mu_a - \lambda}{\rho_a} t} \left(\frac{m_a \mu_a}{m_a - 1}\right) \left(\frac{m_a \mu_a}{\rho_a m_a - 1}\right) e^{-\frac{m_a \mu_a}{\rho_a m_a - 1} t} \quad t > 0$$  \hspace{1cm} (3)

Step 2. Convert the dynamic PERT network as an Activity-on-Node (AoN) structure into a substitute classical PERT network represented as an Activity-on-Arc (AoA) graph.

Step 2.1. By considering the AoN graph, substitute each node with a stochastic arc (activity) whose length is equal to the sojourn time in the corresponding service station. For this purpose, node $a$ in the AoN graph should be replaced with a stochastic activity. Assume $b_1, b_2, \ldots, b_n$ are the incoming arcs to node $a$ and $d_1, d_2, \ldots, d_m$ are the outgoing arcs from it. Then, node $a$ is substituted by activity $(v, w)$, whose length is equal to the sojourn time in the service station $a$. Furthermore, all arcs $b_1, b_2, \ldots, b_n$ terminate with node $v$ while all arcs $d_1, d_2, \ldots, d_m$ begin from node $w$. (For more details, see [40])

Step 2.2. Transform the PERT network, obtained in step 2.1, into a new PERT network with exponentially distributed arc length.

Resources as servers approach: in this approach in which the number of servers and resources allocated in every service station are equal, every arc would be substituted with two series of exponential arc with the parameters $\left(\frac{m_a \mu_a - \lambda}{\rho_a}\right)$ and $\left(\frac{m_a \mu_a}{m_a - 1}\right)$. After replacing all arcs with the proper exponential two series arc, the PERT network obtained in step 2.1, is transformed into a new PERT network.

Resources affecting servers approach: in this approach, the number of servers in every service station is fixed and resources allocated affect the mean of service times. As mentioned in steps 1.1 and 1.2, If one or infinite servers be in the work station, then the length of arc would be exponential with parameters $\mu_a - \lambda$ and $\mu_a$, respectively, and the corresponding arc would not be changed. But, if several servers ($1 < m_a < \infty$) be in the work station, then the corresponding arc would be substituted with two series of exponential arc with the parameters $\left(\frac{m_a \mu_a - \lambda}{\rho_a}\right)$ and $\left(\frac{m_a \mu_a}{m_a - 1}\right)$. After replacing all such arcs with the proper exponential two series arc, the PERT network obtained in step 2.1, is transformed into a new PERT network.

Step 3. Determine a continuous-time Markov process with finite states.

Step 3.1. Determine the states space of system. For this purpose, let $G'(V, A')$ be the PERT network, obtained in step 2.1, with a single source and a single sink, in which $V'$ represents the set of nodes and $A'$ represents the set of arcs of the network in the AoA network. Also, let $G = (V, A)$ be a new PERT network, obtained in step 2.2, in which $V$ represents the set of nodes and $A$ represents the set of arcs of the network in a new AoA graph. Let $s$ and $t$ be the source and sink nodes in the new PERT network, respectively, and the length of arc $a \in A$ be a random variable that is exponentially distributed with parameter $\gamma_a$. For $a \in A$, the starting node and the ending node of arc $a$, are denoted as $\alpha(a)$ and $\beta(a)$, respectively. Henceforth in this section, we analyze the new PERT network to determine a continuous-time Markov process with finite state space.

Definition 1. Let $I(v)$ be the set of arcs ending at node $v$ and $O(v)$ be the set of arcs starting at node $v$ in the new PERT network, which are defined as follows: (see [39])

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

$$O(v) = \{a \in A : \alpha(a) = v\} \quad (v \in V).$$  \hspace{1cm} (4)
Definition 2. For $X \subseteq V$ such that $s \in X$ and $t \in \overline{X} = V - X$, an $(s,t)$ cut is defined as follows:

$$(X, \overline{X}) = \{a \in A : \alpha(a) \in X, \beta(a) \in \overline{X}\}. \quad (5)$$

An $(s,t)$ cut $(X, \overline{X})$ is denominated a uniformly directed cut (UDC), if $(\overline{X}, X) = \emptyset$, i.e. no two arcs in the cut belong to the same path in the project network. Each UDC is clearly a set of arcs, in which the starting node of each arc belongs to $X$ and the ending node of each arc belongs to $\overline{X}$.

Example 1. Consider the network shown in Fig. 2 taken from [25]. According to the definition, the UDcs of this network are $(1, 2), (2, 3), (1, 4, 6), (3, 4, 6)$ and $(5, 6)$.

Definition 3. An $(E,F)$, subsets of $A$, is defined as admissible 2-partition of a UDC $D$ if $D = E \cup F$ and $E \cap F = \emptyset$, and also $I(\beta(a)) \not\subseteq F$ for any $a \in F$.

![Fig. 1. Structure of the proposed method](image1)

![Fig. 2. The example network](image2)
Definition 4. Along with the project execution at time $t$, each activity (arc) can be in one and only one of the active, dormant or idle states, which are defined as follows:

(i) Active: an activity $a$ is active at time $t$ if it is being performed at time $t$.

(ii) Dormant: an activity $a$ is called dormant at time $t$ if it has completed but there is at least one unfinished activity in $I(\beta(a))$ at time $t$.

(iii) Idle: an activity $a$ is denominated idle at time $t$ if it is neither active nor dormant at time $t$.

Also, $Y(t)$ and $Z(t)$ are defined as follows:

$$Y(t) = \{a \in A : a \text{ is active at time } t\},$$
$$Z(t) = \{a \in A : a \text{ is dormant at time } t\},$$

and $X(t) = (Y(t), Z(t))$.

All admissible 2-partition cuts of the network of Fig. 2 are presented in Table 1. A superscript star is applied to denote a dormant activity and all others are active. $E$ and $F$ contain all active and all dormant activities, respectively.

The set of all admissible 2-partition cuts for the network are defined as $S$ and also $\bar{S} = S \cup \{\phi, \phi\}$. Note that $X(t) = (\phi, \phi)$ presents that the all activities are idle at time $t$ and therefore the project is finished by time $t$. It is demonstrated that $X(t), t \geq 0$ is a finite-state absorbing continuous-time Markov process. (for more detail, see [38])

Step 3.2. Obtain the system of differential equations.

As previously mentioned, a UDC is divided into $E$ and $F$ that contain active and dormant activities, respectively. If activity $a$ terminates (with the rate of $\gamma_a$), and $I(\beta(a)) \subset F \cup \{a\}$, there is at least one unfinished activity in $I(\beta(a))$, then

$$E' = E - \{a\}, F' = F \cup \{a\}.$$  
Furthermore, if by completing activity $a$, all activities in $I(\beta(a))$ become idle

$$E' = (E - \{a\}) \cup O(\beta(a)), F' = F - I(\beta(a)).$$

Namely, all activities in $I(\beta(a))$ will become idle and also the successor activities of this activity, $O(\beta(a))$, will become active. Therefore, the components of the infinitesimal generator matrix

$$Q = \begin{bmatrix} Y_a & -\sum_{a \in Y_a} \gamma_a \\ 0 & \gamma_a \end{bmatrix},$$

$(E, F)$ and $(E', F') \in \bar{S}$ are obtained as follows:

$$q^A(E,F, (E',F')) = \begin{cases} \gamma_a, & (a \in Y_a) \\ -\sum_{a \in Y_a} \gamma_a, & (a \not\in Y_a) \end{cases}$$

Also, $q^A(E, F, (E, F')) = 0$. The project is completed. In this Markov process all of the states except $X(t) = (\phi, \phi)$ which is an absorbing state, are transient. Furthermore, the states in $\bar{S}$ should be numbered such that this $Q$ matrix be an upper triangular one. It is assumed that the states are numbered as $1, 2, \ldots, N = |\bar{S}|$ so that $X(t) = (O(s), \phi)$ and $X(t) = (\phi, \phi)$ are state 1 (initial state) and state $N$ (absorbing state), respectively.

Let $T$ be the length of the longest path or the project completion time in the new PERT network, obtained in step 2.2. Obviously, $T = \min \{t \geq 0 : X(t) = N \, \vert \, X(0) = I \}$. Chapman–Kolmogorov backward equations can be used to calculate $F(t) = P(T \leq t)$. If it is defined

$$P_i(t) = P(X(t) = N \, \vert \, X(0) = i), \quad i = 1, 2, \ldots, N,$$

then $F(t) = P_i(t)$.

The system of linear differential equations for the vector $P(t) = [P_1(t) \, P_2(t) \, \ldots \, P_n(t)]^T$ is presented as follows:

$$P'(t) = \begin{bmatrix} \frac{dP_1(t)}{dt} \\ \vdots \\ \frac{dP_n(t)}{dt} \end{bmatrix} = Q \cdot P(t),$$

with

$$P(0) = \begin{bmatrix} 0 & 0 & \ldots & 1 \end{bmatrix}^T,$$

where $P'(t)$ and $Q$ represent the derivation of the state vector $P(t)$ and the infinitesimal generator matrix of the stochastic process $\{X(t), t \geq 0\}$, respectively.

2.2. Multi-objective resource allocation

In this paper, following the research presented by Azaron and Tavakkoli-Moghaddam [25], we propose a multi-objective model to optimally
control the resources allocated to the service stations in a dynamic PERT network based on the mentioned two approaches.

### 2.2.1. Resources as servers

We propose a multi-objective model to optimally control the servers allocated (as resources) to the service stations in a dynamic PERT network, represented as a network of queues, where we allocate more servers to the service station, the mean time spent (sojourn time) in the service station will be decreased and direct cost will be increased, whereas the direct cost of each activity is a non-decreasing function of the amount of the allocated server. Note that the mean of activity duration $\mu_a$ is a constant value.

If we decrease the amount of resource allocated (servers) to the service stations, the project direct cost will therefore be decreased. Conversely, the mean project completion time will then be increased, because these objectives are in conflict with each other. Therefore, the total direct costs and the mean project completion time are dependent on each other and an appropriate trade-off between them is required. Another effective objective that should also be included in the model, is the variance of the project completion time, because the mean and the variance are two complementary concepts. The last objective that should also be considered is the probability that the project completion time does not exceed a certain threshold for on-time delivery performance. Let $P_1(t) = \mathbb{P}(T \leq t)$.

Therefore, this is a multi-objective stochastic programming problem. The objective functions are given as follows:

1. Minimizing the project direct cost
   \[
   \min f_1(m) = \sum_{a \in A'} d_a(m_a) 
   \]

2. Minimizing the mean of project completion time
   \[
   \min f_2(m) = \mathbb{E}(T) = \int_0^\infty (1 - P_1(t)) dt = \int_0^\infty t P_1(t) dt 
   \]

3. Minimizing the variance of project completion time
   \[
   \min f_3(m) = \text{Var}(T) = \int_0^\infty \int_0^\infty (t - \mathbb{E}(T))^2 P_1(t) dt 
   \]

4. Maximizing the probability that the project completion time does not exceed a certain threshold
   \[
   \max f_4(m) = P_1(u) = P(T \leq u) 
   \]

The infinitesimal generator matrix $Q$ would be a function of the control vector $m = [m_a : a \in A']$. Therefore, the non-linear dynamic model is

\[
P'(t) = Q(m)P(t) \quad P_1(0) = 0 \quad \forall i = 1, 2, ..., N - 1 \quad P_1(t) = 1
\]

The next constraint should be regarded to guarantee having a response in the steady-state.

\[
0.3 \leq \rho_a = \frac{\lambda}{m_a \mu_a} \times 1 \Rightarrow \frac{\lambda}{\mu_a} < \frac{\lambda}{0.3 m_a} \quad \forall a \in A'
\]

In the mathematical programming, we do not use such constraints. Hence, $\exists \epsilon$ following the establishment of constraint

\[
\lambda - 0.3 m_a \mu_a \geq \epsilon \quad \forall a \in A' 
\]

**Table 1.** All admissible 2-partition cuts for the example network

<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>(1,2)</td>
<td>5.</td>
<td>(1,4'),(6)</td>
<td>9.</td>
<td>(3',4,6)</td>
<td>13.</td>
<td>(3,4*,6*)</td>
<td>17.</td>
<td>(ϕ,ϕ)</td>
</tr>
<tr>
<td>2.</td>
<td>(2,3)</td>
<td>6.</td>
<td>(1,4,6')</td>
<td>10.</td>
<td>(3,4',6)</td>
<td>14.</td>
<td>(5,6)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>(2,3')</td>
<td>7.</td>
<td>(1,4',6')</td>
<td>11.</td>
<td>(3,4',6)</td>
<td>15.</td>
<td>(5',6)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>(1,4,6)</td>
<td>8.</td>
<td>(3,4,6)</td>
<td>12.</td>
<td>(3',4,6')</td>
<td>16.</td>
<td>(5,6')</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Consequently, the appropriate multi-objective optimal control problem is expressed as below:

\[
\begin{align*}
\text{Min } f_1(m) &= \sum_{a \in A} d_s(m_a) \\
\text{Min } f_2(m) &= \int_0^T tP_1(t) \, dt \\
\text{Min } f_3(m) &= \int_0^T t^2P_1(t) \, dt - \left[ \int_0^T tP_1(t) \, dt \right]^2 \\
\text{Max } f_4(m) &= P_1(u) \\
P(t) &= Q(m), P(t) \\
P_i(0) &= 0 \quad \forall i = 1, 2, \ldots, N - 1 \\
P_i(t) &= 1 \\
m_a, \mu_a - \lambda &\geq \varepsilon \quad \forall a \in A' \\
\lambda - 0.3m_a, \mu_a &\geq 0 \quad \forall a \in A' \\
m_a &\geq L_a \quad \forall a \in A' \\
m_a &\leq U_a \quad \forall a \in A' \\
\sum_{a \in A'} m_a &\leq M \\
m_is \text{ integer} \quad \forall a \in A'
\end{align*}
\]

This continuous-time stochastic programming is impossible to solve (for more details see [25]), therefore, based on the definition of integral thinking, we divide the time interval into \( R \) equal portions with the length of \( t \). Indeed, we transform the differential equations into difference equations. Thus, the corresponding discrete state model can be given as follows:

\[
\begin{align*}
\text{Min } f_1(m) &= \sum_{a \in A} d_s(m_a) \\
\text{Min } f_2(m) &= \sum_{r=0}^{R-1} r\Delta t(P_r(r+1) - P_r(r)) \\
\text{Min } f_3(m) &= \sum_{r=0}^{R-1} (r\Delta t)^2(P_r(r+1) - P_r(r)) \\
&\quad - \left[ \sum_{r=0}^{R-1} r\Delta t(P_r(r+1) - P_r(r)) \right]^2 \\
\text{Max } f_4(m) &= P_1\left( \frac{u}{\Delta t} \right) \\
\text{s.t.: } P(r+1) &= P(r) + Q(m)P(r)\Delta t \quad r = 0, 1, \ldots, R - 1 \\
P(0) &= 0 \quad \forall i = 1, 2, \ldots, N - 1 \\
P_i(r &= 1 \quad r = 0, 1, \ldots, R \\
P_i(r &\leq 1 \quad i = 1, \ldots, N - 1, \quad r = 0, 1, \ldots, R \\
m_a, \mu_a - \lambda &\geq \varepsilon \quad \forall a \in A' \\
\lambda - 0.3m_a, \mu_a &\geq 0 \quad \forall a \in A' \\
m_a &\geq L_a \quad \forall a \in A' \\
m_a &\leq U_a \quad \forall a \in A' \\
\sum_{a \in A'} m_a &\leq M \\
m_is \text{ integer} \quad \forall a \in A'
\end{align*}
\]

2.2.2. Resources affecting servers

In this section, we propose a multi-objective to optimally control the resources allocated to the service stations based on resources affecting servers approach in a multi-server dynamic PERT network, represented as a network of queues. The direct cost of each activity is a non-decreasing function and the mean service time in each service station is a non-increasing function of the amount of resource allocated to it.

Let \( x_a \) be the resource allocated in service station \( a \quad (a \in A') \), also \( d_s(x_a) \) be the direct cost of activity \( a \in A' \) in the PERT network, obtained in section 2.1 step 2.1, while it is assumed to be a non-decreasing function of amount of resources \( x_a \) allocated to it. Thus, the project direct cost (PDC) would be \( \text{PDC} = \sum_{a \in A'} d_s(x_a) \). Also, the mean service time in the service station \( a \in A' \), \( g_s(x_a) \) is assumed to be a non-increasing function of the amount of resource \( x_a \) allocated to it that would be equal to

\[
g_s(x_a) = \frac{1}{\mu_a} \quad \forall a \in A'.
\]

Let \( U_a \) be the maximum amount of resource available to be allocated to the activity \( a \quad (a \in A') \), \( L_a \) be the minimum amount of resource needed to execute the activity \( a, \ x = [x_a : a \in A']^T \) and \( J \) represent the amount of resource available to be allocated to all of activities. Moreover, we define \( u \) as a threshold time that project completion time does not exceed. Let \( B \) be the set of arcs in the PERT network, obtained in section 2.1 step 2.1, where there are infinite servers settled on the corresponding service station. The next constraint should be satisfied to keep the response in the steady-state.

\[
m_a, \mu_a - \lambda &\geq \varepsilon \quad \forall a \in A' - B \\
\lambda - 0.3m_a, \mu_a &\geq 0 \quad \forall a \in A' - B \\
m_a &\geq L_a \quad \forall a \in B \\
\mu_a &\geq \varepsilon \quad \forall a \in B
\]

In practice, \( d_s(x_a) \) and \( g_s(x_a) \) can be obtained by employing linear regression based on the previous similar activities or applying the judgments of experts in this area.

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

\[
\text{Min } f_1(m) = \sum_{a \in A}^{} \frac{x_a}{\mu_a} \quad \forall a \in A' \\
\text{Min } f_2(m) = \int_0^T t^2P_1(t) \, dt - \left[ \int_0^T tP_1(t) \, dt \right]^2 \\
\text{Max } f_4(m) = P_1(u) \\
P(t) = Q(m), P(t) \\
P_i(0) = 0 \quad \forall i = 1, 2, \ldots, N - 1 \\
P_i(t) = 1 \\
m_a, \mu_a - \lambda &\geq \varepsilon \quad \forall a \in A' \\
\lambda - 0.3m_a, \mu_a &\geq 0 \quad \forall a \in A' \\
m_a &\geq L_a \quad \forall a \in A' \\
m_a &\leq U_a \quad \forall a \in A' \\
\sum_{a \in A'} m_a &\leq M \\
m_is \text{ integer} \quad \forall a \in A'
\]
2.3. Goal attainment method

We now need to apply a multi-objective method to solve the proposed models, and we actually apply goal attainment technique for this purpose. Assume there is a multi-objective programming with \( n \) objectives, see (23), where \( f_j(x) \) and \( X \) are \( j \)th objective and feasible region of the problem, respectively.

\[
\text{Min } f_i(x) = \sum_{a=1}^{d_a} d_a(x_a)
\]
\[
\text{Min } f_2(x) = \int_0^t rP(t)\,dt
\]
\[
\text{Min } f_3(x) = \int_0^t r^2P(t)\,dt - \left[ \int_0^t rP(t)\,dt \right]^2
\]
\[
\text{Max } f_4(x) = P_1(u)
\]
\[
s.t.: \quad P_i(t) = Q(x)P(t)
\]
\[
P_i(0) = 0 \quad \forall i = 1, 2, \ldots, n - 1
\]
\[
P_n(t) = 1
\]
\[
g_j(x_a) = \frac{1}{\mu_a} \quad \forall a \in A'
\]
\[
m_a\mu_a - \lambda \geq \varepsilon \quad \forall a \in A' - B
\]
\[
\lambda - 0.3m_a\mu_a \geq 0 \quad \forall a \in A' - B
\]
\[
\mu_a \geq \varepsilon \quad \forall a \in B
\]
\[
x_a \geq L_a \quad \forall a \in A'
\]
\[
x_a \leq U_a \quad \forall a \in A'
\]
\[
\sum_{a=1}^{d_a} x_a \leq J
\]

We divide the time interval into \( R \) equal portions with the length of \( \Delta t \). The corresponding discrete state model as follows

\[
\text{Min } f_1(x) = \sum_{a=1}^{d_a} d_a(x_a)
\]
\[
\text{Min } f_2(x) = \sum_{r=0}^{R-1} r\Delta t(P_r(r+1) - P_r(r))
\]
\[
\text{Min } f_3(x) = \sum_{r=0}^{R-1} (r\Delta t)^2(P_r(r+1) - P_r(r))
\]
\[
- \left[ \sum_{r=0}^{R-1} r\Delta t(P_r(r+1) - P_r(r)) \right]^2
\]
\[
\text{Max } f_4(x) = P_1\left( \left[ \frac{u}{\Delta t} \right] \right)
\]
\[
s.t.: \quad P(r+1) = P(r) + Q(x)P(r)\Delta t \quad r = 0, 1, 2, \ldots, R - 1
\]
\[
P_i(0) = 0 \quad \forall i = 1, 2, \ldots, n - 1
\]
\[
P_n(r) = 1 \quad r = 0, 1, \ldots, R
\]
\[
P_i(r) \leq 1 \quad i = 0, 1, \ldots, N - 1, \quad r = 1, 2, \ldots, R
\]
\[
g_j(x_a) = \frac{1}{\mu_a} \quad \forall a \in A'
\]
\[
m_a\mu_a - \lambda \geq \varepsilon \quad \forall a \in A' - B
\]
\[
\lambda - 0.3m_a\mu_a \geq 0 \quad \forall a \in A' - B
\]
\[
\mu_a \geq \varepsilon \quad \forall a \in B
\]
\[
x_a \geq L_a \quad \forall a \in A'
\]
\[
x_a \leq U_a \quad \forall a \in A'
\]
\[
\sum_{a=1}^{d_a} x_a \leq J
\]

Therefore, the appropriate goal attainment formulation of the multi-objective problem is given by

\[
\text{Min } z
\]
\[
s.t.: \quad f_j(x) - c_j z \leq b_j \quad j = 1, 2, \ldots, n
\]
\[x \in X\]

For solving the multi-objective models proposed in section 2.2 with the goal attainment method, the goals, \( b_j \)'s, and weights, \( c_j \)'s \( (j = 1, 2, 3, 4) \), should be determined for every objective, namely, the project direct costs, the mean of project completion time, the variance of project completion time and the probability that the project completion time does not exceed a certain threshold. Then, by applying (24) the appropriate goal attainment formulation of the multi-objective problem should be formed.

3. Reactive resource allocation in multi-server dynamic PERT networks

In the previous section, a multi-objective model for the resource allocation in multi-server dynamic PERT network was proposed. In the presented model, the uncertainty was considered in the entrance of projects and also in the duration of service stations, whereas, other uncertainty as project network disruption can occur. In this section, project network disruption such as: inserting a new activity (service station), deleting an activity and changes in precedence relations of project are considered. For coping with project network disruption, “reactive resource allocation”
is also suggested. Along with the project execution, a project may be disposed the considerable unforeseen disruptions, therefore, reactive scheduling (rescheduling), with revising or re-optimizing the baseline schedule, aims to repair the baseline schedule and consequently, overcome disruptions.

For overcoming these disruptions, firstly the revised PERT network is obtained by considering the changes in the project network. Then, by using section 2.1, the changed PERT network is transformed into a new PERT network with an exponentially distributed arc length. Finally, by adding the new objective, namely, minimizing the summation of cost of changes, and applying the models represented in sections 2.2.1 and 2.2.2, respectively, for resources as servers approach and resources affecting servers approach, the recovery model is constructed.

3.1. Resources as servers

Let \( m = \{m_a : a \in A\} \) and \( m' = \{m'_a : a \in A\} \) be the resource allocated to service stations in primary and reactive resource allocation, respectively. Also, let \( c = \{c_a : a \in A\} \) be the change cost of resource allocated per every unit in service stations. The main steps of our proposed method for the reactive resource allocation in resources as servers approach are as follows:

**Step 1.** Create revised PERT network by considering the required changes in the project network.

**Step 2.** Transform the changed PERT network into a new PERT network with exponentially distributed arc length by using section 2.1.

**Step 3.** Apply the model represented in section 2.2.1 for the network obtained in step 2, by adding a new objective as \( \min \sum_{a \in A'} \left| x'_a - x_a \right| c_a \), where \( A' \) represents the set of arcs of the network in AoA network, obtained in step 1.

3.2. Resources affecting servers

Let \( x = \{x_a : a \in A\} \) and \( x' = \{x'_a : a \in A\} \) be the resource allocated to service stations in primary and reactive resource allocation, respectively. Also, let \( c = \{c_a : a \in A\} \) be the change cost of resource allocated per every unit in service stations. The main steps of our suggested method for the reactive resource allocation in resources affecting servers approach are as follows:

**Step 1.** Create revised PERT network by considering the required changes in the project network.

**Step 2.** Transform the changed PERT network into a new PERT network with exponentially distributed arc length by using section 2.1.

**Step 3.** Apply the model represented in section 2.2.2 for the network obtained in step 2, by adding a new objective as \( \min \sum_{a \in A'} \left| x'_a - x_a \right| c_a \), where \( A' \) represents the set of arcs of the network in AoA network, obtained in step 1.

4. An illustrative case

To illustrate the analytical proposed method, we solve a numerical example to present the resource allocation in multi-server dynamic PERT networks, which is presented as the network of queue. It is assumed we have a system with the six service stations depicted as the AoN graph in Fig. 3. We want to determine the optimal resource allocation in multi-server dynamic PERT network for both approaches, namely, resources affecting servers approach and resources as servers approach. For solving this example, we also apply the goal attainment method.

**4.1. Resources affecting servers**

The assumptions of this example for the resources affecting servers approach are:

- The new projects, containing all their activities, arrived at the system according to a Poisson process with the rate of \( \lambda = 5 \) per year.
- The activity durations (service processing times) are independent random variables with exponential distributions.
- The threshold time, \( u \), that project completion time does not exceed is 3 years.
- The amount of resources available to be allocated to all service stations is 12.
- In all experiments, the value of \( \varepsilon \) is equal to 0.01.
Table 2. Characteristics of the activities

<table>
<thead>
<tr>
<th>Activity (a)</th>
<th>(d_a(x_a))</th>
<th>(g_a(x_a))</th>
<th>(m_a)</th>
<th>(I_a)</th>
<th>(U_a)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(2x_1 + 1)</td>
<td>0.6 - 0.08x_1</td>
<td>4</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>1.4x_2</td>
<td>0.15 - 0.01x_2</td>
<td>1</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>1.5x_3 + 2</td>
<td>0.16 - 0.01x_3</td>
<td>1</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>1.6x_4 + 1</td>
<td>0.45 - 0.05x_4</td>
<td>3</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>2x_5</td>
<td>0.8 - 0.09x_5</td>
<td>5</td>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>1.8x_6 + 2</td>
<td>0.7 - 0.07x_6</td>
<td>4</td>
<td>1</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 2 shows the characteristics of the activities, where the time unit and the cost unit are, respectively, in year and in thousand dollars.

Now, we substitute the nodes of 1, 4, 5, 6 in Fig. 3 with two series of exponential node, see Fig. 4, and then we determine the system states and transition rates, depicted in Table 3 and Fig. 5, where \(\gamma_a = \frac{(m_a\mu_a - \lambda)m_a\mu_a}{\lambda}\) for \(a = 1, 4, 5, 6\) and \(\gamma_a = \mu_a - \lambda\) for \(a = 2, 3\).

We organize the infinitesimal generator matrix \(Q(\mu)\) according to (7). Table 4 presents the infinitesimal generator matrix \(Q(\mu)\) (diagonal components are equal to minus sum of the other components at the same row).

Table 3. All admissible 2-partition cuts of the project

| 1. (1,2) | 5. (1,4) | 9. (1,4) | 13. (1,4) | 17. (5) |
| 2. (1,2) | 6. (1,4) | 10. (1,4) | 14. (3,4) | 18. (6) |
| 3. (1,4) | 7. (2,3) | 11. (3,4) | 15. (3,4) | 19. (6) |
| 4. (2,3) | 8. (3,4) | 12. (3,4) | 16. (5) | 20. (\phi, \phi) |

Table 4. Matrix \(Q(\mu)\)

<table>
<thead>
<tr>
<th>state</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(\gamma_2)</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>(\gamma_3)</td>
<td>(\gamma_4)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>(\gamma_1)</td>
<td></td>
</tr>
</tbody>
</table>

\(\gamma_1 = \frac{(m_a\mu_a - \lambda)m_a\mu_a}{\lambda}\) for \(a = 1, 4, 5, 6\) and \(\gamma_a = \mu_a - \lambda\) for \(a = 2, 3\).
The objective is to obtain the optimal resources allocated to the different activities by solving (19). For this purpose, we consider the goals, $25 = b_1$, $2.12 = b_2$, $25.03 = b_3$, $9.04 = b_4$ and the weight, $2.01 = c_1$, $1.02 = c_2$, $35.03 = c_3$, $35.04 = c_4$, for the four objectives, and also the various combinations of $R$ and $\Delta t$ for time interval 5 years. To do so, we employ LINGO 8 on a PC Pentium 4, CPU 3 GHz.

The optimal allocated resources, the computational times, CT (mm:ss), and also the values of all objectives for the different combinations of $R$ and $\Delta t$ are shown in Table 5. So, the optimal allocated resources are: $x_1 = 1.211$, $x_2 = 1.194$, $x_3 = 1.006$, $x_4 = 3.318$, $x_5 = 4.092$, and the objective function values are: $f_1 = 27.224$, $f_2 = 2.312$, $f_3 = 0.685$, $f_4 = 0.797$ ($z = 11.119$). Based on Table 5, if the length of $\Delta t$ is decreased, the accuracy of the solution is increased i.e. the value of $z$ is decreased and the computational time, CT, is also increased (for more details, see [25]). Note that the simulation results are: $x_1^{sim} = 1.326$, $x_2^{sim} = 1.124$, $x_3^{sim} = 1.238$, $x_4^{sim} = 1.097$, $x_5^{sim} = 3.413$, $x_6^{sim} = 3.736$, while the simulated mean of project completion time is 2.226.

### 4.2. Resources as servers

The assumptions of this example for the resources as servers approach are:

- The new projects, containing all their activities, arrived at the system according to a Poisson process with the rate of $\lambda = 10$ per year.
- The activity durations (service times) are independent random variables with exponential distributions.
- The amount of resource (server) available to be allocated to all service stations is 20.
- The threshold time, $u$, that project completion time does not exceed is 4 years.
- In all experiments, the value of $\epsilon$ is equal to 0.01.

Table 6 shows the characteristics of the activities, where the time unit and the cost unit are, respectively, in year and in thousand dollars.
Table 6. Characteristics of the activities

<table>
<thead>
<tr>
<th>Activity (a)</th>
<th>$d_a(m_a)$</th>
<th>$\mu_a$</th>
<th>$L_s$</th>
<th>$U_s$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$2m_1 + 1$</td>
<td>5</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>$1.4m_2$</td>
<td>3</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>$1.8m_3 + 4$</td>
<td>4</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td>4</td>
<td>$2.2m_4 + 3$</td>
<td>4.5</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td>5</td>
<td>$1.6m_5$</td>
<td>2.5</td>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td>6</td>
<td>$1.4m_6 + 2$</td>
<td>5.5</td>
<td>1</td>
<td>7</td>
</tr>
</tbody>
</table>

Now, we substitute every node in Fig. 3 with two series of exponential node, see Fig. 6, and as noted in section 4.1., we determine the system states, transition rates and the infinitesimal generator matrix $Q(\mu)$.

The objective is to obtain the optimal servers allocated to the different service stations by solving (18). For this purpose, we consider the goals, $b_1 = 43, b_2 = 1, b_3 = 0.5, b_4 = 0.9$ and the weight, $c_1 = 0.2, c_2 = 0.1, c_3 = 0.35, c_4 = 0.35$, for the four objectives, and also the various combinations of $R$ and $\Delta t$ for time interval 5 years. To do so, we employ LINGO 8 on a PC Pentium 4, CPU 3 GHz. The optimal allocated resources, the computational times, CT (mm:ss), and also the values of all objectives for the different combinations of $R$ and $\Delta t$ are shown in Table 7. So, the optimal allocated servers are: $m_1 = 3$, $m_2 = 4$, $m_3 = 3$, $m_4 = 3$, $m_5 = 5$, $m_6 = 2$, and the objective function values are: $f_1 = 44.4, f_2 = 2.568, f_3 = 1.116, f_4 = 0.854$ ($z = 15.682$).

Note that the simulation results are: $m_1 = 3$, $m_2 = 4$, $m_3 = 3$, $m_4 = 3$, $m_5 = 5$, $m_6 = 2$, while the simulated mean of project completion time is 2.438.

5. Conclusion

In this article, we proposed a multi-objective model to optimally control the resources allocated to the service stations in a multi-server dynamic PERT network for both approaches, namely resources as servers and resources affecting servers, using Markov process and multi objective programming. This dynamic PERT network was represented as a network of queues, where several servers are in each service station and the capacity of the system is infinite.

In both approaches, for modeling a multi-server dynamic PERT network, firstly the network of queues was transformed into a stochastic network and then, the states of the system were defined. Note that the number of system states grows combinatorially with the number of UDCs. Next, a system of differential equations was formed to solve and obtain the approximate completion time distribution for any particular project by creating an appropriate finite-state continuous-time Markov model. A multi-objective model with four conflicted objectives was presented to optimally control the resources allocated to service stations in a multi-server dynamic PERT network.

<table>
<thead>
<tr>
<th>R</th>
<th>$\Delta t$</th>
<th>$m_1$</th>
<th>$m_2$</th>
<th>$m_3$</th>
<th>$m_4$</th>
<th>$m_5$</th>
<th>$m_6$</th>
<th>$z$</th>
<th>$f_1$</th>
<th>$f_2$</th>
<th>$f_3$</th>
<th>$f_4$</th>
<th>$CT$</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>0.0625</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>16.118</td>
<td>44.4</td>
<td>2.612</td>
<td>1.007</td>
<td>0.862</td>
<td>00:03&quot;</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>0.05</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>16.016</td>
<td>44.4</td>
<td>2.602</td>
<td>1.034</td>
<td>0.86</td>
<td>00:03&quot;</td>
<td></td>
</tr>
<tr>
<td>125</td>
<td>0.04</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>15.934</td>
<td>44.4</td>
<td>2.593</td>
<td>1.055</td>
<td>0.858</td>
<td>00:03&quot;</td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>0.025</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>15.809</td>
<td>44.4</td>
<td>2.581</td>
<td>1.086</td>
<td>0.856</td>
<td>00:03&quot;</td>
<td></td>
</tr>
<tr>
<td>250</td>
<td>0.02</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>15.767</td>
<td>44.4</td>
<td>2.577</td>
<td>1.096</td>
<td>0.855</td>
<td>00:04&quot;</td>
<td></td>
</tr>
<tr>
<td>500</td>
<td>0.01</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>15.682</td>
<td>44.4</td>
<td>2.568</td>
<td>1.116</td>
<td>0.854</td>
<td>00:06&quot;</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 6. The substituted AON network of the project
In our model, the total project direct cost was considered as an objective to be minimized and the mean project completion time as another effective objective, should also be accounted to be minimized. The variance of the project completion time was another effective objective in the model, because the mean and the variance are two complementary concepts. The probability that the project completion time does not exceed a certain threshold was considered as the last objective. Finally, the goal attainment method was employed to solve a discrete-time approximation of the primary multi-objective problem.

For obtaining the best optimal allocated resource, we considered the various combinations of portions for a specific primary multi-objective problem. We can rewrite the distribution with parameters for a specific time interval. Based on the presented example, if the length of every portion, is decreased, the accuracy of the solution is increased i.e., the value of the objective is decreased and the computational time, CT, is also increased.

Appendix A:

If there is a server in the service station settled in the th node, then the queueing system is , and therefore, the density function of sojourn time (activity duration plus waiting time in queue) is calculated as follow [41]:

\[
w_a(t) = \left( \frac{\lambda - m_a \mu_a + \mu_a w_a^m(0)}{\lambda - (m_a - 1) \mu_a} \right) e^{-\mu_a t} + \left( 1 - \frac{\lambda - m_a \mu_a + \mu_a w_a^m(0)}{\lambda - (m_a - 1) \mu_a} \right) e^{-(m_a - \lambda - \mu_a) t} \quad t > 0
\]

Where and are, respectively, the arrival and service rate of new project and the service rate of service station . Also, , the probability of being zero queue length, and are obtained as follow:

\[
w_a^m(0) = 1 - \frac{m_a \lambda}{m_a \lambda - \mu_a} P_0
\]

and:

\[
P_0 = \left( \sum_{n=0}^{\infty} \frac{(\lambda / \mu_a)^n}{n!} + \frac{\lambda}{m_a \lambda - \mu_a} \left( \frac{m_a \lambda}{m_a \lambda - \mu_a} \right)^n \right)^{-1}
\]

As is observed, obtaining is very hard. We can rewrite the as follow, which is similar to two series of exponential distribution with parameters and :

\[
w_a(t) = (1 - w_a^m(0)) e^{-\mu_a t} + \left( 1 - \frac{w_a^m(0)}{m_a \mu_a - \lambda} \right) e^{-(m_a - \lambda - \mu_a) t}
\]

\[
0 < t < 0
\]

It seems that we can approximate density function of time spent in service station with two series exponential. Therefore, our approximate for density function of time spent in service station would be two series exponential with parameters and , where

\[
\rho_a = \frac{\lambda}{m_a \mu_a}.
\]

This approximation is quite simple and easy and there is no need to calculate and which is boring, especially when is large. Moreover, our approximation can be used in mathematic programming problem and Markov chain, conveniently. We want to evaluate the mean of sojourn time, and therefore, the expected number of projects in node as is observed, obtaining is boring, especially when is large. Moreover, our approximation can be used in mathematic programming problem and Markov chain, conveniently. We want to evaluate the mean of sojourn time, and therefore, the expected number of projects in node (A-4), and therefore, the expected number of projects in node (A-5) as calculated as follows:

\[
W = \frac{1}{\mu_a} \left( 1 + \frac{\rho_a^m m_a - 1}{m_a (1 - \rho_a)} \right)
\]

Also, Halfin and Whitt [43] developed a closed-form approximation for the expected sojourn time in work station as was calculated as follows:

\[
W = \frac{1}{\mu_a} \left( 1 + \frac{1}{m_a (1 - \rho_a)} \left( \frac{1}{1 + \sqrt{2 \pi \beta \Phi(\beta) e^{\beta / \beta}}} \right) \right)
\]

Where is the cumulative distribution function of standard normal distribution having mean 0 and variance 1. On the other hand, our approximation for the expected sojourn time in work station would be:
\[ W \approx \left( \frac{\rho_a}{m_a \mu_a} \right) + \left( \frac{m_a - 1}{m_a \mu_a} \right) \]  \quad (A-7)

Table A-1. Numerical value of exact and approximation expected number of projects in work station

<table>
<thead>
<tr>
<th>( m_a )</th>
<th>( \rho )</th>
<th>( L_a )</th>
<th>Our app.</th>
<th>Error(%)</th>
<th>( m_a )</th>
<th>( \rho )</th>
<th>( L_a )</th>
<th>Our app.</th>
<th>Error(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>0.30</td>
<td>0.21</td>
<td>29.73</td>
<td></td>
<td>0.1</td>
<td>1.50</td>
<td>1.41</td>
<td>5.93</td>
<td></td>
</tr>
<tr>
<td>0.3</td>
<td>0.93</td>
<td>0.73</td>
<td>21.66</td>
<td></td>
<td>0.3</td>
<td>4.50</td>
<td>4.33</td>
<td>3.81</td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>1.74</td>
<td>1.50</td>
<td>13.63</td>
<td></td>
<td>0.5</td>
<td>7.51</td>
<td>7.50</td>
<td>0.15</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.7</td>
<td>3.25</td>
<td>3.03</td>
<td>6.63</td>
<td>15</td>
<td>0.7</td>
<td>10.83</td>
<td>11.43</td>
<td>5.58</td>
</tr>
<tr>
<td>0.8</td>
<td>4.99</td>
<td>4.80</td>
<td>3.78</td>
<td></td>
<td>0.8</td>
<td>13.28</td>
<td>14.40</td>
<td>8.46</td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>10.05</td>
<td>9.90</td>
<td>1.53</td>
<td></td>
<td>0.9</td>
<td>18.92</td>
<td>20.70</td>
<td>9.39</td>
<td></td>
</tr>
<tr>
<td>0.98</td>
<td>50.10</td>
<td>49.98</td>
<td>0.24</td>
<td></td>
<td>0.98</td>
<td>59.36</td>
<td>61.74</td>
<td>4.02</td>
<td></td>
</tr>
<tr>
<td>0.1</td>
<td>0.50</td>
<td>0.41</td>
<td>17.78</td>
<td></td>
<td>0.1</td>
<td>2.00</td>
<td>1.91</td>
<td>4.45</td>
<td></td>
</tr>
<tr>
<td>0.3</td>
<td>1.51</td>
<td>1.33</td>
<td>11.94</td>
<td></td>
<td>0.3</td>
<td>6.00</td>
<td>5.83</td>
<td>2.86</td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>2.63</td>
<td>2.50</td>
<td>4.96</td>
<td></td>
<td>0.5</td>
<td>10.00</td>
<td>10.00</td>
<td>0.04</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.7</td>
<td>4.38</td>
<td>4.43</td>
<td>1.18</td>
<td>20</td>
<td>0.7</td>
<td>14.22</td>
<td>14.93</td>
<td>5.03</td>
</tr>
<tr>
<td>0.9</td>
<td>11.36</td>
<td>11.70</td>
<td>2.97</td>
<td></td>
<td>0.9</td>
<td>22.96</td>
<td>25.20</td>
<td>9.77</td>
<td></td>
</tr>
<tr>
<td>0.98</td>
<td>51.47</td>
<td>51.94</td>
<td>0.92</td>
<td></td>
<td>0.98</td>
<td>63.57</td>
<td>66.64</td>
<td>4.83</td>
<td></td>
</tr>
<tr>
<td>0.1</td>
<td>1.00</td>
<td>0.91</td>
<td>8.89</td>
<td></td>
<td>0.1</td>
<td>2.50</td>
<td>2.41</td>
<td>3.56</td>
<td></td>
</tr>
<tr>
<td>0.3</td>
<td>3.00</td>
<td>2.83</td>
<td>5.73</td>
<td></td>
<td>0.3</td>
<td>7.50</td>
<td>7.33</td>
<td>2.29</td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>5.04</td>
<td>5.00</td>
<td>0.72</td>
<td></td>
<td>0.5</td>
<td>12.50</td>
<td>12.50</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0.7</td>
<td>7.52</td>
<td>7.93</td>
<td>5.53</td>
<td>25</td>
<td>0.7</td>
<td>17.65</td>
<td>18.43</td>
<td>4.45</td>
</tr>
<tr>
<td>0.8</td>
<td>9.64</td>
<td>10.40</td>
<td>7.92</td>
<td></td>
<td>0.8</td>
<td>20.84</td>
<td>22.40</td>
<td>7.5</td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>15.02</td>
<td>16.20</td>
<td>7.87</td>
<td></td>
<td>0.9</td>
<td>27.07</td>
<td>29.70</td>
<td>9.71</td>
<td></td>
</tr>
<tr>
<td>0.98</td>
<td>55.28</td>
<td>56.84</td>
<td>2.82</td>
<td></td>
<td>0.98</td>
<td>67.87</td>
<td>71.54</td>
<td>5.41</td>
<td></td>
</tr>
</tbody>
</table>

Fig. A-1. The exact and our approximation of expected number of projects

Obviously, our approximation is very simple and its components easily, without boring computations, are controllable, i.e., the complexity of our approximation is less than other approximations. Furthermore, as mentioned before, our approximation can be used in mathematic programming problem and Markov chain, conveniently.

In Table A-1, the expected number of projects in workstation \( a \) \( (L_a) \) and the error of our approximation are represented.

As \( m_a \leq 5 \) and \( \rho_a \leq 0.3 \), our approximation for expected number of projects in work stations is very poor, therefore, for coping with this shortage, we consider \( 0.3 \leq \rho_a < 1 \). Also, in Fig. A-2, the exact cumulative distribution function and our approximate cumulative distribution function for \( m_a = 5 \) and \( m_a = 10 \) with various utilization factors, is shown.

Moreover, in Table A-2, the maximum difference (MD) between the exact cumulative distribution function and our approximate cumulative distribution function for various number of server and utilization factors, is shown.

Consequently, we have:

1. If \( m_a = 1 \), then the queueing system would be an \( M / M / 1 \) queue, and the density function of time spent in the service station \( a \), \( w_a(t) \), would be exponentially with parameter \( \mu_a - \lambda \), therefore, \( w_a(t) \) is calculated as follows:

\[ w_a(t) = (\mu_a - \lambda) e^{-\mu_a t} \quad t > 0 \]  \quad (A-8)

2. If \( m_a = \infty \), then the queueing system is \( M / M / \infty \), and the density function of time spent in the service station \( a \) would be exponentially expressed with parameter \( \mu_a \), therefore, \( w_a(t) \) is calculated as follows:
\[ w_a(t) = \mu_a e^{-\rho_a t} \quad t > 0, \text{ if } m_a = \infty \quad (A-9) \]

Table A-2. Maximum difference (MD) between the exact and our approximate distribution function

<table>
<thead>
<tr>
<th>( m_a )</th>
<th>( \rho_a )</th>
<th>MD</th>
<th>( m_a )</th>
<th>( \rho_a )</th>
<th>MD</th>
<th>( m_a )</th>
<th>( \rho_a )</th>
<th>MD</th>
<th>( m_a )</th>
<th>( \rho_a )</th>
<th>MD</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>0.132</td>
<td></td>
<td>0.3</td>
<td>0.034</td>
<td></td>
<td>0.1</td>
<td>0.016</td>
<td></td>
<td>0.1</td>
<td>0.011</td>
<td></td>
</tr>
<tr>
<td>0.3</td>
<td>0.105</td>
<td></td>
<td>0.3</td>
<td>0.030</td>
<td></td>
<td>0.3</td>
<td>0.017</td>
<td></td>
<td>0.3</td>
<td>0.012</td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td>0.082</td>
<td></td>
<td>0.5</td>
<td>0.061</td>
<td></td>
<td>0.5</td>
<td>0.038</td>
<td></td>
<td>0.5</td>
<td>0.027</td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td>0.078</td>
<td>10</td>
<td>0.7</td>
<td>0.093</td>
<td>20</td>
<td>0.7</td>
<td>0.070</td>
<td>30</td>
<td>0.7</td>
<td>0.055</td>
<td></td>
</tr>
<tr>
<td>0.8</td>
<td>0.079</td>
<td></td>
<td>0.9</td>
<td>0.087</td>
<td></td>
<td>0.9</td>
<td>0.088</td>
<td></td>
<td>0.9</td>
<td>0.083</td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td>0.055</td>
<td></td>
<td>0.97</td>
<td>0.035</td>
<td></td>
<td>0.98</td>
<td>0.045</td>
<td></td>
<td>0.98</td>
<td>0.049</td>
<td></td>
</tr>
</tbody>
</table>

3. If \( 1 < m_a < \infty \), then the queueing system is \( M/M/m_a \), and the density function of time spent in the service station \( a \) would be approximately two series exponential with parameters \( (m_a \mu_a - \lambda)/\rho_a \) and \( (m_a \mu_a - \lambda)/m_a - 1 \). Therefore, \( w_a(t) \) is approximately calculated as follows:

\[ w_a(t) = \left( \frac{m_a \mu_a - \lambda}{m_a - 1} \right) \left( \frac{m_a \mu_a - \lambda}{\rho_a} \right) e^{-\left( \frac{m_a \mu_a - \lambda}{m_a - 1} \right) t} \]

References


\[ L_p = \frac{\alpha \beta^p}{1 - \rho} \]  
