A heuristic method for consumable resource allocation in multi-class dynamic PERT networks

Saeed Yaghoubi[1*]
[*] Corresponding author
Email: yaghoubi@iust.ac.ir

Siamak Noori[1]
Email: snoori@iust.ac.ir

Mohammad Mahdavi Mazdeh[1]
Email: mazdeh@iust.ac.ir

[1] Department of Industrial Engineering, Iran University of Science and Technology, Tehran, Iran

Abstract

This investigation presents a heuristic method for consumable resource allocation problem in multi-class dynamic Project Evaluation and Review Technique (PERT) networks, where new projects from different classes (types) arrive to system according to independent Poisson processes with different arrival rates. Each activity of any project is operated at a devoted service station located in a node of the network with exponential distribution according to its class. Indeed, each project arrives to the first service station and continues its routing according to precedence network of its class. Such system can be represented as a queuing network, while the discipline of queues is first come, first served. On the basis of presented method, a multi-class system is decomposed into several single-class dynamic PERT networks, whereas each class is considered separately as a minisystem. In modeling of single-class dynamic PERT network, we use Markov process and a multi-objective model investigated by Azaron and Tavakkoli-Moghaddam in 2007. Then, after obtaining the resources allocated to service stations in every minisystem, the final resources allocated to activities are calculated by the proposed method.

Keywords

Project management, Multi-class dynamic PERT network, Queuing

Introduction

Today, multi-project scheduling which considers all projects of organization as one system and therefore sharing limited resources among multiple projects is mostly noticed, whereas up to 90% of organizations handle the projects in a multi-project environment (Payne 1995). On the other hand, for better management, in some organizations, the project-oriented approach is primarily adopted and the operations of the organization are dependently executed on projects.

Archive of SID

www.SID.ir

In the literature of multi-project scheduling, *multi-project resource constrained scheduling problem* (MPRCSP) in static and deterministic conditions is a major topic of most researches. Generally, two approaches for analyzing MPRCSP exist. One approach by linking all projects of organization synthetically together into a large single project, analyzes MPRCSP, while the other approach by considering the projects as independent component and also resources constraints, proposes an objective function that contains all projects (possibly appropriately weighted) for analyzing MPRCSP.

Wiest (1967) and Pritsker et al. (1969) were the first pioneering researchers in studying MPRCSP who presented, respectively, a zero–one programming approach and a heuristic model for analyzing this problem. Then Kurtulus and Davis (1982) and Kurtulus and Narula (1985), by employing the priority rules and describing measures, studied the MPRCSP. Moreover, some investigations have been concentrated on MPRCSP by applying multi-objective and multi-criteria approaches; for example, Chen (1994) proposed the zero–one goal programming model in MPRCSP for the maintenance of mineral processing, and Lova et al. (2000) analyzed MPRCSP with applying a lexicographically two criteria. Also recently, Kanagasabapathi et al. (2009) by defining performance measures and considering maximum tardiness and mean tardiness of projects studied the MPRCSP in a static environment.

On the other hand, some researchers such as Tsubakitani and Deckro (1990), Lova and Tormos (2001), Kumanan et al. (2006), Goncalves et al. (2008), Ying et al. (2009), and Chen and Shahandashti (2009) proposed heuristic and metaheuristic algorithms for solving MPRCSP. Kruger and Scholl (2008) extended the MPRCSP by considering transfer times and its cost.

In all of the mentioned investigations, MPRCSP has been analyzed in static and deterministic environment, while a few studies have been concentrated on multi-project scheduling under uncertainty conditions. Fatemi-Ghomi and Ashjari (2002) by considering a multi-channel queuing introduced a simulation model for multi-project resource allocation with stochastic activity durations. Nozick et al. (2004) also presented a nonlinear mixed-integer programming model to optimally allocate the resources, while the probability distribution of activity duration and allocated resources depend on together. Moreover, event-driven approach and Critical Chain Project Management (CCPM) approach were proposed for overcoming uncertainty in multi-project system by Kao et al. (2006) and Byali and Kannan (2008), respectively.

Normally, greater resource allocation will incur a greater expense; therefore, we have a trade-off between the time to complete an activity and its cost (a function of the assigned resource). When the task duration is a linear function of the allocated resource, we have the linear time–cost trade-off problem which was analyzed by Fulkerson (1961). The nonlinear case of the time–cost trade-off problem was studied by Elmaghraby and Salem (1982, 1984) and Elmaghraby (1993, 1996). Moreover, Elmaghraby and Morgan (2007) presented a fast method for optimizing resource allocation in a single-stage stochastic activity network.

Clearly, multi-project scheduling is more elaborate than single-project scheduling. On the other hand, in some organizations, not only the task durations are uncertain, but also new projects dynamically enter to the organization over the time horizon. In such conditions, the organization is faced with a multi-project system in which the scheduling procedure would be more elaborate than before. Adler et al. (1995) studied such multi-project system by considering the organization as a ‘stochastic processing network’ and using simulation. They

Archive of SID

www.SID.ir

assumed that the organization is comprised of a collection of 'service stations' (work stations) or 'resources', where several types (classes) of projects exist in a system and one or more identical parallel servers can be settled in each service station. Therefore, such multi-project system, denominated as *Dynamic PERT Network*, can be considered as a queuing network and is attractive for organizations having the same projects, for example, maintenance projects.

Anavi-Isakow and Golany (2003) by applying simulation study used the concept of CONWIP (constant work-in-process) in multi-class dynamic Project Evaluation and Review Technique (PERT) networks. They described two control mechanisms, constant number of projects in process (CONPIP) and constant time of projects in process (CONTIP), namely, CONPIP mechanism restricts the number of projects, while CONTIP mechanism limits the total processing time by all the projects that are active in the system.

In addition, Cohen et al. (2005, 2007) by using cross entropy, based on simulation, investigated the resource allocation problem in multi-class dynamic PERT networks, where the resources can work in parallel. Indeed, the number of resources allocated and servers in each service station are equal (e.g., mechanical work station with mechanics, electrical work station with electricians), and the total number of resources in the system also is constant.

On the other hand, Azaron and Tavakkoli-Moghaddam (2007) proposed an analytical multi-objective model to optimally control consumable resources allocated to the activities in a dynamic PERT network, where only one type of project exists in the system and the number of servers in every service station is either 1 or infinity. They assumed that the activity durations are exponentially distributed random variables, resources allocated affect the mean of service times, and the new projects are generated according to a Poisson process. A risk element was considered in a dynamic PERT network by Li and Wang (2009) who, by using general project risk element transmission theory, proposed a multi-objective risk-time–cost trade-off problem. Recently, Azaron et al. (2011) introduced an algorithm in computing optimal constant lead time for each particular project in a repetitive (dynamic) PERT network by minimizing the average aggregate cost per project.

In practice, due to various projects' requirements, most organizations execute the projects from several classes, where new projects from different classes arrive to the system dynamically over the time horizon and service stations that stochastically serve to projects. In such conditions, organization is encountered with multi-class dynamic PERT networks, while projects from different classes differ in their precedence networks, the mean of their service time in every service station, and also their arrival rates.

Reviewing the above-mentioned investigations indicated that the multi-class dynamic PERT network has been studied only using simulation. As the main contribution of this paper, we present a heuristic method for the consumable resource allocation problem (or time–cost trade-off problem) in multi-class dynamic PERT networks. Note that the resource allocation problem in multi-class dynamic PERT networks is a generalization of the resource allocation problem in dynamic PERT network, investigated by Azaron and Tavakkoli-Moghaddam (2007) by considering only one type (class) of projects in multi-project system.

In this paper, it is assumed that the new projects from different classes, including all their activities, are entered to the system according to independent Poisson processes with different arrival rates and each activity of any project is executed at a devoted service station settled in

Archive of SID

www.SID.ir

a node of the network based on first come, first served (FCFS) discipline. Moreover, the number of servers in every service station is either 1 or infinity, while the service times (activity durations) are independent random variables with exponential distributions. Indeed, each project arrives to the first service station and continues its routing according to precedence network of its class.

For determining the resources allocated to the service stations, the multi-class system is decomposed into several single-class dynamic PERT networks, whereas each class is considered separately as a minisystem. For modeling of a single-class dynamic PERT network, we use Markov process and a multi-objective model, investigated by Azaron and Tavakkoli-Moghaddam (2007), namely, we first convert the network of queues into a stochastic network. Then, by constructing a proper finite-state absorbing continuous-time Markov model, a system of differential equations is created. Moreover, we apply a multi-objective model containing four conflicted objectives to optimally control the consumable resources allocated to service stations in the single-class dynamic PERT network and further employ the goal attainment method to solve a discrete-time approximation of the primary multi-objective problem. Finally, after obtaining the resources allocated to service stations in every minisystem, the final resources allocated to servers are calculated by the proposed method.

This paper is composed of five sections. The remainder is organized as follows. In 'Single class dynamic PERT networks,' we model the single-class dynamic PERT network by employing a finite-state continuous-time Markov process and apply a multi-objective model to optimally control the consumable resources allocated to service stations in a single-class dynamic PERT network. In 'Consumable resource allocation in multi-class dynamic PERT networks,' a heuristic model is proposed for consumable resources allocated in the multi-class dynamic PERT networks. We solve an example in 'An illustrative case' section and conclusion is given in 'Conclusions.'

Single-class dynamic PERT networks

In this section, we represent a multi-objective model to optimally control the consumable resources allocated to the service stations investigated by Azaron and Tavakkoli-Moghaddam (2007). For this purpose, we first explain an analytical method to compute the distribution function of project completion time and then we describe a multi-objective model in single-class dynamic PERT network.

It is assumed that a project is represented as an activity-on-node (AoN) graph, and also the new projects, including all their activities, are arrived to the system according to a Poisson process with the rate of λ. Moreover, each activity of a project is executed in a devoted service station settled in a node of the network based on FCFS discipline.

Such system can be considered as a network of queues, where the service times represent the durations of the corresponding activities. It is also assumed that the number of servers in each service station to be either one or infinity, while the service times (activity durations) are independent random variables with exponential distributions.

Archive of SID

www.SID.ir

Project completion time distribution

For presenting an analytical method to compute the distribution function for single-class dynamic PERT network, the method of Kulkarni and Adlakha (1986) will be used because this method presents an analytical, simple, and easy approach to implement on a computer and a computationally stable algorithm to evaluate the distribution function of the project completion time. This method models PERT network with independent and exponentially distributed activity durations by continuous-time Markov chains with upper triangular generator matrices. The special structure of the chain allows us to develop very simple algorithms for the exact analysis of the network.

It is assumed that the service time in service station *a* is exponentially distributed with the rate of μ_a, where the arrival stream of projects to each service station is according to a Poisson process with the rate of λ. Moreover, the number of server in node *a* is either one or infinity and therefore the node *a* treats as an *M*/*M*/1 or *M*/*M*/∞ model. The main steps of method to determine a finite-state absorbing continuous-time Markov process to computing the distribution function in single-class dynamic PERT network are as follows:

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

Step 1.1 If there is one server in service station *a*, then the queuing system would be *M*/*M*/1, and the density function of time spent in the service station *a* would be exponentially expressed with parameter $\mu_a - \lambda_a \quad (\mu_a \succ \lambda)_a$; therefore, $w_a(t)$ is calculated as follows:

$$w_a(t) = (\mu_a - \lambda).e^{-(\mu_a - \lambda)t} \quad t \succ 0 \qquad (1)$$

Step 1.2 If there are infinite servers in service station *a*, then the queuing system would be *M*/*M*/∞, and the density function of time spent in the service station *a* would be exponentially expressed with parameter μ_a; therefore, $w_a(t)$ is calculated as follows:

$$w_a(t) = \mu_a.e^{-\mu_a t} \quad t \succ 0 \qquad (2)$$

Step 2 Transform the single-class dynamic PERT network, represented as an AoN graph, to a classic PERT network represented as an activity-on-arc (AoA) graph.

When considering AoN graph, substitute each node with a stochastic arc (activity) whose length is equal to the sojourn time in the corresponding queuing system. For this purpose, node *a* in the AoN graph is replaced with a stochastic activity. Assume that $b_1,b_2,\ldots,b_n$ are the incoming arcs to node *a* and $d_1,d_2,\ldots,d_m$ are the outgoing arcs from it in the queuing network. Then, node *a* is substituted by activity (*v*,*w*), whose length is equal to the activity duration *a*. Furthermore, all arcs $b_1,b_2,\ldots,b_n$ terminate to node *v* while all arcs $d_1,d_2,\ldots,d_m$ originate from node *w* [for more details, see Azaron and Modarres (2005)].

Step 3 Calculate the distribution function of the project completion time (longest path time) in the AoA obtained in step 2, while activity durations are distributed exponentially. For the computation of longest path time distribution in AoA, the method of Kulkarni and Adlakha (1986) is applied.

Archive of SID

www.SID.ir

Let $G = (V, A)$ be the PERT network, obtained in step 2, 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 stochastic network (AoA). Let s and t be the source and sink nodes, respectively, and length of arc $a \in A$ be a random variable that exponentially distributed with parameter γ_a. For $a \in A$, the starting node and the ending node of arc a are denoted as $\alpha(a)$ and $\beta(a)$, respectively.

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, which are defined as follows:

$$I(v) = \{a \in A : \beta(a) = v\} \quad (v \in V), \tag{3}$$

$$O(v) = \{a \in A : \alpha(a) = v\} \quad (v \in V). \tag{4}$$

Definition 2. For $X \subset V$ such that $s \in X$ and $t \in \bar{X} = V - X$, an (s,t) cut is defined as follows:

$$(X, \bar{X}) = \left\{ a \in A : \alpha(a) \in X, \beta(a) \in \bar{X} \right\}. \tag{5}$$

An (s,t) cut $(X, \bar{X})$ is denominated a uniformly directed cut (UDC), if $(\bar{X}, X) = \phi$, i.e., there are no two arcs in the cut belonging 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 that arc belongs to $\bar{X}$.

Example 1. Consider the stochastic network shown in Figure 1 taken from Azaron and Tavakkoli-Moghaddam (2007). According to the definition, the UDCs of this network are (1,2); (2,3); (1,4,6); (3,4,6); and (5,6).

Figure 1 Example 1.

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

Again consider Example 1 that (1,4,6) is one of the UDCs. For example, this cut can be decomposed into $E = \{1, 6\}$ and $F = \{4\}$. In this case, the cut is an admissible 2-partition because $I(\beta(4)) = \{3, 4\} \not\subset F$. Furthermore, if $E = \{6\}$ and $F = \{1, 4\}$, then the cut is not an admissible 2-partition because $I(\beta(1)) = \{1\} \subset F = \{1, 4\}$

Definition 4. Along the projects execution at time t, each activity 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.

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

$$Y(t) = \{a \in A,\ a \text{ is active at time } t\}, \tag{6}$$

$$Z(t) = \{a \in A,\ a \text{ is dormant at time } t\}, \tag{7}$$

Archive of SID

www.SID.ir

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

A UDC is divided into E and F that contain active and dormant activities, respectively. All admissible 2-partition cuts of network of Figure 1 are presented in Table 1.

Table 1 All admissible 2-partition cuts of the example network

Cut	Value
1	(1,2)
2	(2,3)
3	(2,3^a)
4	(1,4,6)
5	(1,4^a,6)
6	(1,4,6^a)
7	(1,4^a,6^a)
8	(3,4,6)
9	(3^a,4,6)
10	(3,4^a,6)
11	(3,4,6^a)
12	(3^a,4,6^a)
13	(3,4,6)
14	(5,6)
15	(5^a,6)
16	(5,6^a)
17	(ø,ø)

A superscripted roman letter 'a' is applied to denote a dormant activity and all others are active.

The set of all admissible 2-partition cuts of the network is defined as S and also $\overline{S} = S \cup \{(\phi, \phi)\}$. Note that $X(t) = (ø,ø)$ 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 details, see Kulkarni and Adlakha (1986)).

If activity a terminates with the rate of γ_a, and $I(\beta(a)) \not\subset F \cup \{a\}$, namely, there is at least one unfinished activity in $I(\beta(a))$, then $E' = E - \{a\}, F' = F \cup \{a\}$. Furthermore, if by terminating activity a, all activities in $I(\beta(a))$ become idle $(I(\beta(a)) \subset F \cup \{a\})$, then $E' = (E - \{a\}) \cup O(\beta(a))$, $F' = F - I(\beta(a))$. Namely, all activities in I(β(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 denoted by $Q = [q\{(E,F),(E',F')\}]$, where (E,F) and $(E',F') \in \overline{S}$ are obtained as follows:

Archive of SID

www.SID.ir

$$q\{(E,F),(E',F')\}=\begin{cases}\gamma_a & if: a\in E, I(\beta(a))\not\subset F\cup\{a\}, E'=E-\{a\}, F'=F\cup\{a\}\\ \gamma_a & if: a\in E, I(\beta(a))\subset F\cup\{a\}, E'=(E-\{a\})\cup O(\beta(a)), F'=F-I(\beta(a))\\ -\sum_{a\in E}\gamma_a & if: E'=E, F'=F\\ 0 & \text{otherwise.}\end{cases} \quad (8)$$

A continuous-time Markov process, $\{X(t),\ t \geq 0\}$, is with finite-state space $\overline{S}$ and since $q\{(ø,ø),(ø,ø)\} = 0$, the project is completed. In this Markov process, all of states except $X(t) = (ø,ø)$ that is the absorbing state are transient. Furthermore, the states in $\overline{S}$ should be numbered such that this Q matrix be an upper triangular one. It is assumed that the states are numbered as $1,2,...,N=|\overline{S}|$ such that $X(t) = (O(s), ø)$ and $X(t) = (ø,ø)$ 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 PERT network, obtained in step 2. Obviously, $T=\min\{t\geq 0: X(t)=N \,|\, X(0)=1\}$.

Chapman-Kolmogorov backward equations can be used to calculate $F(t)=P(T\leq t)$. If it is defined

$$P_i(t)=P(X(t)=N\,|\,X(0)=i) \quad i=1,2,...,N, \qquad (9)$$

then $F(t)=P(T\leq t)$.

The system of linear differential equations for the vector $P(t)=[P_1(t) \quad P_2(t) \quad .... \quad P_N(t)]^T$ is presented by

$$P'(t)=\frac{dP(t)}{dt}=Q.P(t)$$
$$P(0)=[0 \quad 0 \quad .... \quad 1]^T, \qquad (10)$$

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.

Multi-objective consumable resource allocation

In this section, we apply a multi-objective model to optimally control the consumable resources allocated to the service stations in a single-class dynamic PERT network, representing as a network of queues, where we allocate more resources to service station and where the mean service time will be decreased and the direct cost of service station per period will be increased. This means the mean service time in each service station is a non-increasing function and the direct cost of each service station per period is a non-decreasing function of the amount of resource allocated to it, i.e., the total direct costs of service station per period and the mean project completion time are dependent together and an appropriate trade-off between them is required. Also, the variance of the project completion time should

Archive of SID

www.SID.ir

be accounted in the model because the mean and the variance are two complementary concepts. The last objective that should be considered is the probability that the project completion time does not exceed a certain threshold for on-time delivery performance.

Let x_a be the resource allocated in service station a ($a \in A$); also, $d_a(x_a)$ represents the direct cost of service station a ($a \in A$) per period in the PERT network, obtained in 'Project completion time distribution' section step 2, 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 $\text{PDC} = \sum_{a \in A} d_a(x_a)$. Also, the mean service time in the service station a, $g_a(x_a)$, is assumed to be a non-increasing function of the amount of resource x_a allocated to it that would be equal to

$$g_a(x_a) = \frac{1}{\mu_a} \quad (\forall a \in A) \tag{11}$$

Let U_a be the maximum amount of resource available to be allocated to the activity a ($a \in A$), L_a be the minimum amount of resource needed to execute the activity a, $x = [x_a : a \in A]^T$, and J be the amount that represents the resource available to be allocated to all of activities. Moreover, we define u as a threshold time that project completion time does not exceed it. In practice, $d_a(x_a)$ and $g_a(x_a)$ can be obtained by employing linear regression based on the previous similar activities or applying the judgments of experts in this area.

Therefore, the objective functions are given by the following:

1 Minimizing the project direct cost per period

$$\text{Min } f_1(x) = \sum_{a \in A} d_a(x_a) \tag{12}$$

2 Minimizing the mean of project completion time ($P'_1(t)$ is the density function of project completion time)

$$\text{Min } f_2(x) = E(T) = \int_0^\infty (1 - P_1(t))\,dt = \int_0^\infty t P_1'(t))\,dt \tag{13}$$

3 Minimizing the variance of project completion time

$$\text{Min } f_3(x) = \text{Var}(T) = \int_0^\infty t^2 P_1'(t))\,dt - \left[\int_0^\infty t P_1'(t))\,dt\right]^2 \tag{14}$$

4 Maximizing the probability that the project completion time does not exceed a certain threshold

$$\text{Max } f_4(x) = P_1(u) = P(T \le u) \tag{15}$$

Archive of SID

www.SID.ir

The infinitesimal generator matrix Q would be a function of the control vector $\mu = [\mu_a : a \in A]^T$ ($x = [x_a : a \in A]^T$). Therefore, the nonlinear dynamic model is

$$\begin{aligned} &P'(t) = Q(\mu).P(t) \\ &P_i(0) = 0 \quad \forall i = 1,2,...,N-1 \\ &P_N(t) = 1 \end{aligned} \tag{16}$$

Let B and C be the set of arcs in the PERT network, obtained in 'Project completion time distribution' section step 2 that there are, respectively, one server and infinite servers settled on the corresponding service stations ($A = B \cup C$). The next constraint should be satisfied to show the response in the steady-state:

$$\begin{aligned} &\mu_a \succ \lambda \quad \forall a \in B \\ &\mu_a \succ 0 \quad \forall a \in C \end{aligned} \tag{17}$$

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

$$\begin{aligned} &\mu_a \geq \lambda + \varepsilon \quad \forall a \in B \\ &\mu_a \geq \varepsilon \quad \forall a \in C \end{aligned} \tag{18}$$

Consequently, the appropriate multi-objective optimal control problem is

$$\begin{aligned} &\text{Min } f_1(x) = \sum_{a \in A} d_a(x_a) \\ &\text{Min } f_2(x) = \int_0^\infty tP_1'(t))\,dt \\ &\text{Min } f_3(x) = \int_0^\infty t^2 P_1'(t))\,dt - \left[\int_0^\infty tP_1'(t))\,dt\right]^2 \\ &\text{Max } f_4(x) = P_1(u) \\ &s.t: P'(t) = Q(\mu).P(t) \\ &\quad P_i(0) = 0 \quad \forall i = 1,2,...,N-1 \\ &\quad P_N(t) = 1 \\ &\quad g_a(x_a) = \frac{1}{\mu_a} \quad \forall a \in A \\ &\quad \mu_a \geq \lambda + \varepsilon \quad \forall a \in B \\ &\quad \mu_a \geq \varepsilon \quad \forall a \in C \\ &\quad x_a \geq L_a \quad \forall a \in A \\ &\quad x_a \leq U_a \quad \forall a \in A \\ &\quad \sum_{a \in A} x_a \leq J. \end{aligned} \tag{19}$$

This stochastic programming is impossible to solve, therefore, based on the definition of integral thinking, we divide time interval into R equal portions with the length of Δt. Indeed,

Archive of SID

www.SID.ir

we transform the differential equations into difference equations. Thus, the corresponding discrete state model can be given as follows (for more details see Azaron and Tavakkoli-Moghaddam 2007):

$$\text{Min } f_2(x) = \sum_{r=0}^{R-1} r\Delta t\left(P_1(r+1) - P_1(r)\right)$$

$$\text{Min } f_3(x) = \sum_{r=0}^{R-1} (r\Delta t)^2\left(P_1(r+1) - P_1(r)\right) - \left[\sum_{r=0}^{R-1} r\Delta t\left(P_1(r+1) - P_1(r)\right)\right]^2$$

$$\text{Max } f_4(x) = P_1\left(\left[\frac{u}{\Delta t}\right]\right)$$

$$s.t: P(r+1) = P(r) + Q(\mu)P(r)\Delta t \quad r = 0,1,2,...,R-1$$

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

$$P_N(r) = 1 \quad r = 0,1,...,R$$

$$P_i(r) \le 1 \quad i = 0,1,...,N-1, \quad r = 1,2,...,R$$

$$g_a(x_a) = \frac{1}{\mu_a} \quad \forall a \in A$$

$$\mu_a \ge \lambda + \varepsilon \quad \forall a \in B$$

$$\mu_a \ge \varepsilon \quad \forall a \in C$$

$$x_a \ge L_a \quad \forall a \in A$$

$$x_a \le U_a \quad \forall a \in A$$

$$\sum_{a \in A} x_a \le J. \tag{20}$$

Goal attainment method

We now need to use a multi-objective method to solve (20), and we actually use goal attainment technique for this purpose. The goal attainment method needs to determine a goal, b_j, and a weight, c_j, for every objective, namely, the total project direct cost per period, the mean project completion time, the variance of project completion time, and the probability that the project completion time does not exceed a certain threshold. c_js represent the importance of objectives, whereas, if an objective has the smallest c_j, then it will be the most important objective. c_js (j = 1,2,3) are commonly normalized such that $\sum_{j=1}^{3} c_j = 1$. Goal attainment method is actually a variation of goal programming method intending to minimize the maximum weighted deviation from the goals.

Since goal attainment method has a fewer variables to work with, compared to other simple and interactive multi-objective methods, it will be computationally faster and more suitable to solve our complex optimization problem. Therefore, the appropriate goal attainment formulation of the resource allocation problem is given by

Archive of SID

www.SID.ir

$$
\begin{aligned}
&\text{Min} \quad z \\
&s.t: \sum_{a\in A} d_a\left(x_a\right) - c_1 z \le b_1 \\
&\sum_{r=0}^{R-1} r\Delta t\left(P_1(r+1) - P_1(r)\right) - c_2 z \le b_2 \\
&\sum_{r=0}^{R-1} (r\Delta t)^2\left(P_1(r+1) - P_1(r)\right) - \left[\sum_{r=0}^{R-1} r\Delta t\left(P_1(r+1) - P_1(r)\right)\right]^2 - c_3 z \le b_3 \\
&P_1\left(\left[\frac{u}{\Delta t}\right]\right) - c_4 z \le b_4 \\
&P(r+1) = P(r) + Q(\mu)P(r)\Delta t \quad r = 0,1,2,...,R-1 \\
&P_i(0) = 0 \quad \forall i = 1,2,...,N-1 \\
&P_N(r) = 1 \quad r = 0,1,...,R \\
&P_i(r) \le 1 \quad i = 0,1,...,N-1, \quad r = 1,2,...,R \\
&g_a\left(x_a\right) = \frac{1}{\mu_a} \quad \forall a \in A \\
&\mu_a \ge \lambda + \varepsilon \quad \forall a \in B \\
&\mu_a \ge \varepsilon \quad \forall a \in C \\
&x_a \ge L_a \quad \forall a \in A \\
&x_a \le U_a \quad \forall a \in A \\
&\sum_{a\in A} x_a \le J
\end{aligned}
\tag{21}
$$

Consumable resource allocation in multi-class dynamic PERT networks

In practice, due to various projects' requirements, most organizations execute the projects from several classes, where new projects from different classes arrive to the system dynamically over the time horizon and service stations that stochastically serve to projects. In such conditions, organization is encountered with multi-class dynamic PERT networks, while projects from different classes differ in their precedence networks, the mean of their service time in every service station, and also their arrival rates.

In this section, we present a heuristic algorithm for consumable resource allocation problem in multi-class dynamic PERT networks, while this problem is a generalization of the resource allocation problem in dynamic PERT network, as investigated by Azaron and Tavakkoli-Moghaddam (2007) by considering only one type (class) of projects in multi-project system. For this purpose, it is assumed that we have *K* different classes from projects, where the new projects of class $i\,(i=1,...,K)$ arrive to system according to Poisson process with the rate of λ_i, and the service times in service station *a* for the projects of class *i* are exponentially distributed with the rate of μ_a^i. On the basis of this algorithm, multi-class dynamic PERT networks problem is decomposed into *K* single-class dynamic PERT network problem, while each class is considered separately as a minisystem. In every minisystem $i\,(i=1,...,K)$, all projects from different classes are considered as class *i* with the arrival rate of $\lambda\left(=\sum_{i=1}^{K}\lambda_i\right)$. Then every minisystem is solved by using the steps found in the sections of 'Single class dynamic PERT networks' as a single-class dynamic PERT network problem.

Archive of SID

www.SID.ir

Let $G_i = (V_i, A_i)$ be a directed stochastic network of class i $(i = 1, ..., K)$, in which V_i represents the set of nodes and A_i represents the set of arcs of the network in class i. Let s_i and t_i be the source and sink nodes in the AoA graph of class i, respectively. Let x_a^i represent the amount of resource to be allocated to the service station a ($a \in A_i$) in minisystem i, while all projects from different classes are considered as class i with the arrival rate of $\lambda \left(= \sum_{i=1}^{K} \lambda_i\right)$ and $x^i = \left[x_a^i : a \in A\right]^T$, where $A = \bigcup_{i=1}^{K} A_i$. Then the resources allocated to service station a in minisystem i should be computed by the model represented in 'Multi-objective consumable resource allocation' section. Therefore, the resources allocated to service station a would be

$$x_a = \frac{\sum_{a \in A_i} \left(\lambda_i . x_a^i\right)}{\sum_{a \in A_i} \lambda_i} \quad \forall a \in A \tag{22}$$

Let $x = \left[x_a : a \in A\right]^T$; therefore, the resources allocated to service stations are calculated as follows:

$$x = \sum_{i=1}^{K} \frac{\lambda_i}{\lambda} x^i \tag{23}$$

Moreover, Let z^i represent the objective function of minisystem i using goal attainment formulation. Therefore, the objective function of multi-class system would be

$$z = \sum_{i=1}^{K} \frac{\lambda_i}{\lambda} z^i \tag{24}$$

Consequently, the proposed algorithm is as follows.

Proposed Algorithm:

Step 1. Convert the multi-class dynamic PERT networks problem into K single-class dynamic PERT network problem (minisystem). For this purpose, in every minisystem i, consider the all projects from different classes as class i with the arrival rate of $\lambda \left(= \sum_{i=1}^{K} \lambda_i \right)$.

Step 2. For every minisystem i, compute the resources allocated to service stations, $x^i = [x_a{}^i : a \in A]^T$, and objective function, z^i, by the model represented in 'Multi-objective consumable resource allocation' section.

Step 3. Calculate the resources allocated to service stations, $x = [x_a : a \in A]^T$, by $x = \sum_{i=1}^{K} \frac{\lambda_i}{\lambda} x^i$ and also obtain the objective function of multi-class system by $z = \sum_{i=1}^{K} \frac{\lambda_i}{\lambda} z^i$.

Archive of SID

www.SID.ir

An illustrative case

In this section, to illustrate the proposed heuristic algorithm, we consider the three classes of projects depicted in Figure 2. The assumptions are as follows:

- The system has three different classes of projects.
- The new projects from different classes, including all their activities, are generated according to Poisson processes with the rates of $\lambda_1 = 2.5$, $\lambda_2 = 1.5$, $\lambda_3 = 1$ per year.
- The activity durations in service station a for the projects of class i ($i = 1, 2, 3$) are exponentially distributed. Table 2 shows the characteristics of the activities in different classes, where the time unit and the cost unit are, respectively, in year and in thousand dollars.
- There are infinite servers in service station 3 and other stations have only one server settled in the nodes.
- The capacity of the system is infinite.
- The threshold time that the project completion time does not exceed it for every minisystem is 3.
- The maximum amount of consumable resource available to be allocated to all service stations is $J = 15$.
- In all experiments, the value of ε is equal 0.01.
- The goals and the weights of objectives for goal attainment method in every class, are $b_1 = 25$, $b_2 = 1$, $b_3 = 0.25$, $b_4 = 0.95$ and $c_1 = 0.3$, $c_2 = 0.1$, $c_3 = 0.3$, $c_4 = 0.3$, respectively. Note that the goals and the weights of objectives for different classes can be different.

Figure 2 The precedence relations for the three project types.

Archive of SID

www.SID.ir

Table 2 Characteristics of the activities

Activity (a)	$d_a(x_a)$	Project type	$g_a(x_a)$	L_a	U_a
1	$1.2x_1+1$	1	$0.15-0.02x_1$	1	5
		2	$0.17-0.02x_1$		
		3	$0.12-0.015x_1$		
2	$1.2x_2+0.8$	1	$0.2-0.015x_2$	1	6
		2	$0.18-0.01x_2$		
		3	$0.15-0.01x_2$		
3	$1.8x_3+1$	1	$0.8-0.1x_3$	1	5
		2	$0.9-0.11x_3$		
		3	$0.85-0.1x_3$		
4	$x_4+0.7$	1	$0.12-0.01x_4$	1	5
		2	$0.1-0.01x_4$		
		3	$0.14-0.02x_4$		
5	$1.3x_5+1.5$	1	$0.14-0.02x_5$	1	6
		2	$0.12-0.01x_5$		
		3	$0.15-0.02x_5$		
6	$1.5x_6+0.5$	1	$0.18-0.01x_6$	1	5
		2	$0.16-0.01x_6$		
		3	$0.2-0.02x_6$		

In this example, three-class dynamic PERT networks problem is decomposed into three single-class dynamic PERT network problem, while each class is considered separately as a minisystem. In every minisystem $i(i = 1,2,3)$, all projects from different classes are considered as class i with the arrival rate of $\lambda = 5(= 2.5+1.5+1)$. Then, every minisystem is solved as single-class dynamic PERT network problem by using the steps found in the sections of 'Single class dynamic PERT networks.'

The objective is to obtain the resources allocated to the different activities by using proposed algorithm in 'Consumable resource allocation in multi-class dynamic PERT networks'. For this purpose, the system states and transition rates for every minisystem are determined (depicted in Figure 3), where the states are showed in nodes and the rate diagrams are represented on arcs.

Figure 3 The admissible 2-partition cuts and rate diagram for every single class in example.

We obtain the infinitesimal generator matrix $Q(\mu)$ for forming the corresponding discrete state model in every class. We consider the mentioned goals and weights and also the various combinations of R and Δt. To do so, we employ LINGO 8 on a PC Pentium 4, CPU 3 GHz. The optimal allocated resources in every minisystem, the computational times, CT (mm/ss), and also the values of all objectives for the different combinations of R and Δt, are shown in Table 3 for every minisystem. So based on proposed algorithm in 'Consumable resource allocation in multi-class dynamic PERT networks' section the allocated consumable

Archive of SID

www.SID.ir

resources are $x_1 = \frac{1.737 \times 2.5 + 2.875 \times 1.5 + 1 \times 1}{5} = 1.931$, x_2 = 3.9797, x_3 = 1.6626, x_4 = 1.1976, x_5 = 1.9224, and x_6 = 4.3017 (z = 5.7993).

Archive of SID

www.SID.ir

Table 3 The computational results

Project type	*R* value	Δt	x_1	x_2	x_3	x_4	x_5	x_6	z	f_1	f_2	f_3	f_4	CT
1	50	0.16	1.732	4.312	1.143	1	1.893	4.920	7.332	25.651	1.733	0.399	0.935	00′:07″
	80	0.1	1.735	4.313	1.132	1	1.896	4.923	7.331	25.646	1.733	0.538	0.931	00′:25″
	100	0.08	1.737	4.314	1.126	1	1.898	4.925	7.331	25.64	1.733	0.590	0.931	00′:31″
2	50	0.16	2.619	4.685	3.056	1	1	2.639	5.435	26.025	1.543	0.271	0.972	00′:04″
	80	0.1	2.874	4.379	2.411	1	1	3.335	5.342	25.847	1.534	0.377	0.970	00′:08″
	100	0.08	2.875	4.381	2.408	1	1	3.336	5.342	25.846	1.534	0.408	0.965	00′:21″
3	50	0.16	1	2.545	2.070	2.019	3.056	4.218	2.665	25.800	1.267	0.218	0.983	00′:10″
	80	0.1	1	2.541	1.887	1.988	3.366	4.192	2.656	25.797	1.266	0.308	0.982	00′:19″
	100	0.08	1	2.542	1.886	1.988	3.367	4.192	2.656	25.797	1.266	0.333	0.980	00′:35″

Archive of SID

www.SID.ir

To illustrate the performance of proposed algorithm, we evaluate its results with a simulation method that randomly allocate the resources to service stations and simulate the multi-class dynamic PERT network through Mont Carlo simulation. Note that Fatemi Ghomi and Hashemin (1999) proposed conditional Monte Carlo simulation and crude Monte Carlo simulation to compute the network completion time distribution function. However, the simulation method will be as follows:

Simulation method:

n=1

Generate one initial solution $x=\left[x_a : a\in A\right]^T$, randomly, where $x_a \in \left[L_a, U_a\right]$ and $\sum_{a\in A} x_a \le J$

Simulate the multi-class dynamic PERT network (Figure 2) according to the initial solution

Calculate the objective function, *z*, for the initial solution

Repeat

Generate a new solution, $x^{\text{new}}=\left[x_a^{\text{new}} : a\in A\right]^T$, randomly, where $x_a^{\text{new}} \in \left[L_a, U_a\right]$ and $\sum_{a\in A} x_a \le J$

Simulate the multi-class dynamic PERT network according to the new solution

Calculate the objective function, z^{new}, for the new solution

Calculate $\Delta z = z^{\text{new}} - z$

If $\Delta z < 0$ then $x = x^{\text{new}}$, $z = z^{\text{new}}$

$n = n+1$

Until $n \ge N$, where *N* is number of simulation run

Present *x* and *z*

We set the number of simulation run to be *N* = 8,000. According to simulation method, the simulation results are $x_1^{sim}=2.178$, $x_2^{sim}=3.769$, $x_3^{sim}=2.025$, $x_4^{sim}=1.212$, $x_5^{sim}=2.064$, $x_6^{sim}=3.723$ $(z^{sim}=5.051)$. In Figure 4, a comparative analysis of our proposed model in 'Consumable resource allocation in multi-class dynamic PERT networks,' and simulation results for this example is presented. Based on this example, the objective function values obtained through both algorithms are close.

Figure 4 Comparative analysis of our proposed model and simulation results.

Conclusions

In this article, we proposed a heuristic method for the consumable resource allocation problem (or time–cost trade off problem) in multi-class dynamic PERT networks. It was assumed that the new projects of different classes, including all their activities, are entered to the system according to an independent Poisson processes, and each activity of any project is executed at a devoted service station settled in a node of the network according to its class. The number of servers in every service station is either 1 or infinity, while the service times (activity durations) are independent random variables with exponential distributions. Indeed, each project arrives to the first service station and continues its routing according to its

Archive of SID

www.SID.ir

precedence network of corresponding class. Such system was considered as a queuing network, while the discipline of queues is first come, first served.

For determining the resources allocated to the service stations, the multi-class system was decomposed into several single-class dynamic PERT networks, whereas each class was considered separately as a minisystem. For modeling of single-class dynamic PERT network, we used Markov process and a multi-objective model, investigated by Azaron and Tavakkoli-Moghaddam (2007), namely, we first converted the network of queues into a stochastic network. Then, by constructing a proper finite-state absorbing continuous-time Markov model, a system of differential equations was created. Next, we applied a multi-objective model containing four conflicted objectives to optimally control the resources allocated to service stations in single-class dynamic PERT network and further used the goal attainment method to solve a discrete-time approximation of the primary multi-objective problem. Finally, after obtaining the resources allocated to service stations in every minisystem, the final resources allocated to servers were calculated by proposed algorithm.

In multi-objective 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 that 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.

For obtaining the best optimal allocated consumable resource in single-class dynamic PERT network, we considered the various combinations of portions for specific time interval. Based on the presented example, If the length of every portion is decreased, the accuracy of solution is increased, i.e., the value of objective is decreased and the computational time, CT, is also increased.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

SY studied the associated literature review and presented the mathematical model and simulation algorithm. He was also responsible for revising the manuscript. SN managed the study and participated in the problem solving approaches. MMM approved the model and analyzed the required data and conducted the experimental analysis. All authors read and approved the final manuscript.

Authors' information

SY is an Assistant Professor of Industrial Engineering at Iran University of Science and Technology (IUST), Tehran, Iran. His current research interests include stochastic processes and their applications, project scheduling and supply chains. He has contributed articles to different international journals, such as *European Journal of Operational Research*, *International Journal of System Science*, *Iranian Journal of Science & Technology*, and

Archive of SID

www.SID.ir

International Journal of Advanced Manufacturing Technology. SN is the Associate Professor of Industrial Engineering Department in IUST since October 1990. The teaching experiences are mainly in project management, human resource management, and engineering economics. MMM also is the Assistant Professor of Industrial Engineering Department in IUST. His research interests include performance measurement systems, inventory planning, and control and strategic planning.

References

Adler PS, Mandelbaum A, Nguyen V, Schwerer E (1995) From project to process management: an empirically-based framework for analyzing product development time. Manag Sci 41(3):458–484

Anavi-Isakow S, Golany B (2003) Managing multi-project environments through constant work-in-process. Int J Proj Manag 21(1):9–18

Azaron A, Modarres M (2005) Distribution function of the shortest path in networks of queues. OR Spectr 27:123–144

Azaron A, Tavakkoli-Moghaddam R (2007) Multi-objective time–cost trade-off in dynamic PERT networks using an interactive approach. Eur J Oper Res 180:1186–1200

Azaron A, Fynes B, Modarres M (2011) Due date assignment in repetitive projects. Int J Prod Econ 129:79–85

Byali RP, Kannan MV (2008) Critical Chain Project Management - a new project management philosophy for multi project environment. J Spacecraft Technol 18:30–36

Chen V (1994) A 0–1 goal programming-model for scheduling multiple maintenance projects at a copper mine. Eur J Oper Res 6:176–191

Chen PH, Shahandashti SM (2009) Hybrid of genetic algorithm and simulated annealing for multiple project scheduling with multiple resource constraints. Autom Constr 18:434–443

Cohen I, Golany B, Shtub A (2005) Managing stochastic, finite capacity, multi-project systems through the cross entropy methodology. Ann Oper Res 134:183–199

Cohen I, Golany B, Shtub A (2007) Resource allocation in stochastic, finite-capacity, multi-project systems through the cross entropy methodology. J Sched 10:181–193

Elmaghraby SE (1993) Resource allocation via dynamic programming in activity networks. Eur J Oper Res 64:199–215

Elmaghraby SE (1996) Optimal procedures for the discrete time/cost trade-off problem in project networks. Eur J Oper Res 88:50–68

Elmaghraby SE, Morgan CD (2007) A sample-path optimization approach for optimal resource allocation in stochastic projects. J Econ Manag 3:367–389

Archive of SID

www.SID.ir

Elmaghraby SE, Salem AM (1982) Optimal project compression under convex functions, I and II. Appl Manag Sci 2:1–39

Elmaghraby SE, Salem AM (1984) Optimal linear approximation in project compression. IIE Trans 16:339–347

Fatemi Ghomi SMT, Ashjari B (2002) A simulation model for multi-project resource allocation. Int J Proj Manag 20:127–130

Fatemi Ghomi SMT, Hashemin SS (1999) A new analytical algorithm and generation of Gaussian quadrature formula for stochastic network. Eur J Oper Res 114:610–625

Fulkerson DR (1961) A network flow computation for project curves. Manag Sci 7:167–178

Goncalves JF, Mendes JJM, Resende MGC (2008) A genetic algorithm for the resource constrained multi-project scheduling problem. Eur J Oper Res 189:1171–1190

Kanagasabapathi B, Rajendran C, Ananthanarayanan K (2009) Performance analysis of scheduling rules in resource-constrained multiple projects. Int J Ind Syst Eng 4:502–535

Kao H-P, Hsieh B, Yeh Y (2006) A petri-net based approach for scheduling and rescheduling resource-constrained multiple projects. J Chinese Inst Ind Eng 23(6):468–477

Kruger D, Scholl A (2008) Managing and modelling general resource transfers in (multi-) project scheduling. OR Spectr 32:369–394

Kulkarni V, Adlakha V (1986) Markov and Markov-regenerative PERT networks. Oper Res 34:769–781

Kumanan S, Jegan JG, Raja K (2006) Multi-project scheduling using an heuristic and a genetic algorithm. Int J Adv Manuf Technol 31:360–366

Kurtulus IS, Davis EW (1982) Scheduling: categorization of heuristic rules performance. Manag Sci 28:161–172

Kurtulus IS, Narula SC (1985) Multi-project scheduling: analysis of project performance. IIE Trans 17:58–66

Li C, Wang K (2009) The risk element transmission theory research of multi-objective risk-time–cost trade-off. Compute Math with Appl 57:1792–1799

Lova A, Tormos P (2001) Analysis of scheduling schemes and heuristic rules performance in resource-constrained multiproject scheduling. Ann Oper Res 102:263–286

Lova A, Maroto C, Tormos P (2000) A multicriteria heuristic method to improve resource allocation in multiproject scheduling. Eur J Oper Res 127:408–424

Nozick LK, Turnquist MA, Xu N (2004) Managing portfolios of projects under uncertainty. Ann Operation Res 132:243–256

Archive of SID

www.SID.ir

Payne JH (1995) Management of multiple simultaneous projects: a state-of-the-art review. Int J Proj Manag 13:163–168

Pritsker AAB, Watters LJ, Wolfe PM (1969) Multiproject scheduling with limited resources: a zero–one programming approach. Manag Sci 16(1):93–108

Tsubakitani S, Deckro RF (1990) A heuristic for multi-project scheduling with limited resources in the housing industry. Eur J Oper Res 49:80–91

Wiest JD (1967) A heuristic model for scheduling large projects with limited resources. Manag Sci 13(6):359–377

Ying Y, Shou Y, Li M (2009) Hybrid genetic algorithm for resource constrained multi-project scheduling problem. J Zhejiang University (Eng Sci) 43:23–27

Archive of SID

www.SID.ir

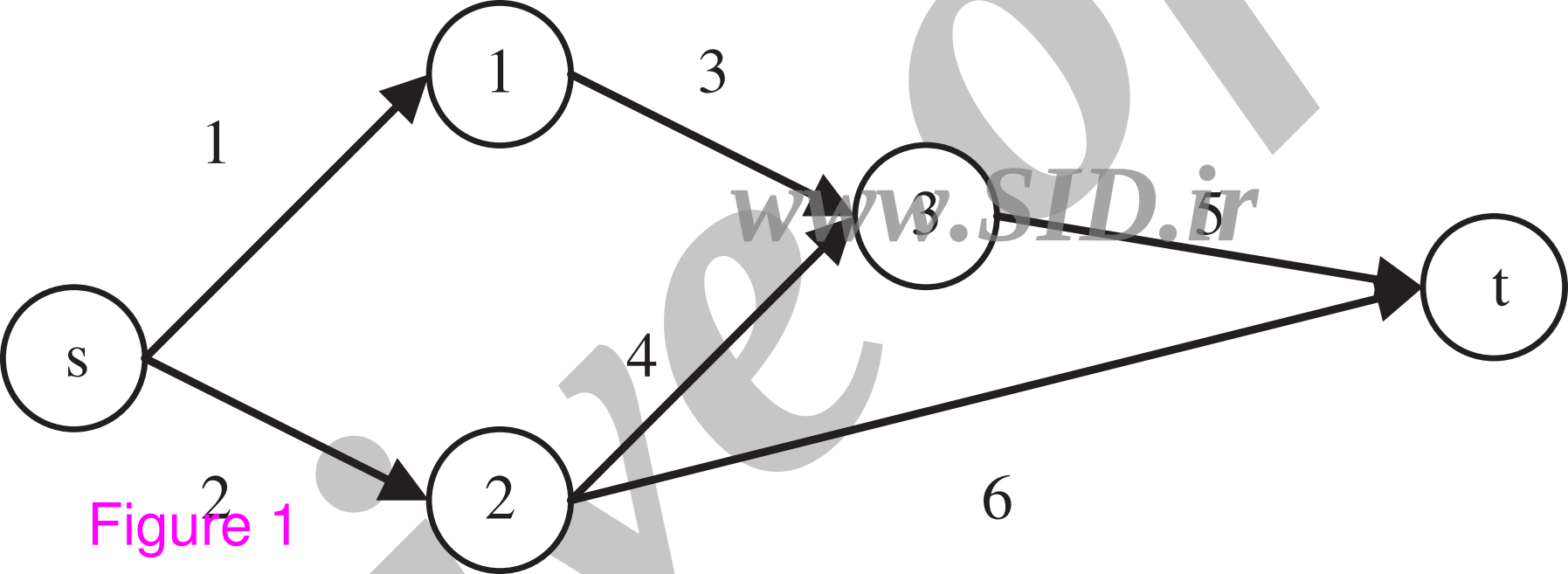


Figure 1

www.SID.ir

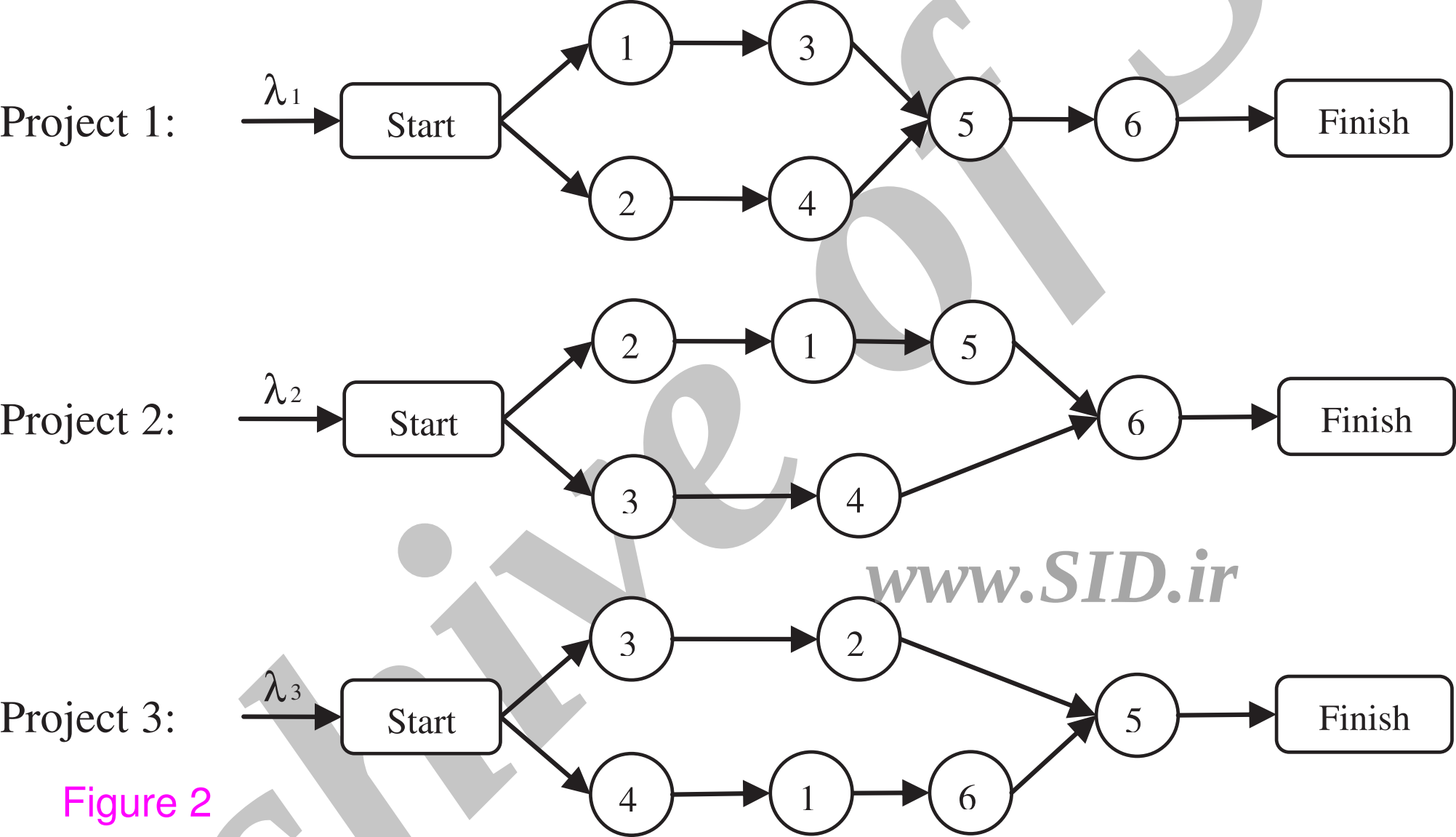


Figure 2

www.SID.ir

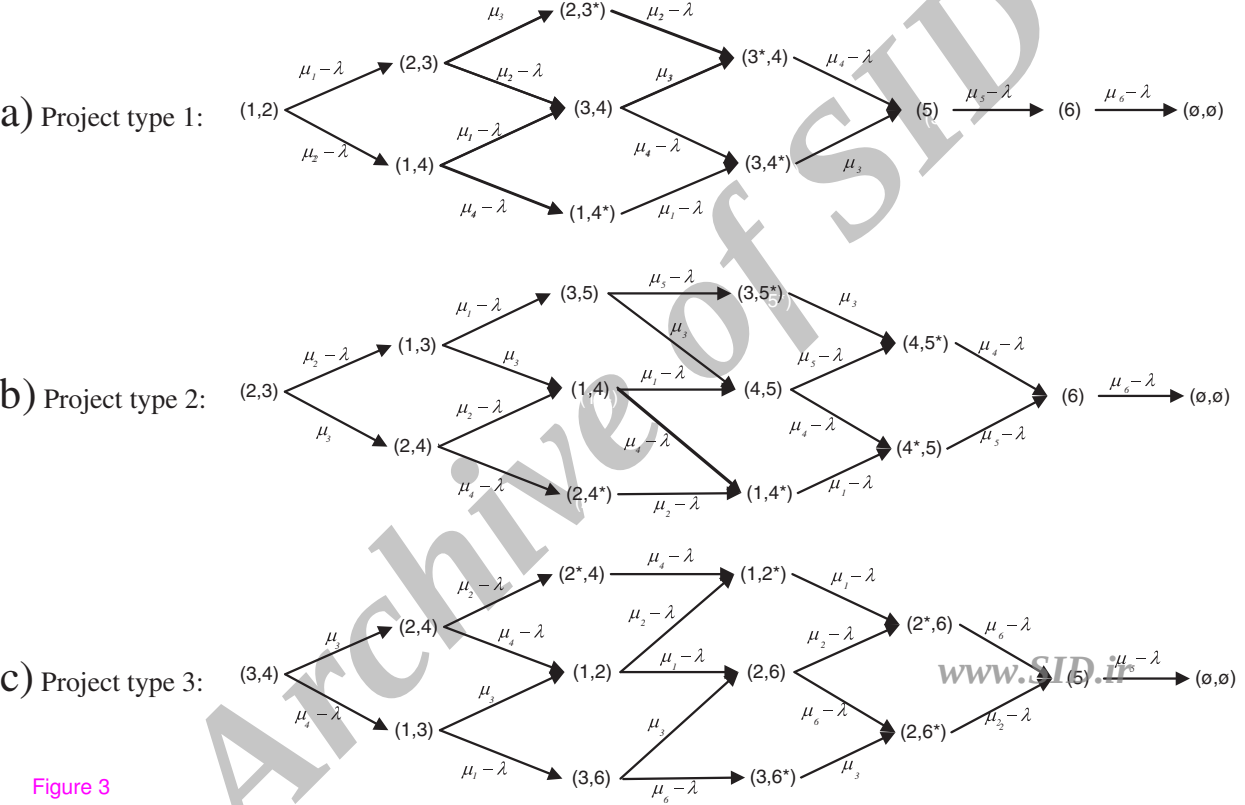
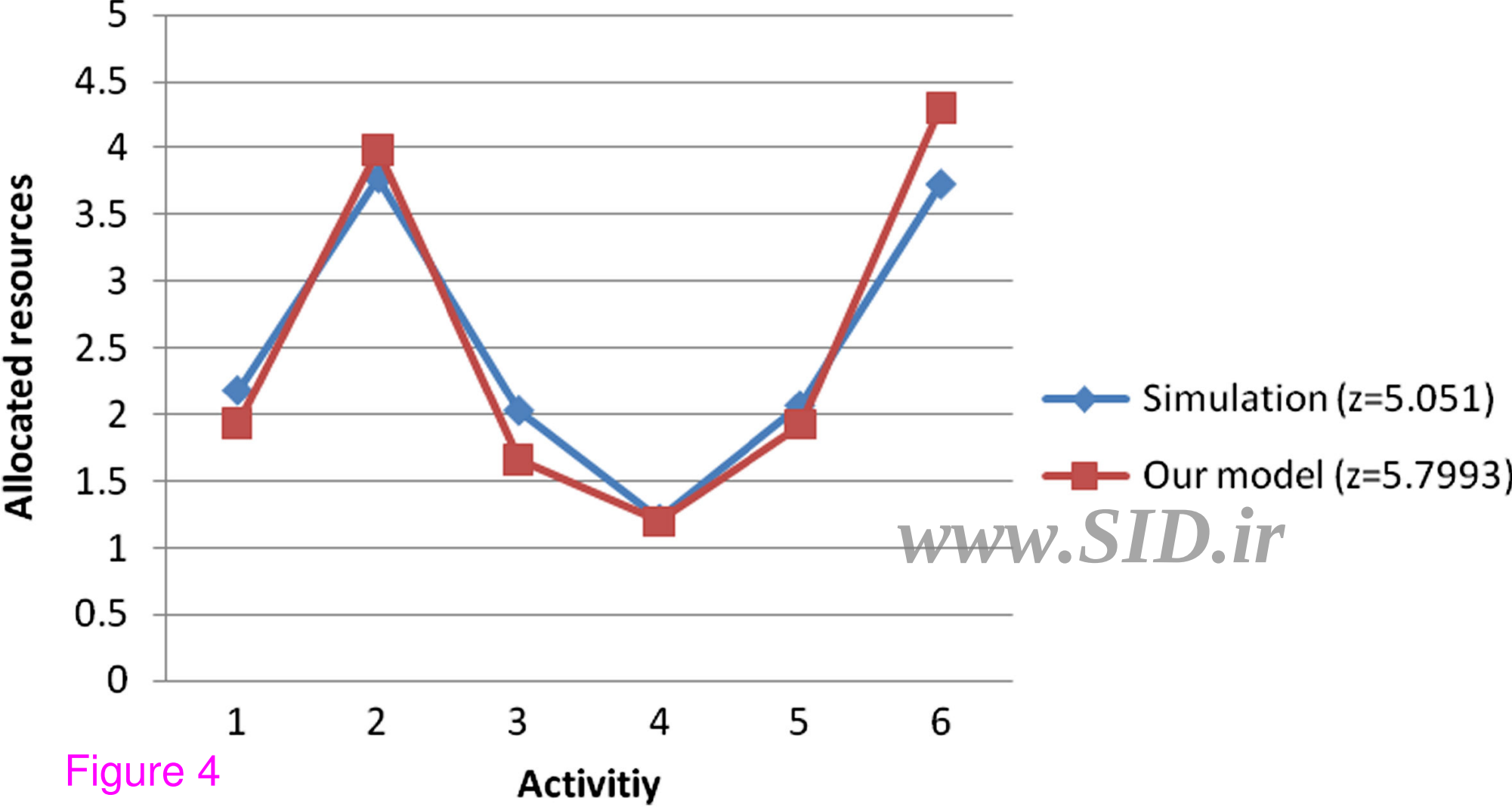


Figure 3

Archive of SID
www.SID.ir



www.SID.ir

Figure 4