

An improved fuzzy critical chain approach in order to face uncertainty in project scheduling

E. Roghanian ^a, M. Alipour ^{b,*}, and M. Rezaei ^c

^a *Department of Industrial Engineering, K. N. Toosi University of Technology, Tehran, Iran.*

^b *Sustainable Energies Group, AUT Office of Sustainability, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran.*

^c *Department of Industrial Engineering, Tarbiat Modares University, Tehran, Iran.*

Abstract

This paper proposes an improved critical chain method with fuzzy approach to schedule projects under uncertainty. Utilizing fuzzy numbers can be wise because of the natural uncertainty in estimation of required work to accomplish the project tasks. To minimize the project duration as an objective taking into account resource limitations and other constrains, a Resource Constraint Project Scheduling Problem model (RCPSP) is adopted. The output of this model determines the project schedule. Then, a new buffer sizing approach based on the square root of the sum of the squares (SSQ) method is introduced which is modified by coefficients to overcome the schedule risk more efficiently. As a result, a project buffer is added to the end of the project schedule and is used if necessary. The methodology can be used as a control procedure.

Keywords: Project scheduling; Critical chain project management (CCPM); Buffer management; Fuzzy numbers.

1. Introduction

As stated in *PMBOK® Guide* (PMI 2008), project management processes are grouped into five categories known as the project management process groups, including: Initiating, Planning, Executing, Controlling, and Closing. At the second stage (planning process group), scheduling involves construction of a project-based plan that specifies each activity: the precedence and

*Corresponding author. E-mail address: Alipour.m.86@gmail.com (M. Alipour).

resource feasible start, completion dates, the amounts of the various resource types that will be needed during each time period, and as a result, the budget (Demeulemeester & Herroelen 2002). Besides, one of the nine project management knowledge areas is the project time management. For the project managers, one of the main planning goals is to determine the total project completing duration (or makespan). Accurate estimation of the project duration is a complex activity because of its dependence on a large number of factors, which usually cannot be controlled adequately. Estimating the activity durations, assigning the start time and finally, preparing a schedule (in the form of a Gantt chart) is the most important determinant to achieve this goal.

Utilization of traditional methods such as Critical Path Method (CPM) by deterministic activity duration fail to overcome the common uncertainty in the project elements. To tackle this challenge, Goldratt and Cox (1984) developed a new concept named Theory of Constraints (TOC). Later, Goldratt (1997) proposed a direct application of TOC in project management, known as Critical Chain Project Management (CCPM) or buffer management. This method is a competent approach to handle the project duration risk as well as minimizing the makespan. The main idea of the approach is as the follows:

- (1) Removing the hidden safety in activity durations to protect activities from late starting, which he called Student syndrome;
- (2) Preventing busy pretension by staff named Parkinson's Law;
- (3) Preventing the late finishing activities because of Murphy's Law or multitasking.

These hidden safeties after removal from the former places are located following a special set of activities called a chain to preclude the project delay. The common buffer types used in CCPM are project buffer, feeding buffer, and resource buffer. The main step in the CCPM

procedure is to determine the buffer sizes (or buffer sizing). Some of the existing methods for buffer sizing use two kinds of task duration called *safe estimate* (or about 95% confidence duration estimate) and *the average (50%) estimate*.

Therefore, CCPM by these assumptions is associated with probability theory. However, probability theory has two problems to be used in this case:

1. Probability theory cannot be applied for non-routine projects because of lack of statistical data (Long & Ohsato 2008).
2. Asking a probability distribution or even two task durations with completion probability of 50% and 95% from experts is rather difficult and imprecise.

Thus, using the fuzzy sets theory, a new critical chain project scheduling is developed in this study to overcome the uncertainty and resource constraints. A Resource-Constrained Project Scheduling Problem (RCPSP) model is adopted to minimize the project duration as an objective taking into account resource limitations and other constraints. There are different buffer sizing methods proposed in the literature. In this research, a brief description of some of the identified methods is given and their advantages and disadvantages are discussed. Then, a new buffer sizing method is proposed to have a more efficient critical chain scheduling. Finally, a numerical example is provided to compare the efficiency of the proposed model with other common approaches.

The remainder of the paper is organized as follows. Next section brings a brief review of critical chain concepts. Section 3 describes the improved fuzzy critical chain approach under uncertainty, which is the body of this research. A numerical example illustrating the proposed methodology is developed in section 4. And concluding remarks are given in section 5.

2. Critical chain project management concepts

2.1. The necessity of a new approach

Recent researches and real experiences suggest that there are a few barriers in existing methods of project scheduling. Some of them include: late completion, over spending, and other undesirable effects that any project manager is familiar with. In Goldratt's (1997) point of view, high certainty of activity completion is the primary reason for project overruns. For example, in the traditional methods like CPM, task durations are estimated by safety times (in 90% probability of completion). Thus, when all activities join together, the project would have a high probability of on-time completion. Finally, the fundamental reason for overruns in most projects is: employees know that safety time is built into the estimates. As a result, starting on time is not of any significance. If starts are made on time, there is a tendency not to go at full steam because of the feeling of having time in hand. Consequently, employees will not work with high performance. This is likely to imply more pressure to perform more than two tasks, called multitasking, which causes lateness or time blindness due to the lack of concentration (Goldratt 1997; Leach 2014).

2.2. The theory of constraints

The theory of constraints is a methodology for identifying basic problems and constraints (limiting factors) to propose a proper solution. Constraints or bottlenecks are parts of the systems that affect the objectives, restricting the output of the entire system (Ahlemann et al. 2013). Making a profit -both in the short and long term- is the ultimate goal of TOC (Goldratt & Cox 1984; 1997). To achieve this goal, there are five focusing steps that address influential constraints:

1. Identify the system's constraints(s);
2. Exploit constraint in the system;
3. Subordinate everything else to the above decision;
4. Elevate the system's constraint(s);
5. If there is an un-covered constraint, go back to step 1, and prevent inertia to cause a system's constraint.

One of the significant results of TOC is critical chain idea (Tukel et al. 2006).

2.3. Fundamentals of critical chain scheduling

Delay in projects is one of the paramount challenges in the existing methods. To avoid this obstacle, CCPM has been applied and utilized as a project management strategy. This strategy avoids major impact of Parkinson's Law at the task level while accounting for Murphy's Law at the project level (Ghaffari & Emsley 2015). To minimize the effect of Parkinson's Law, CCPM builds the schedule with target duration estimations based on a 50% confidence level. These durations are estimated regardless of task due dates and milestones by eliminating multitasking (Hall 2015; Horroelen & leus 2001). These eliminated safeties are placed at the end as a buffer (Figure 1). Stated estimations will correct this feeling that there is time in hand. Consequently, employees work with a higher performance to finish their tasks on time which contributes to eliminating multitasking.

Figure 1 . PERT/CPM and TOC comparison.

2.4. Critical chain recognition

The reason for the development of “Critical Chain” is the existence of chronic problems that available methods and approaches have not been able to obviate (Rand 2000). There are various activities involved in a real project while several resources are available in every project. CCPM seeks to optimize these resource limits, thus each activity should have its specific resources and duration. How is it decided which activity should be worked on at any specific time? There are four steps in recognition of the critical chain in a project (Goldratt 1997):

- (1) Allocating the durations. Project leaders ought to allocate each activity an appropriate duration under the stated condition.
- (2) Moving activities forward. In this step, each activity is pushed forward as much as possible. One of the considerations is that the precedence relationships are met between the activities.
- (3) Solving resource over-allocations. This is one of the substantial steps in accessing feasible schedule. After this step, there should not be any over-allocations in any resources. For example, if two activities in a schedule demand one employee at a specific time, it will not be done (considering only one employee is available).
- (4) Setting buffers. In this step, each buffer is located in a specific place.

Critical chain is defined as the longest chain of dependent constraints. To determine this, it is necessary to take into account any dependencies that might exist between activities, as they require the same resources (Rand 2000; Yang & Fu 2014). Resources conflict is resolved by moving tasks earlier in time (Newbold 1998). Horroelen and Leus (2001) state that the critical chain is known as the longest chain which determines the project lead time. Critical chain and critical path are the same in some ways, however, there are three principal differences between them (Bevilacqua et al. 2009):

- (1) The way experts assign the activity time;
- (2) The time that managers use project buffer;
- (3) The way a resource conflict is considered.

2.5. Project buffer and feeding buffers

In the CCPM method, mean task durations are used instead of durations with safety time. The difference between these two types of duration for each activity in critical chain is shifted to the end of the project as a project buffer (shown in [Figure 2](#)). This buffer tries to ensure the date of completion. It may occur that a project manager uses some of the project buffer or does not use it at all. In actual projects, there is more than one chain, and thus, CCPM predicts a buffer to protect these chains from delay. This buffer is called feeding buffer (feeding buffer in [Figure 2](#)) and is placed where a non-critical chain activity joins the critical chain. Finally, these buffers protect the critical chain activities from disruptions and late starting (Horroelen & Leus [2001](#)).

Figure 2. Project buffer and feeding buffer.

3. Improved fuzzy critical chain approach to protect project schedule under uncertainty

3.1. The proposed methodology steps

In this subsection, the proposed methodology steps are represented. These steps can be traced to prepare and update an efficient schedule:

Step 1. Preparing fuzzy numbers for total work (total resource) needed to complete each task based on expert judgment.

Step 2. Defuzzification (Crisping) of the fuzzy numbers to a 95% point (safe estimate of the required work for each activity) and a 50% point (mean estimate of the required work for each activity).

Step 3. Solving a RCPSP model; with the 50% defuzzified numbers for each task's required work (from the previous step), as its inputs and assigned start times and activity durations are its outputs.

Step 4. Sizing the project buffer and feeding buffers by using the previous outputs.

Step 5. Controlling the project schedule by fuzzy CCPM approach.

3.2. Step 1: Estimating the required work (resource) for each task based on expert's judgment

The first step in the proposed methodology is asking experts to estimate the total required work (or resource k) to accomplish activity j in the form of a fuzzy number $TR(j,k)$. This is because of the usual uncertainty in the estimation procedure. Each fuzzy set is characterized by a membership function $\mu_{\bar{A}}(x)$ defining the membership degree of $TR(j,k)$ which is usually in the $[0,1]$ range (Roghianian & Alipour 2014). In this reseach, Trapezoidal Fuzzy Numbers ($TrFN(a_{jk}, b_{jk}, c_{jk}, d_{jk})$) are applied to represent the resources required for the tasks. $\mu_{\bar{A}}(TR(j,k))$ is the membership of $TR(j,k)$ depicted in Figure 3 and formulated in Equation (1) (Cheng 1998).

$$\mu_{\bar{A}}(TR(j,k)) = \begin{cases} (TR(j,k)-a_{jk})/(b_{jk}-a_{jk}), & \text{for } TR(j,k) \in [a_{jk}, b_{jk}] \\ 1, & \text{for } TR(j,k) \in [b_{jk}, c_{jk}] \\ (d_{jk}-TR(j,k))/(d_{jk}-c_{jk}), & \text{for } TR(j,k) \in [c_{jk}, d_{jk}] \\ 0, & \text{Otherwise} \end{cases} \quad (1)$$

Where $TR(j,k)$ is the total required resource of type k to complete activity j .

Figure 3. The membership function of the fuzzy number for the required resource k for activity j .

3.3. Step2: preparing the safe estimate and mean estimate of resource consumption for each activity

In classic CPM method, where experts are asked to estimate task duration (or required resource) to accomplish activities, they are usually interested in giving a duration with plenty of certainty or a duration with up to 90% confidence ($D^h(j)$). However, a task duration is required which its extra self-buffer has been removed; or a mean duration on which the task schedule has been established ($D^m(j)$). Consequently, these two kinds of durations are used to determine the buffers.

Long and Ohsato (2008) used the fixed required works for each task and put them in a RCPSP model. The output of this model for each task is a duration which they use as mean estimate durations. Fuzzy numbers were then used to determine safe estimate of the durations, but the proposed method determines these mean estimate durations ($D^m(j)$) by providing mean estimates of required resource k for task j ($TR^m(j,k)$).

These mean estimates of total required resource are calculated by using a concept named high agreement index (AI) introduced by Kaufman and Gupta (1985). This concept is used to depict the agreement of two fuzzy events A and B . AI shows the satisfied level of the fuzzy event A when A is compared with the fuzzy event B . It also illustrates the satisfied level of the fuzzy event A when A is compared with the fuzzy event B :

$$AI(A,B) = \frac{S(A \cap B)}{S(A)} \quad (2)$$

$$\text{Where } S(A) = \int \mu_A(x) dx \quad (3)$$

$$\text{And } S(A \cap B) = \int \mu_{A \cap B}(x) dx \quad (4)$$

Where $\mu_A(x)$ is the membership function of fuzzy number A and $S(A)$ is its integral on the defined domain.

AI concept is applied here to make a connection between fuzzy sets and probability theory. It means, if AI and the needed percentage of confidence is equated, the required defuzzified value will be obtained from Equation (5).

$$AI = \frac{\frac{(c_{jk} - b_{jk} + d_{jk} - a_{jk})}{2} - \frac{(d_{jk} - TR^x(j, k))^2}{2(d_{jk} - c_{jk})}}{\frac{(c_{jk} - b_{jk} + d_{jk} - a_{jk})}{2}} \quad (5)$$

If \tilde{A} is depicted by $TrFN(a_{jk}, b_{jk}, c_{jk}, d_{jk})$ and \tilde{B} by $TrFN(0, 0, TR^x(j, k))$, then $TR^x(j, k)$ is obtained as follows (see [Figure 4](#)).

$$TR^x(j, k) = d_{jk} - \sqrt{(1 - AI) * \left(c_{jk} - b_{jk} + d_{jk} - \frac{a_{jk}}{2} \right) * (2d_{jk} - c_{jk})} \quad (6)$$

$$\text{Where } TR^x(j, k) = \begin{cases} (TR^h(j, k), & \text{if } AI = 0.95 \\ (TR^m(j, k), & \text{if } AI = 0.5 \end{cases} \quad (7)$$

Figure 4. High agreement index (AI) of required resource k for task j .

3.4. Step 3: Solving the resource- constrained project scheduling problem model

Project scheduling deals with assigning and allocating of determinate resources to a set of activities connected in a bounded network (Fang & Wang 2012). The most significant goal in this situation is to accomplish the project by these constrained resources with minimum lateness.

As a result, RCPSP consists of minimizing the duration of completing a project, subject to precedence relations and resource constraints and availability.

The first stage in this step is to model the problem. As classified by Demeulemeester and Herroelen (2002), there are many types of basic formulation and solution methods for the RCPSP models, such as: linear programming based approaches (Lawrence 1988; Klein 1999; Mingozzi et al. 1998; Olaguibel and Goerlich 1993; Pritsker et al. 1969), branch- and- bound procedures (Agin 1966; Brucker et al. 1998; De Reyck and Herroelen 1998; Heilmann 2003), heuristic and metaheuristic approaches like Genetic Algorithm (Barrios et al. 2011; Chen and Weng 2009; Valls et al. 2008), simulated annealing (Bouleimen and Lecoeq 2003), Tabu Search (Pan et al. 2008), ant colony (Duan and Liao 2010), particle swarm (Chen 2011; Zhang et al. 2005) and etc. However, there is a conceptual form of modeling used by practitioners as following:

Minimize f_n

Subject to

$$f_i \leq f_j - d_j \quad \text{for} \quad \text{all } (i, j) \in A \quad (8)$$

$$f_1 = 0 \quad (9)$$

$$\sum_{i \in s_t} r_{ik} \leq a_k \quad \text{for} \quad k = 1, 2, \dots, m \quad \text{and} \quad t = 1, \dots, f_n \quad (10)$$

In this model, activities are considered as the dummy start and dummy finish of the project.

The parameters and variables are defined as:

- f_i : Finish time of activity i ;
- d_i : Duration of activity i ;
- r_{ik} : Resource requirement of activity i for resource type k per time unit;
- s_t : Set of activities that are in progress at time t ;

- a_k : Available amount of resource type k per time unit;
- k : Number of resources;
- A : Set of activities.

The model depicted above is a conceptual model with some elementary constraints. There are various modeling forms with different assumptions. The modeling procedure used in this study is called Discrete Time/ Resource Trade-off Problem (DTRTP). Demeulemeester et al. (1996) are the first researchers to deal with discrete time resource trade-offs. In this problem, it is assumed that the total required work (resource) to accomplish each activity is known and the task duration is one of the decision variables. Thus, there are plenty of approaches to schedule a task in different durations and by different required resource per day tradeoffs (with fixed total work). Each of the task durations can vary within their upper and lower bounds. These bounds determine the reasonable ranges for each activity duration and are stated by experts.

There is a DTRTP model applied by Long and Ohsato (2008) which is used in the proposed methodology:

Objective function:

$$\text{Minimize } T = \max_{j=1, \dots, N} (AF_j) \quad (11)$$

Subject to:

$$1. D^l(j) \leq D^*(j) \leq D^u(j) \quad j=1, \dots, N \quad (12)$$

$$2. AS(j_1) + D^*(j_1) \leq AS(j) \quad j=1, \dots, N \quad \text{and} \quad \forall j_1 \in P(j) \quad (13)$$

$$3. RR(k, t) \leq RA(k, t) \quad k = 1, \dots, R \quad \text{and} \quad t = 1, \dots, T^u \quad (14)$$

$$4. RR(k, t) = \sum_{j \in set(t)} r(j, k) = \sum_{j \in set(t)} \frac{TR(j, k)}{D^*(j)} \quad (15)$$

$$5. T^u = \sum_{j=1, \dots, N} D^u(j) \quad (16)$$

$$6. AF(j) = AS(j) + D^*(j) \quad j=1, \dots, N \quad (17)$$

$$7. ES(j) \leq AS(j) \leq LS(j) \quad j=1, \dots, N \quad (18)$$

Where:

- $AS(j), AF(j)$: Assigned start and finish times of activity j ;
- $D^*(j)$: Assigned duration of activity j ;
- $[D^l(j), D^u(j)]$: Acceptable range for duration of activity j determined by experts;
- $ES(j), LS(j)$: Earliest and latest start of activity j by the standard CPM, where $ES(j)$ is determined by forward calculation and $LS(j)$ by backward calculation from the project duration upper bound (T^u);
- k : Resource type $k, k = 1, \dots, R$;
- N : Total number of activities;
- $P(j)$: A set of immediate predecessors of activity j ;
- $r(j,k)$: Daily required resource k to perform activity j ;
- R : Number of resources;
- $RA(k,t)$: Total availability of resource k at day t ;
- $RR(k,t)$: Required resource k at day t ;
- $Set(t)$: Set of all activities in progress at day t , where $AS(j) \leq t \leq AF(j)$;
- T : Project duration under resource constraints;
- $TR(j,k)$: Total required resource k to perform activity j ;
- T^u : Project duration upper bound.

The first constraint represents certain ranges for each activity duration which is determined by expert judgment. The second constraint stands for the precedence network and is based on forward calculations. It is assumed that using “as soon as possible” approach is desirable here (Long and Ohsato 2008). The third constraint ensures that the required resource of type k at day t is less than or equal to the resource availability of type k at day t .

There have been numerous exact and heuristic procedures and solutions presented in the literature for the RCPSP models. Contrary to the RCPSP and MRCPSP (Multi-mode RCPSP), the literature on the DTRTP and MDTRTP (Multiple DTRTP) is sparse (Ranjbar et al. 2009). Given the NP-hardness of the RCPSP (Blazewicz et al. 1983), there is not any certain solution procedure to optimally solve these types of problems in polynomial time. As a generalization of the RCPSP, the MRCPSP is strongly NP-hard (Kolisch 2013). Demeulemeester and Herroelen (2000) indicate that the DTRTP is strongly NP-hard. As a generalization of the DTRTP, MDTRTP is also strongly NP-hard.

In this study, a Genetic Algorithm metaheuristic procedure is used to solve the DTRTP model in the Matlab Genetic Algorithm Toolbox. $TR^m(j, k)$ (obtained from the previous step), $D^l(j)$, $D^u(j)$, $ES(j)$, $LS(j)$ and predecessors relationship are the inputs, and the assigned start times ($AS(j)$) and durations of activities ($D^m(j)$) are the outputs of the RCPSP model.

3.5. Step 4. Sizing the project buffer and feeding buffers

3.5.1. Buffer sizing importance

In CCPM, duration of activities is the mean estimate (or about 50% confidence estimate) and thus, certain activities could not be completed on time and face delay. Reasons such as uncertainty, environmental condition, and other undesirable effects can engender such

conditions. This method proposes project buffer to deal with such problems. Buffer is used for activities that fail to finish their tasks on time. This is an important step in the CCPM methodology to determine the size of buffers.

3.5.2. Existing buffer sizing methods

There are different buffer sizing methods proposed in the literature with specific features. Here, a brief description for some of these identified methods is provided along with their advantages and limitations (Geekie & Steyn 2008; Kokoskie 2001).

3.5.2.1. The cut and paste method

The Cut and Paste Method (C&PM) sets the buffer size equal to one-half of the duration of the longest chain feeding into the buffer. In this method, critical and feeding chains are computed through 50% of the safe estimates as task durations. Following the determination of the critical chain, the next step is to sum and then use half of this sum as feeding buffer. The feeding buffers are added to the end of the feeding chain where the feeding chain merges with the critical chain. The advantages of the C&PM are its simplicity to apply and providing a large enough buffer. The disadvantage of the method is that the size of the buffer increases linearly with the length of the feeding chain.

3.5.2.2. The Square Root of the Sum of the Squares

The Square Root of the Sum of the Squares (SSQ) method (also known as the Root Square Error Method (RSEM)) uses information of mean duration for each task in the chain. In SSQ, size of the buffer is set to the sum of the squares of the differences between the low risk duration

and the mean duration for each task along the longest chain or the largest result considering each chain (Leach 2005; New bold 1998):

$$PB = 2 \times \sqrt{\left(\frac{S_1 - A_1}{2}\right)^2 + \left(\frac{S_2 - A_2}{2}\right)^2 + \dots + \left(\frac{S_n - A_n}{2}\right)^2} = \left[\sum_{i=1}^n (S_i - A_i)^2 \right]^{\frac{1}{2}} \quad (19)$$

Where:

- PB is project buffer;
- n is the number of all activities in the critical chain;
- S_i is high confidence (safe) estimate of activity i ;
- A_i is 50:50 (average) estimate of activity i .

The SSQ approach allows known task variation in task duration to take into account, though, it may lead to undersized buffers for long chains.

3.5.2.3. The Adaptive procedure with density

The Adaptive procedure with density (APD) method proposed by Tukel et al. (2006) takes the density of the sub-networks feeding the buffer into account. Buffers are sized according to these sub-network density values to assure that a higher number of precedence relations leads to a bigger buffer size to enhance their power to absorb potential delays. Let $\sigma_{feeding-path}$ be the standard deviation of the feeding chain, then the feeding buffer size is calculated as:

$$\text{Buffer size} = K \times \sigma_{feeding-path} \quad (20)$$

$$K = 1 + \frac{TOTPRE}{NUMTASK} \quad (21)$$

Where:

- *TOTPRE* is the total number of precedence relationships on the sub-network under consideration;
- *NUMTASK* is the total number of tasks on the sub-network.

3.5.2.4. Fuzzy buffer sizing method

Proposed by Shi and Gong (2009), Fuzzy buffer sizing method (FBSM) deals with resource constraints and fuzzy uncertainties in project management. It considers coefficients to integrate three uncertainties of resource provision, network complexity and risk preference of project manager to optimize the buffer configuration. These coefficients are defined as follows:

$$(I) a_i = \max \left\{ \sum_{k=1}^m \frac{r_{kt}}{R_t} \right\} \quad t \in [st_i, st_i + D_i] \quad (22)$$

Where:

- R_t : A resource with an upper limit quantity in the period of time t ;
- r_{kt} : Total resource quantity of activity k needed in the period of time t ;
- m : Number of activities in time t ;
- st_i : Start time of activity i ;
- D_i : Duration of activity i .

Based on the Figure 5, $A(a_i)$ coefficient is obtained as follows:

$$A(a_i) = \begin{cases} 0, & a_i \in (0, a] \\ \frac{g}{b-a}(a_i - a), & a_i \in (a, b] \\ \frac{1-g}{c-b}(a_i - b), & a_i \in (b, c] \\ 1, & a_i \in (c, 1] \end{cases} \quad (23)$$

Figure 5. Fuzzy number in three-point representation.

In the Equation (23), $A(a_i)$ is used to depict the tightness of resource consuming, and the value of each parameters ($a, b, c, g \in (0,1)$) is decided regarding the environmental factors of the project, characteristics and experiences of project manager and team member. If $a_i \leq a$, it means that doing correlative activities would not affect each other. When $a \leq a_i \leq b$, the resource consumption is a little tense and the project is implemented cautiously and smoothly. As for $b \leq a_i \leq c$, the resource using is strenuous and resource transferring is effective but in higher risk. When $c < a_i$, resource is highly consumed and doing correlative activities will affect each other resulting in project delay (Shi & Gong 2009). With a positive correlation between delay probability and the number of precedence relationships, precedence relationships of tasks indicating the network complexity is defined as:

$$(II) \beta_i = 1 + \frac{N_p - 1}{N_t} \quad (24)$$

Where β_i is the network complexity for the activity i , N_p is the total number of precedence relationships of activity i on the critical chain, and N_t is the total number of tasks on the chain.

Suppose that μ_i is average completion time, τ_i and $\varphi(x)$ are the risk preference and the cumulative standard normal distribution, respectively, $(1-\varepsilon)$ is the accomplishment probability chosen by project leader, and $(1-\varepsilon')$ as benchmark of accomplish probability. The risk coefficient is:

$$(III) \tau_i = \frac{\varphi(1-\varepsilon) - \mu_i}{\varphi(1-\varepsilon') - \mu_i} \quad (25)$$

Finally, the project buffer is described as (Shi & Gong 2009):

$$PB = \left[\sum \left[(1 + A(a_i)) \beta_i \tau_i \sigma_{iy} \right]^2 \right]^{\frac{1}{2}} \quad (26)$$

Where σ_{iy}^2 is the variation of all activities in the critical chain and PB stands for the project buffer. This method utilizes risk coefficient and resource constraints and probability of accomplishment of activities' duration, but it does not use activities' average and safety duration. Besides, finding accomplishment probability and benchmark of activities' duration cannot be accurately performed.

3.5.3. The proposed modified method for buffer sizing

The basic formula of this modified method relies on the SSQ method, although each statement of SSQ is multiplied by correction coefficients. The proposed CCPM method as foresaid, uses fuzzy numbers for required resources to overcome uncertainty. Based on this rule which declares more variations in fuzzy numbers to encompass more uncertainties, the modified method considers an appropriate coefficient. Accordingly, if an activity needs a resource with a high variation fuzzy number, it needs a buffer size of more than an activity with a low variation one. This coefficient should have two important properties. First, for constant non-fuzzy numbers, it must be equal to one. Second, it ought to increase as the fuzzy variation rises.

Consider $\tilde{A}(x)$ as a fuzzy number with membership function $\mu_{\tilde{A}}(x)$ which has a distribution versus $supp(x)$ (where $supp(x) = \{x \mid \mu_{\tilde{A}}(x) \geq 0\}$). This distribution contains a mean and a variation. Lee and Li (1988) propose the use of mean and standard deviation based on the probability measures of fuzzy events to rank fuzzy numbers. According to this approach, Cheng (1998) introduced an index CV , and RD_i can be defined based on this concept:

$$(I) RD_i = 1 + CV_i = 1 + \frac{SD_i}{TR_{e_i}} \quad (27)$$

Where:

- CV_i is the coefficient of variation for the fuzzy number i ;
- SD_i is standard deviation of the fuzzy number related to the required resource of activity i ;
- TR_{e_i} is the mean of the fuzzy number related to the required resource of activity i ;
- RD_i is the scatter index for i 'th fuzzy number. It is equal to one when SD_i is equal to zero.

RD_i increases as SD_i increases.

The AI index is applied to calculate SD_i and TR_{e_i} instead of using Lee and Li's formulation (1988). In this method, $TR^x(i)$ is computed through Equation (6) for an adequate number of AI s (e.g., $AI = 0, 0.01, 0.02, \dots, 1$). Subsequently, the mean and standard deviation of these obtained $TR^x(i)$ as the SD_i and TR_{e_i} of the fuzzy numbers can be calculated. That is, each fuzzy number is converted to the probabilistic distributions and their mean and standard deviation is calculated:

$$a_i = \max \left\{ \sum_{k=1}^m \frac{r_{kt}}{R_t} \right\}, t \in [st_i, st_i + D] \quad (28)$$

$$(II) A(a_i) = \begin{cases} 0, & a_i \in (0, a] \\ \frac{g}{b-a}(a_i - a), & a_i \in (a, b] \\ \frac{1-g}{c-b}(a_i - b), & a_i \in (b, c] \\ 1, & a_i \in (c, 1] \end{cases} \quad (29)$$

$$(III) \beta_i = 1 + \frac{N_p - 1}{N_i} \quad (30)$$

$$(IV) \sigma_i = \frac{U_i}{2} = \frac{D^h(i) - D^m(i)}{2} \quad (31)$$

Where $D^m(i)$ s are the outputs of step 3 and $D^h(i)$ is computed based on the $TR^h(i)$. As experienced in many projects, TR/D (units per day) duration is constant unless there is a new management policy. So it is formulated as:

$$\frac{TR^h(i)}{D^h(i)} = \frac{TR^m(i)}{D^m(i)} \quad (32)$$

$D^h(i)$ can be calculated by Equation (32) as below.

$$D^h(i) = \frac{TR^h(i)}{TR^m(i) / D^m(i)} \quad (33)$$

Finally, the project buffer is defined as Equation (34).

$$PB = 2 \left[\sum_{i=1}^n \left[(1 + A(a_i)) \beta_i RD_i \sigma_i \right]^2 \right]^{\frac{1}{2}} \quad (34)$$

If $\sigma_i = 0$, $a_i = 1$, $RD_i = 1$ and $N_p = 1$, then the answer of the method is equal to SSQ. That is, the proposed framework is similar to the SSQ method, but does not consider each activity like the other. A key advantage of this method lies where it can be adopted for feeding buffer since all steps can be used and calculated for feeding chains.

Finally, the project buffer for a specific project is equal to maximum PB on different critical chains.

3.6. Step 5. Controlling the project schedule by the fuzzy CCPM approach

Following construction of the complete schedule, the project can be run and from the further execution reports, this schedule is updated. Updating can be performed through reusing the proposed methodology steps for the remaining works of each activity. As a result, the remaining

duration and work per time unit is updated and the project buffer is resized. Interruptions may occur in the updates.

4. Numerical Example

A numerical illustration is conducted in this section to demonstrate the performance of the proposed buffer sizing methodology using a project network with seven activities. Results are then compared with the SSQ method. Activities and their predecessors' relationship are listed in [Table 1](#), and the project network is displayed in [Figure 6](#). $D^l(i)$ and $D^u(i)$ are the lower and upper bounds for duration of activity i (illustrated in column 2 and 3 of [Table 1](#)).

Table 1. Input data for the numerical example.

4.1. Step1

Experts are asked to estimate the amount of resource that is needed for each activity to be completed. According to the proposed methodology, estimations are trapezoidal fuzzy numbers. $TR(j,k) = (a,b,c,d)$ is the estimate of total required resource k to perform task j (given in [Table 1](#)) and daily resource availability is 85 men per day.

Figure 6. A numerical example of the precedence network.

4.2. Step 2

Using Equation (6), $TR(j, k)$ is defuzzified to a 95% and 50% point ($TR^h(i)$ and $TR^m(i)$). The results are rounded subsequently (provided in columns 11 and 12 of [Table 1](#)). As an example, calculation of 50% point for the first activity is as the follows.

$$TR^m(1) = 340 - \sqrt{(340 - 289) \times (289 - 275 + 340 - 216) \times (1 - 0.5)} = 280.6788$$

4.3. Step 3

$TR^m(i)$, $D^l(i)$, $D^u(i)$, $ES(i)$ and $LS(i)$ are the inputs of the RCPSP model described in section 3.4. Inputs such as objective function, linear and nonlinear constraint are prepared in M-files. Genetic Algorithm Toolbox is applied using the following setting: generation number =1000, population size=150, crossover rate= 0.8 and mutation rate= 0.01. Running the model using Intel® core™ i5 CPU, 4 GB RAM lasts about 10s. $AS(i)$ (assign start of activity i) and $D(i)$ (duration of activity i) are the outputs of this model. These results are provided in [Table 2](#). From [Table 2](#), it is evident that the total duration of the project is $17 + 3.8 = 20.8$ days.

Table 2. The output of RCPSP model.

Next, a Gantt chart based on the durations is drawn ([Figure 7](#)). Critical chain could be found given the resources. As is displayed in [Figure 7](#), there is only one critical chain in this project: #1-#3-#6-#7.

Figure 7. A numerical example of the Gantt chart.

4.4. Step 4

The project buffer size is calculated in step 4 of the proposed method. First, a_i is computed through Equation (28). For the first activity as an example, $a_1 = 82.65 / 85 = 0.97$. Then, $A(a_i)$ is calculated by Equation (29) (let $a=0.5$, $b=0.7$, $c=0.9$, $g=0.4$ used by Shi and Gong 2009). As $a_i \in (0.9, 1]$, then the membership function ($A(a_i)$) is equal to 1. The other results are available in Table 3. Next, β_i is computed via Equation (30). For instance, since the first activity has no predecessor, then $\beta_1 = 1 + \frac{(0-1)}{4} = 0.75$ (Table 3). R_i is calculated according to the distribution of the fuzzy number versus $Supp(w)$. For the first activity: $E_1 = 322.1645$ and $std_1 = 21.06588$, thus $RD_1 = 1.07$. Calculations are represented in Table 4. To compute U_i , first $D^h(i)$ is obtained using Equation (33).

$$D^h(1) = \frac{TR^h(1)}{TR^m(1) / D^m(1)} = \frac{332}{82.64} = 4.01$$

Through Equation (31), $\sigma_i = \frac{D^h(1) - D^m(1)}{2} = \frac{4 - 3.4}{2} = 0.3$. Other results are presented in

Table 3. Based on these calculations and by Equation (34), project buffer can be concluded. Here, there is no need to calculate the maximum PBs as there is just one critical chain.

Table 3. The calculated coefficient and project buffer.

Table 4. The RD_i calculation coefficient.

Finally, project buffer is added at the end of the project schedule (Figure 7).

4.5. Step 5

To start the project by project managers, all activities' start time and durations are determined. Based on CCPM, 50% of activities finish on time while the other half face delay, thus, the project buffer is consumed whenever necessary. The maximum total duration of the project is $20.8 + 6.1 = 26.9$.

After running the activities in a specific time that managers decide, the project at status date can be updated. These updates help managers to have a real feedback from the project duration.

The final result of the procedure can be compared to the SSQ approach to show the efficiency of the proposed method (Table 5). The project buffer for the developed project is 6.1 whereas for SSQ method, project buffer is equal to 4. As it was stated, the minimum amount of buffer should be 25% critical chain duration. The outcome of the proposed methodology is thus better than the SSQ method. Moreover, SSQ and other similar methods do not take into account task risk, network density, resource usage or any corrective coefficient.

Table 5. Comparison of the buffer sizing methods.

5. Conclusion

There are various risk-creating factors surrounding a project which result in delay and disruption on projects. To avoid these undesirable consequences and schedule a project successfully, this research proposed a critical chain based methodology.

In CCPM, it is necessary to find the average estimated duration for each task that needs to be scheduled. However, executing the activities in this short duration is arduous and usually faces some respites. Fuzzy sets can deal with such a challenge since it has the ability to effectively

handle imprecise numbers. Thus in the proposed method, the work needed to accomplish activities was depicted by fuzzy numbers. By linking fuzzy sets and probability theory, the inputs of a RCPSP model were prepared. The average duration and assigned start of each task obtained from the RCPSP model were used in the model. It was assumed that using “as soon as possible” approach is desirable.

Project buffer was applied to consume when necessary, since the mean estimate (risky) duration was used for planning the activities and regarding CCPM concepts. The proposed method for sizing the project buffer as an important step relied on the SSQ method which was modified by coefficients to overcome the schedule risk more efficiently. This procedure can be reused in further project updates to reschedule the project; and consequently, utilized as a project controlling procedure.

Future researchers are encouraged to propose other methods considering further kinds of buffers, such as resource buffers, cost buffers, etc. The proposed method can also be modified to be used in an upper level, i.e., enterprise project management.

References

- Ahlemann F, El Arbi, F, Kaiser MG, Heck A. 2013. A process framework for theoretically grounded prescriptive research in the project management field. *International Journal of Project Management*. 31(1): 43–56.
- Barrios A, Ballestín F, Valls V. 2011. A double genetic algorithm for the MRCPSP/max. *Computers & Operations Research*. 38(1): 33–43.
- Bevilacqua M, Ciarapica FE, Giacchetta G. 2009. Critical chain and risk analysis applied to high – risk industry maintenance: A case study. *International Journal of Project Management*. 27(4): 419–432.
- Blazewicz j, Lenstra JK, Kan, AR. 1983. Scheduling subject to resource constraints: Classification and complexity. *Discrete Applied Mathematics*. 5(1): 11–24.
- Bouleimen KLEIN, Lecocq HOUSNI. 2003. A new efficient simulated annealing algorithm for the resource – constrained project scheduling problem and its multiple mode version. *European Journal of Operational Research*. 149(2): 268–281.

- Brucker P, Knust S, Schoo A, Thiele O. 1998. A branch and bound algorithm for the resource constrained project scheduling problem. *European Journal of Operational Research*. 107(2): 272–288.
- Chen PH, Weng H. 2009. A two – phase GA model for resource – constrained project scheduling. *Automation in Construction*. 18(4): 485–498.
- Chen RM. 2011. Particle swarm optimization with justification and designed mechanisms for resource – constrained project scheduling problem. *Expert Systems with Applications*. 38(6): 7102-7111.
- Cheng CH. 1998. A new approach for ranking fuzzy numbers by distance method. *Fuzzy sets and systems*. 95(3): 307–317.
- De Reyck B, Herroelen W. 1998. A branch – and – bound procedure for the resource – constrained project scheduling problem with generalized precedence relations. *European Journal of Operational Research*. 111(1): 152–174.
- Demeulemeester E, Herroelen W. 2000. The discrete time/resource trade-off problem in project networks: a branch-and-bound approach. *IIE transactions*. 32(11): 1059-1069.
- Demeulemeester E, Herroelen W. 2002. Project scheduling–A research handbook. Vol. 49 of *International Series in Operations Research & Management Science*.
- Demeulemeester EL, Herroelen WS, Elmaghraby SE. 1996. Optimal procedures for the discrete time/cost trade-off problem in project networks. *European Journal of Operational Research*. 88(1): 50-68.
- Duan Q, Liao TW. 2010. Improved ant colony optimization algorithms for determining project critical paths. *Automation in construction*. 19(6): 676-693.
- Fang C, Wang L. 2012. An effective shuffled frog-leaping algorithm for resource-constrained project scheduling problem. *Computers & Operations Research*. 39(5): 890-901.
- Geekie A, Steyn H. 2008. Buffer sizing for the critical chain project management method. *South African Journal of Industrial Engineering*. 19(1): 73-88.
- Ghaffari M, Emsley MW. 2015. Current status and future potential of the research on Critical Chain Project Management. *Surveys in Operations Research and Management Science*. 20(2): 43-54.
- Goldratt EM, Cox J. 1984. *The Goal*. The North River Press, Great Barrington, MA, USA.
- Goldratt EM. 1997. *Critical Chain: A Business Novel*. The north River Press, Great Barrington, MA, USA.
- Hall NG. 2015. Further Research Opportunities in Project Management. In *Handbook on Project Management and Scheduling Vol. 2* (pp. 945-970). Springer International Publishing.
- Heilmann R. 2003. A branch-and-bound procedure for the multi-mode resource-constrained project scheduling problem with minimum and maximum time lags. *European Journal of Operational Research*. 144(2): 384–365.
- Horroelen W, Leus R. 2001. On the merit and pitfalls of critical chain scheduling. *Journal of Operations Management*. 19(5): 559–577.

- Kaufman A, Gupta M. 1985. Introduction to fuzzy arithmetic: Theory and applications. New York: Van Nostrand Reinhold Company.
- Klein R. 1999. Scheduling of resource-constrained projects (Vol. 10). Springer Science & Business Media.
- Kokoskie G. 2001. A Comparison of Critical Chain Project Management (CCPM) Buffer Sizing Techniques. GEORGE MASON UNIV FAIRFAX VA.
- Kolisch R. 2013. Project scheduling under resource constraints: efficient heuristics for several problem classes. Springer Science & Business Media.
- Lawrence K. 1988. Resource-Constrained Project Scheduling with Preemption of Jobs. PhD diss., University of Michigan.
- Leach LP 2014. Critical Chain Project Management. Artech House Inc., Boston.
- Leach LP. 2005. Critical Chain Project Management. Artech House Inc., Norwood, MA.
- Lee ES, Li RJ. 1988. Comparison of fuzzy numbers based on the probability measure of fuzzy events. Computers & Mathematics with Applications. 15(10): 887–896.
- Long LD, Ohsato A. 2008. Fuzzy critical chain method for project scheduling under resource constraints and uncertainty. International Journal of Project Management. 26(6): 688–698.
- Mingozzi A, Maniezzo V, Ricciardelli S, Bianco L. 1998. An Exact Algorithm for the Resource Constrained Project Scheduling Problem Based on a New Mathematical Formulation. Management Science. 44(5): 714–729.
- Newbold RC. 1998. Project management in the fast lane: applying the theory of constraints. St. Lucie Press, New York.
- Olaguibel RAV, Goerlich JT. 1993. The project scheduling polyhedron: dimension, facets and lifting theorems. European Journal of Operational Research. 67(2): 204-220.
- Pan NH, Hsaio PW, Chen KY. 2008. A study of project scheduling optimization using Tabu Search algorithm. Engineering Applications of Artificial Intelligence. 21(7): 1101–1112.
- PMI. 2008. A guide to the Project Management Body Of Knowledge (PMBOK®). Project Management Institute, Inc, Pennsylvania.
- Pritsker AAB, Watters LJ, Wolfe PM. 1969. Multi project scheduling with limited resources: A zero – one programming approach. Management Science. 16(1): 93–108.
- Rand GK. 2000. Critical chain: the theory of constraints applied to project management. International Journal of Project Management. 18(3): 173–177.
- Ranjbar M, De Reyck B, Kianfar F. 2009. A hybrid scatter search for the discrete time/ resource trade-off problem in project scheduling. European Journal of Operational Research. 193(1): 35–48.
- Roghalian E, Alipour M. 2014. A fuzzy model for achieving lean attributes for competitive advantages development using AHP-QFD-PROMETHEE. Journal of Industrial Engineering International. 10(3): 1-11.

Shi Q, Gong T. 2009. An improved project buffer buffer sizing approach to critical chain management under resources constraints and fuzzy uncertainty. International Conference on Artificial Intelligence and computational Intelligence.

Tukel OI, Rom WO, Eksioglu SD. 2006. An investigation of buffer sizing techniques in critical chain scheduling. European Journal of Operational Research. 172(2): 401–416.

Valls V, Ballestin F, Quintanilla S. 2008. A hybrid genetic algorithm for the resource – constrained project scheduling problem. European Journal of Operational Research. 185(2): 495–508.

Yang S, Fu L. 2014. Critical chain and evidence reasoning applied to multi-project resource schedule in automobile R&D process. International Journal of Project Management. 32(1): 166-177.

Zhang H, Li X, Li H, Huang F. 2005. Particle swarm optimization – based schemes for resource – constrained project scheduling. Automation in Construction. 14(3): 393–404.

Table 1. Input data for the numerical example.

Activity	$D^l(j)$	$D^u(j)$	Predecessors	$\bar{TR}(j,k)$				TR^m	TR^h	TR^m	TR^h
	(Days)	(Days)		a	b	c	d	Real	Real	Rounded	Rounded
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)
1	2	8	-	216	275	289	340	280.6788	331.6107	281	332
2	1	7	1	92	141	155	214	150.6596	205.0423	151	205
3	3	9	1	223	296	310	376	301.7638	365.5014	302	366
4	1	8	2,3	184	200	214	267	216.2999	259.8299	216	260
5	1	5	2,4	56	88	102	145	97.9415	138.3449	98	138
6	2	8	3	134	191	205	271	200.4096	261.0170	200	261
7	3	9	5,6	263	303	325	380	318.1736	371.2564	318	371

Table 2. The output of RCPSP model.

Act	AS	D	TR/D
1	0	3.4	82.65
2	4.2	6.9	21.88
3	3.4	5.6	53.93
4	11.1	4	54.00
5	15.1	1.9	51.58
6	9	8	25.00
7	17	3.75	83.68

Table 3. The calculated coefficient and project buffer.

Act	α_i	$A(\alpha_i)$	β_i	RD_i	σ_i	PB
1	0.97	1	0.75	1.0710	0.3	6.1
2	0.89	0.5757	-	1.1394	1.25	
3	0.89	0.5757	1	1.0825	0.6	
4	0.93	1	-	1.0786	0.4	
5	0.90	0.6027	-	1.1587	0.4	
6	0.93	1	1	1.1173	1.2	
7	0.98	1	1.25	1.0653	0.35	

Table 4. The RD_i calculation coefficient.

P(TR)	TR	P(TR)	TR	P(TR)	TR	P(TR)	TR	P(TR)	TR
0.01	293.00	0.21	302.29	0.41	312.84	0.61	325.40	0.81	341.89
0.02	293.44	0.22	302.78	0.42	313.41	0.62	326.10	0.82	342.90
0.03	293.89	0.23	303.28	0.43	313.99	0.63	326.81	0.83	343.95
0.04	294.33	0.24	303.78	0.44	314.57	0.64	327.54	0.84	345.03
0.05	294.78	0.25	304.28	0.45	315.16	0.65	328.27	0.85	346.14
0.06	295.23	0.26	304.78	0.46	315.75	0.66	329.02	0.86	347.28
0.07	295.68	0.27	305.29	0.47	316.35	0.67	329.77	0.87	348.47
0.08	296.13	0.28	305.81	0.48	316.95	0.68	330.54	0.88	349.71
0.09	296.59	0.29	306.33	0.49	317.56	0.69	331.32	0.89	351.00
0.10	297.05	0.30	306.85	0.50	318.17	0.70	332.11	0.90	352.35
0.11	297.51	0.31	307.37	0.51	318.80	0.71	332.91	0.91	353.77
0.12	297.98	0.32	307.90	0.52	319.42	0.72	333.73	0.92	355.27
0.13	298.45	0.33	308.43	0.53	320.06	0.73	334.57	0.93	356.87
0.14	298.92	0.34	308.97	0.54	320.70	0.74	335.42	0.94	358.58
0.15	299.39	0.35	309.51	0.55	321.35	0.75	336.28	0.95	360.45
0.16	299.86	0.36	310.05	0.56	322.00	0.76	337.17	0.96	362.51
0.17	300.34	0.37	310.60	0.57	322.66	0.77	338.07	0.97	364.86
0.18	300.82	0.38	311.15	0.58	323.34	0.78	338.99	0.98	367.63
0.19	301.31	0.39	311.71	0.59	324.01	0.79	339.93	0.99	371.26
0.20	301.80	0.40	312.27	0.60	324.70	0.80	340.90	1.00	380.00
TRe1: 322.16			STD1: 21.07			CV1: 0.07			

Table 5. Comparison of the buffer sizing methodology.

Methodology	Proposed method	SSQ
Project buffer size	6.1	4

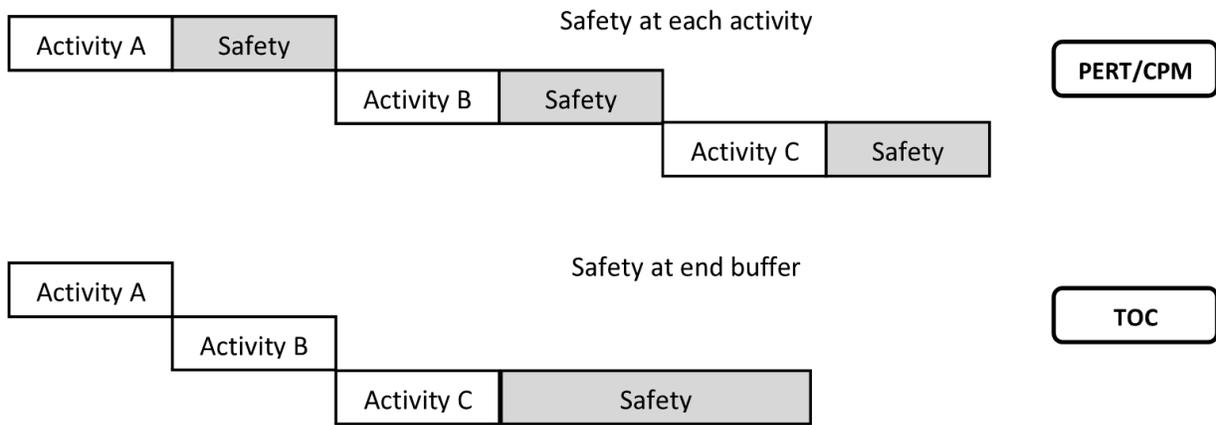


Figure 1. PERT/CPM and TOC comparison.

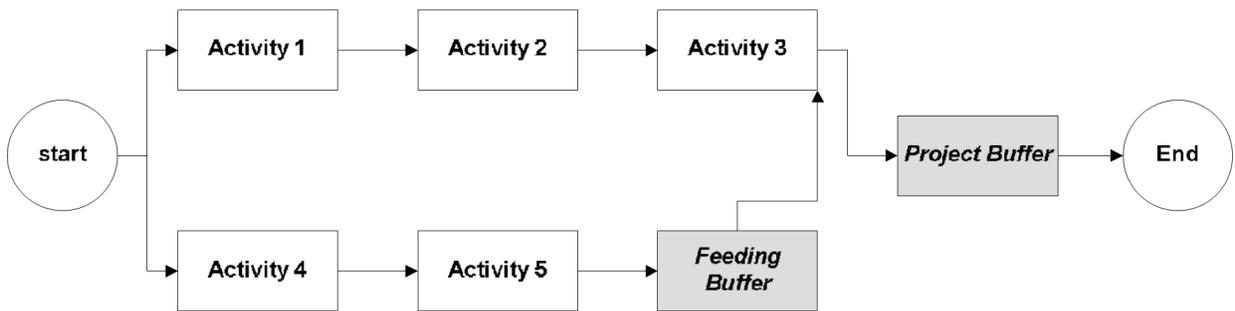


Figure 2. Project buffer and Feeding buffer.

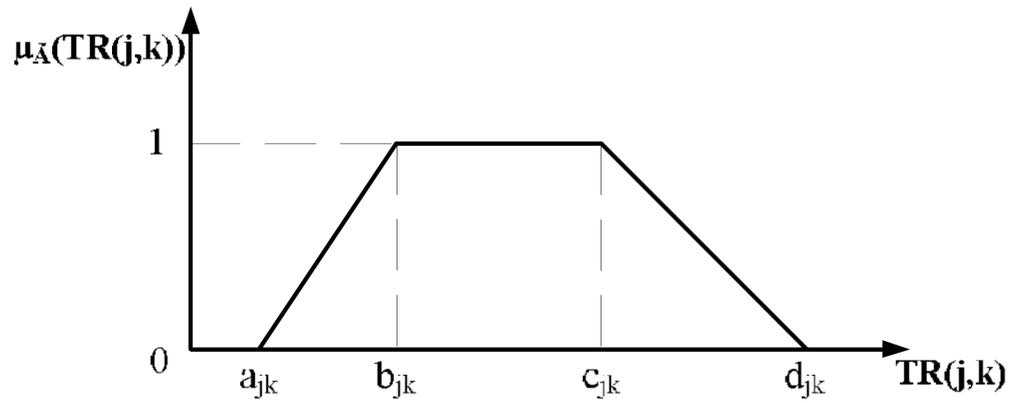


Figure 3. The membership function of the fuzzy number of required resource k for task j .

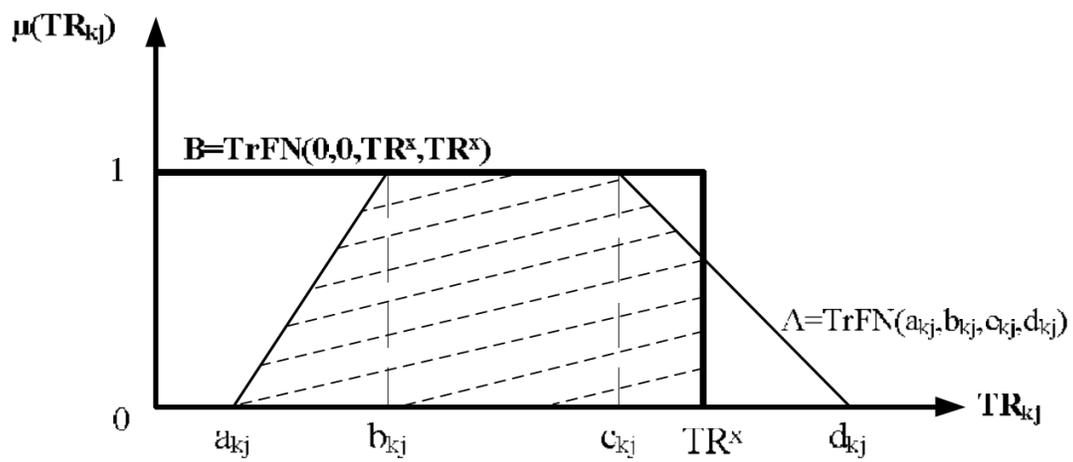


Figure 4. High agreement index (AI) for required resource k for task j .

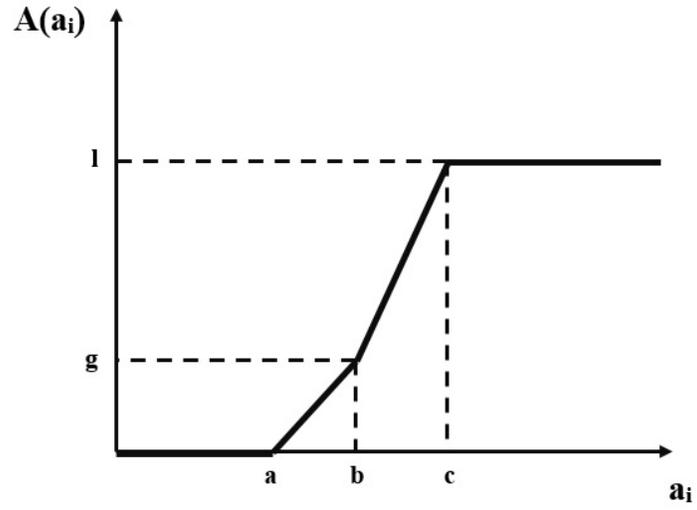


Figure 5. Project buffer and feeding buffer.

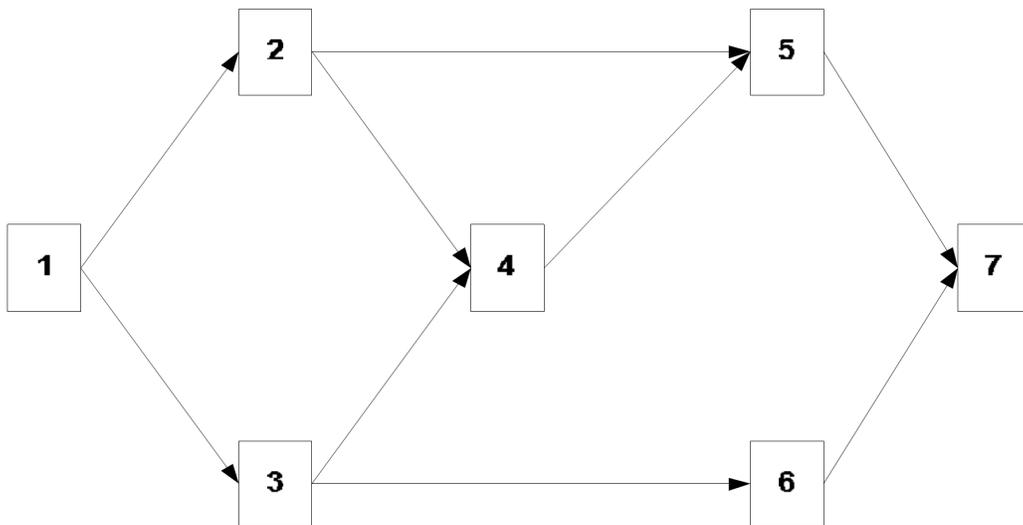


Figure 6. A numerical example of the precedence network.

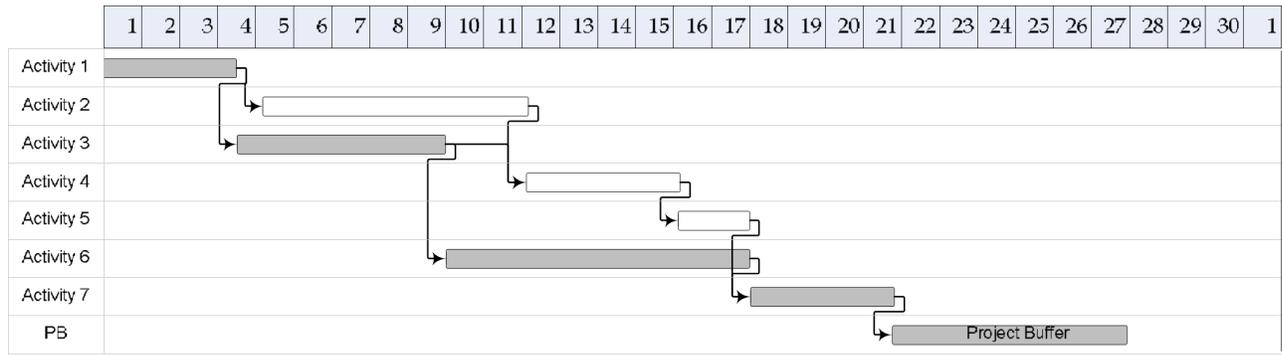


Figure 7. A numerical example of the Gantt chart.