Improving Dijkstra's algorithm for Estimating Project Characteristics and Critical Path

Adilakshmi Siripurapu[1], Ravi Shankar Nowpada[2], K. Srinivasa Rao[3]

•

Dept. of Basic Science and Humanities, Vignan's Institute of Information Technology (A), Duvvada, Visakhapatnam, AP, India[1]
Dept. of Mathematics, Institute of Science, GITAM (Deemed to be University), Visakhapatnam, AP, India[2]
Dept. of Operations, GITAM school of Business, GITAM (Deemed to be University), Visakhapatnam, AP, India[3]

laxmimaths2008@gmail.com[1]
Drravi68@gmail.com[2]
skolli2@gitam.edu

Abstract

Developing a project planning structure for all industries is a technological challenge involving evaluating several restrictions for each activity's respective task and its planning tools. Any restriction affects the completion time, operating costs, and overall project performance. Programme Evaluation Review Technique (PERT) and Critical Path Method (CPM) processes made many researchers study the possible ways of finding the critical paths and activities in the network. The advancement of the CPM and PERT towards a probabilistic environment is still a long way off. However, Artificial intelligence approaches such as the Genetic Algorithm, Dijkstra's algorithm, and others are utilized for network analysis within the project management framework. This study is to help the project manager plan schedule for a construction project to determine the expected completion time. In this research paper, we describe a method for obtaining the earliest and latest times of a critical path using modified Dijkstra's algorithm with triangular fuzzy numbers. Forward pass and backward pass algorithms are designed to find the optimal path for the proposed method. Numerical examples are also illustrated for the same. Simulation results are included by the use of the "C" program. Finally, a comparison is made with the traditional method PERT.

Keywords: Critical Path, Dijkstra's Algorithm, Earliest and latest times, modified Dijkstra's algorithm, PERT.

1. Introduction

In the project network development process, the project controller's objective is to develop a primary plan. The critical path method is one of the most significant approaches in network study. The concept of the critical path allows the decision-maker to control the project's cost and schedule, and it can improve the quality of the work. This method is commonly used in various industries to analyze and improve the efficiency of a project. Many cases have been discussed where the activity times are deterministic; the PERT method applies to a probabilistic environment. Different methods and various working techniques are applied in project management. Every procedure has its own time to complete the task. Gantt chart, network diagram, CPM and PERT are a few strategies commonly used to handle projects.

Every activity's length is considered when estimating the longest path in CPM. In PERT, the activity's estimated time is assumed. Due to that, PERT handles by taking three-point estimates (most likely [m], optimistic [a], and pessimistic [b] and if the activity allocation fits the beta distribution. Malcom first proposed a PERT approximation using the Beta distribution in 1959 [7]. An efficient critical path analysis algorithm (CHAN) based on the automatic test pattern generation (ATPG) approach PODEM was presented by Chang in 1993[1]. Traditionally, the beta distribution was used in the PERT. Solomon Sackey et al., 2018 proposed that an altered PERT was enlarged and utilized to model scheduling risk. The proposed PERT model was based on suspicion of a 95% certainty level. According to the due date and the probability of lag, the project completion probability is computed in five perspectives for both approaches. A sample is taken to assess the error rate for every example. The average error rate was calculated using the traditional PERT technique and the updated PERT method for all cases. The revised PERT version improved the average error rate by 2.46%, correlated with 3.31% of the traditional PERT approach. This way of considering has confirmed that the revised PERT approach can more precisely assess the completion probability better than the traditional PERT. By the way, because the new PERT was entirely based on suspicion, it is ambitious to decide with confidence that it is far superior to the conventional PERT model [8]. Li et al., 2007; The Monte Carlo simulation studied the real-world project's network program. The outcome revealed that the stochastic network program gave significant scheduling data better than the conventional network program [4]. Lee, 2005 proposed Stochastic Project Scheduling Simulation (SPSS) software approach affects the possibility of finishing a project with a deadline set by the software's operator. The SPSS program can simulate activity time using several probability distributions and uniform, triangular, and normal distributions. SPSS also computes the CI for entire project activities [3]. In general, project network simulation is utilized to improve the feasibility and reliability of the PERT study. Cheng et al., 2004 explained the use of MonteCarlo simulation in PERT to achieve a stochastic period of activity [2]. A simulation model that generates a 20 histogram for the distribution of the completed activity network. CPM/PERT simulation technique that adds discrete event simulation plan technique and the critical path determination process. According to the authors, "for each activity, Earliest start ($E\tilde{S}_{ij}$), Latest start ($L\tilde{S}_{ij}$), Earliest finish ($E\tilde{F}_{ij}$), Latest finish ($E\tilde{F}_{ij}$), and Total float ($T\tilde{F}_{ij}$) times should be included in the CPM analysis." The $E\tilde{S}_{ij}$ and $E\tilde{F}_{ij}$ of the project network are computed during its forward pass, whereas the $L\tilde{S}_{ij}$,$L\tilde{F}_{ij}$ and $T\tilde{F}_{ij}$ are determined throughout its backward pass. The $T\tilde{F}_{ij}$ is utilized to evaluate the project's criticality by Lu et al., 2000[5]. Mac Crimmon et al., 1964; one drawback of the PERT is that though many pathways must follow to finish a project, the project time is reduced and does not exceed the average project time [6]. Shankar et al. 2010 used modified Dijkstra's algorithm to estimate project duration [9]. Xiaokang Han et al. proposed in 2021 an improved ant colony algorithm to determine the critical path by setting the path distance and time as negative, while the transition probability remains unchanged, is proposed [10].

In 1965, Zadeh introduced [11] the concept of fuzzy set theory. In today's highly competitive world, many problems in fuzzy mathematics have been produced. When the activity periods in a project environment are deterministic, many real-life events change faster by utilizing the idea of fuzziness.

In this paper, with the help of the program evaluation review technique (PERT) and critical path method (CPM) along with Dijkstra's algorithm approached with an example to formulate the critical path and project duration. The main objective values are taken in fuzzy numbers; we can rank the fuzzy number to find the best alternative.

2. Methodology

2.1 Proposal Algorithm

An altered Dijkstra's Algorithm determines the maximum time between a start node (referred to as the "source node") and further nodes in a network. In this technique, the weights of the edges are utilized to determine the path that optimizes the overall distance (weight) within the start node and further nodes. Modified Dijkstra's Algorithm is only suitable for positive weighted graphs because the weights of the edges must be added during the procedure to determine the longest path.

Basic concepts in an altered Dijkstra's Algorithm

- An altered Dijkstra's Algorithm starts at the node you choose (the source node) and analyzes the graph to identify the longest path within that node and all further nodes in the network.
- The model considers the currently known longest path within an individual node and the origin node, and it modifies the values if the longest path is identified.
- The model finds the maximum distance from one event to another event; the node is labeled as "visited" and adds to the path.
- The procedure is continued till all the nodes in the graph are connected to the path. In the process, we have a path that adds the source node to all further nodes by taking the longest possible path to an individual node.
- The source node is at zero distance from itself. Initially gives '0' labels to all vertices.
- Use the infinity sign to indicate the distance from the source node to all other nodes for the time being because it has not yet been estimated.
- We will find the earliest times in modified Dijkstra's algorithm using the forward pass algorithm and the latest times using the backward pass algorithm.

2.2 Forward pass calculations in Dijkstra's algorithm

Step1: In sequence$v_1 = 1, v_2 = 2, \ldots \ldots \ldots v_n = n$, allocate n vertices.
Step2: Assign permanent label '0' to the primary vertex $v_1 = 1$ and provisional label '0' to the rest of $n - 1$ edges.
Step3: Every vertex j that is not permanently labeled would receive a new provisional label.

$$\text{i.e, } E_j = \max\{oldlabelofj, (oldlabelofi + t_{ij})\}$$

Where i is permanently labeled with the new vertex and t_{ij} is the duration of activity between vertices i and j, if an edge is not connected to i and j,$t_{ij} = \infty$.
Step4: The next vertex turns into the fixed (visited) label.
Step3 and step4 repeated until $v_n = n$ gets a fixed label. The $E_j's$ permanently labeled values are the earliest times as $E_1 = 0$.

2.3 Backward pass calculations in Dijkstra's algorithm

Step1: Set n vertices to$v_n = n, v_{n-1} = n - 1, \ldots \ldots v_1 = 1$.
Step2: Allocate fixed label $L_n = E_n$to the vertex $v_n = n$ and temporary labels to remains $n - 1$ vertices.
Step3: Any node j that does not get a constant label gets a new provisional label.

$$\text{i.e., } L_j = \min[oldlabelofi, (oldlabelofj + t_{ij})]$$

Where j is the fixed labeled with the new vertex $t_{ij}$is the duration
of activity among vertices i and$j$
Step4: as per step1, the next vertex will become a fixed label or permanent label.
Repeated step3 and step4 until then the initial vertex $v_1 = 1$ gets a fixed label.

2.4 The proposal ranking in a Triangular fuzzy number

Let$\tilde{A} = (a, b, c)$be the Triangular fuzzy number and consider the Triangle centroid as the ranking in the Triangular fuzzy number and its diagram expressed in Figure 1.

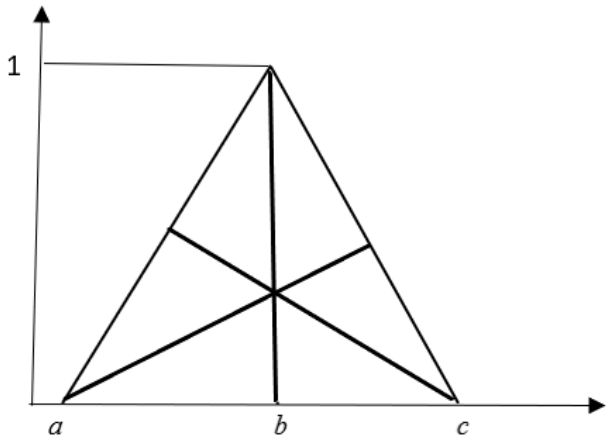


Figure 1: *Diagram representation of centroid of TFN*

The centroid of the Triangle is$\frac{a+b+c}{3}$. Consider, the centroid of triangle is a new ranking in Triangular fuzzy number.
Therefore, the new ranking in Triangular fuzzy number is;

$$\mathcal{R}(\tilde{A}) = \frac{a+b+c}{3}.$$

3 Numerical Analysis

Here I collected applications from Network sources presented in Table1. Moreover, a related network diagram is presented in Figure 2.

Table1: *Application Problem*

Activity	Code	Predecessor	*a*	*m*	*b*
1→2	P	-	5	6	7
1→3	Q	-	1	3	5
1→4	R	-	1	4	7
2→5	S	P	1	2	3
3→6	T	Q	1	2	9
4→6	U	R	1	5	9
4→7	V	R	2	2	8
6→7	W	T, U	4	4	10
5→8	X	S	2	5	8
7→8	Y	W, V	2	2	8

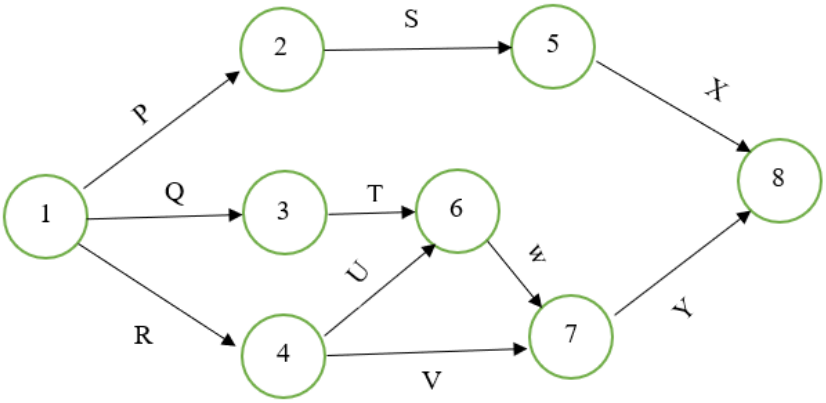


Figure 2: *Diagram representation of application problem*

3.1 Duration of activities calculated by adopting with TFN

The duration of activities taken as the Triangular fuzzy number is depicted in Table 2. Moreover, a related diagram is represented in Figure 3.

Table 2: *Expected time of activities with TFN*

Activity	*a*	*m*	*b*	*TFN*
1→2	5	6	7	(5,6,7)
1→3	1	3	5	(1,3,5)
1→4	1	4	7	(1,4,7)
2→5	1	2	3	(1,2,3)
3→6	1	2	9	(1,2,9)
4→6	1	5	9	(1,5,9)
4→7	2	2	8	(2,2,8)
6→7	4	4	10	(4,4,10)
5→8	2	5	8	(2,5,8)
7→8	2	2	8	(2,2,8)

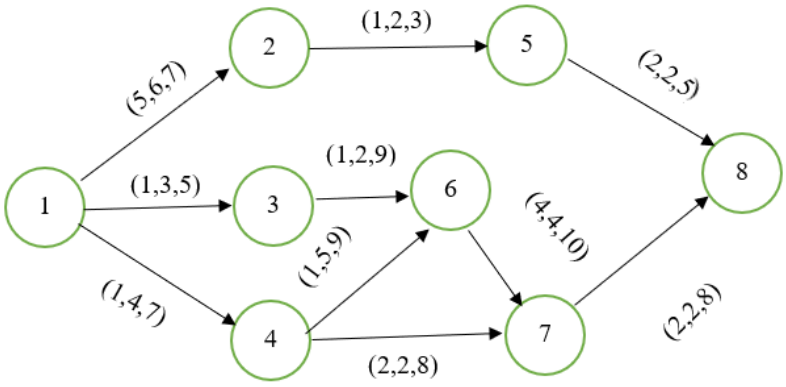


Figure3: *Activities of project network with TFN*

3.2 Earliest times

Earliest times of every node in the project network using forward pass algorithm with TFN seen in Table 3.

Table 3: *Earliest times of every node with a TFN*

Vertex Number								Earliest time
1	2	3	4	5	6	7	8	
(0,0,0)	(0.,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	(0,0,0)	$E_1 = (0,0,0)$
(0,0,0)(F)	(5,6,7)	(1,3,5)	(1,4,7)	∞	∞	∞	∞	
(0,0,0)(F)	(5,6,7)(F)	(1,3,5)	(1,4,7)	∞	∞	∞	∞	$E_2 = (5,6,7)$
(0,0,0)(F)	(5,6,7)(F)	(1,3,5)	(1,4,7)	(6,8,10)	∞	∞	∞	
(0,0,0)(F)	(5,6,7)(F)	(1,3,5)(F)	(1,4,7)	(6,8,10)	∞	∞	∞	$E_3 = (1,3,5)$
(0,0,0)(F)	(5,6,7)(F)	(1,3,5)(F)	(1,4,7)	(6,8,10)	(2,5,14)	∞	∞	
(0,0,0)(F)	(5,6,7)(F)	(1,3,5)(F)	(1,4,7)(F)	(6,8,10)	(2,5,14)	∞	∞	$E_4 = (1,4,7)$
(0,0,0)(F)	(5,6,7)(F)	(1,3,5)(F)	(1,4,7)(F)	(6,8,10)	(2,9,16)	(3,6,15)	∞	
(0,0,0)(F)	(5,6,7)(F)	(1,3,5)(F)	(1,4,7)(F)	(6,8,10)(F)	(2,9,16)	(3,6,15)	∞	$E_5 = (6,8,10)$
(0,0,0)(F)	(5,6,7)(F)	(1,3,5)(F)	(1,4,7)(F)	(6,8,10) (F)	(2,9,16)	(3,6,15)	(8,10,15)	
(0,0,0)(F)	(5,6,7)(F)	(1,3,5)(F)	(1,4,7)(F)	(6,8,10) (F)	(2,9,16)(F)	(3,6,15)	(8,10,15)	$E_6 = (2,9,16)$
(0,0,0)(F)	(5,6,7)(F)	(1,3,5)(F)	(1,4,7)(F)	(6,8,10) (F)	(2,9,16)(F)	(6,13,26)	(8,10,15)	
(0,0,0)(F)	(5,6,7)(F)	(1,3,5)(F)	(1,4,7)(F)	(6,8,10) (F)	(2,9,16)(F)	(6,13,26)(F)	(8,10,15)	$E_7 = (6,13,26))$
(0,0,0)(F)	(5,6,7)(F)	(1,3,5)(F)	(1,4,7)(F)	(6,8,10)(F)	(2,9,16)(F)	(6,13,26)(F)	(8,15,34)	
(0,0,0)(F)	(5,6,7)(F)	(1,3,5)(F)	(1,4,7)(F)	(6,8,10)(F)	(2,9,16)(F)	(6,13,26)(F)	(8,15,34)(F)	E_8= (8,15,34)

3.3 Latest times

The latest times of every node in the project network using backward pass algorithm with Triangular fuzzy ranking formula are seen in Table 4.

Table 4: ***Latest times of every node with a TFN***

Vertex number								Latest time
8	7	6	5	4	3	2	1	
(8,15,34)(F)	(8,15,34)	(8,15,34)	(8,15,34)	(8,15,34)	(8,15,34)	(8,15,34)	(8,15,34)	$L_8 = E_8$
(8,15,34)(F)	(6,13,26)	(8,15,34)	(6,13,29)	(8,15,34)	(8,15,34)	(8,15,34)	(8,15,34)	
(8,15,34)(F)	(6,13,26)(F)	(8,15,34)	(6,13,29)	(8,15,34)	(8,15,34)	(8,15,34)	(8,15,34)	$L_7 = (6,13,26))$
(8,15,34)(F)	(6,13,26)(F)	(2,9,16)	(6,13,29)	(4,11,18)	(8,15,34)	(8,15,34)	(8,15,34)	
(8,15,34)(F)	(6,13,26)(F)	(2,9,16)(F)	(6,13,29)	(4,11,18)	(8,15,34)	(8,15,34)	(8,15,34)	$L_6 = (2,9,16)$
(8,15,34)(F)	(6,13,26)(F)	(2,9,16)(F)	(6,13,29)	(1,4,7)	(1,7,7)	(8,15,34)	(8,15,34)	
(8,15,34)(F)	(6,13,26)(F)	(2,9,16)(F)	(6,13,29)(F)	(1,4,7)	(1,7,7)	(8,15,34)	(8,15,34)	$L_5 = (6,13,29)$
(8,15,34)(F)	(6,13,26)(F)	(2,9,16)(F)	(6,13,29)(F)	(1,4,7)	(1,7,7)	(5,11,26)	(8,15,34)	
(8,15,34)(F)	(6,13,26)(F)	(2,9,16)(F)	(6,13,29)(F)	(1,4,7)(F)	(1,7,7)	(5,11,26)	(8,15,34)	$L_4 = (1,4,7)$
(8,15,34)(F)	(6,13,26)(F)	(2,9,16)(F)	(6,13,29)(F)	(1,4,7)(F)	(1,7,7)	(5,11,12)	(0,0,0)	
(8,15,34)(F)	(6,13,26)(F)	(2,9,16)(F)	(6,13,29)(F)	(1,4,7)(F)	(1,7,7)(F)	(5,11,12)	(0,0,0)	$L_3 = (1,7,7)$
(8,15,34)(F)	(6,13,26)(F)	(2,9,16)(F)	(6,13,29)(F)	(1,4,7)(F)	(1,7,7)(F)	(5,11,12)	(0,0,0)	
(8,15,34)(F)	(6,13,26)(F)	(2,9,16)(F)	(6,13,29)(F)	(1,4,7)(F)	(1,7,7)(F)	(5,11,12)(F)	(0,0,0)	$L_2 = (5,11,12)$
(8,15,34)(F)	(6,13,26)(F)	(2,9,16)(F)	(6,13,29)(F)	(1,4,7)(F)	(1,7,7)(F)	(5,11,12)(F)	(0,0,0)(F)	L_1=(0,0,0)

From the above two tables,
$E_1 = L_1 = (0,0,0), E_4 = L_4 = (1,4,7), E_6 = L_6 = (2,9,16),\ E_7 = L_7 = (6,13,26) E_8 = L_8 = (8,15,34)$.
As a result, the critical path is 1→4→6→7→8, and the project duration is (8, 15, 34).

Now, the TFN (8, 15, 34) is converted into normal time using Triangular fuzzy ranking formula $\frac{a+b+c}{3}$. The defuzzified value of (8, 15, 34) is 19.

Therefore, the project ends in 19 days.

4 Traditional methods

4.1 Program Evaluation Review Technique (PERT)

Program Evaluation Review Technique (PERT) is a project management method for estimating how long it will take to complete a project successfully. There is an approach breakdown structure in project management that split a project into minor projects or activities. Every activity has its timeframe; it demands requirements and gives a result. Much of the time, these activity times are non-deterministic. In specific circumstances, the traditional PERT obtains three-point estimates; optimistic, pessimistic, and most likely. It is a simple strategy that uses a beta distribution mechanism.

The estimation duration for every activity can be predicted by the beta distribution means of the following weighted average:

$$ET = \frac{(Optimistic + 4 * Most\ likely + Pessimistic)}{6}$$

Here we are calculating activity durations using the mean of probabilistic times and presented them in Table 5. The related network diagram is presented in Figure 4.

Table 5: *Duration of activities with probabilistic mean*

Activity	a	m	b	$ET=\frac{a+4m+b}{6}$
1→2	5	6	7	6
1→3	1	3	5	3
1→4	1	4	7	4
2→5	1	2	3	2
3→6	1	2	9	3
4→6	1	5	9	5
4→7	2	2	8	3
6→7	4	4	10	5
5→8	2	5	8	5
7→8	2	2	8	3

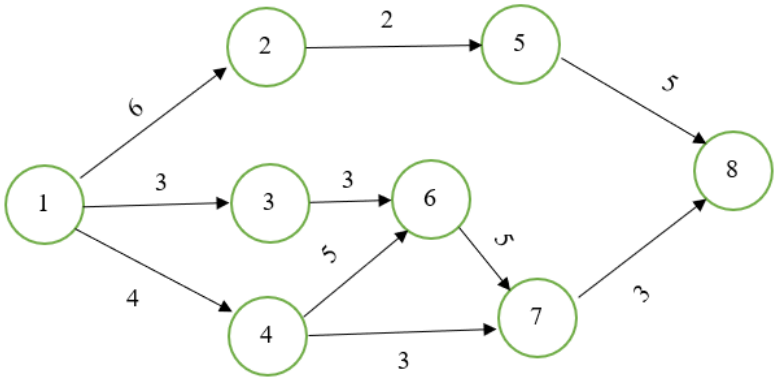


Figure 4: *Activity duration with probabilistic time's network diagram*

4.2 Procedure to find the Critical path:

Step 1: Establish a project network *G (V, E).*

Step 2: Express every activity time as probabilistic time.

Step 3: Determine the earliest start time of activity using forward pass calculations. Let the earliest time as zero for the initial event, $\tilde{E}_1 = 0$.

Then $\tilde{E}_j = max\{\tilde{E}_i + \tilde{t}_{ij}\}$ where *i*= number of preceding events

Step 4: Compute the earliest finish time of activity;

$$(E\tilde{F}_{ij}) = Earliest\ start\ time + the\ activity\ duration$$

$$i.e.\ E\tilde{F}_{ij} = E\tilde{S}_{ij} + \tilde{t}_{ij} = \tilde{E}_i + \tilde{t}_{ij}$$

Step 5: Estimate the latest finish time of activity using backward pass calculations.

$$\tilde{E}_n = \tilde{L}_n.\ \text{So that } \tilde{L}_i = L\tilde{F}_{ij} = min\{\tilde{L}_j - \tilde{t}_{ij}\}, i = n-1, n-2, \dots\dots, 2, 1.$$

Step 6: Calculate the latest start time of activity $(L\tilde{S}_{ij}) = L\tilde{F}_{ij} - \tilde{t}_{ij}$

Step 7: Total float $(T\tilde{F}_{ij}) = L\tilde{F}_{ij} - E\tilde{F}_{ij}\ \ or\ \ L\tilde{S}_{ij} - E\tilde{S}_{ij}$

Table 6: *Earliest and Latest tomes of project activities with probabilistic mean*

Activity	Node	Activity duration	$E\tilde{S}_{ij}$	$E\tilde{F}_{ij}$	$L\tilde{S}_{ij}$	$L\tilde{F}_{ij}$	$T\tilde{F}_{ij}$
1→2	P	6	0	6	4	10	4
1→3	Q	3	0	3	3	6	3
1→4	R	4	0	4	0	4	0*
2→5	S	2	6	8	10	12	4
3→6	T	3	3	6	6	9	3
4→6	U	5	4	9	4	9	0*
4→7	V	3	4	7	11	14	7
6→7	W	5	8	13	12	17	4
5→8	X	5	9	14	9	14	0*
7→8	Y	3	14	17	14	17	0*

The critical activities are 1→4, 4→6, 5→8, 7→8.
Therefore, the critical path is 1→4→6→7→8, and the project completion time is 17.

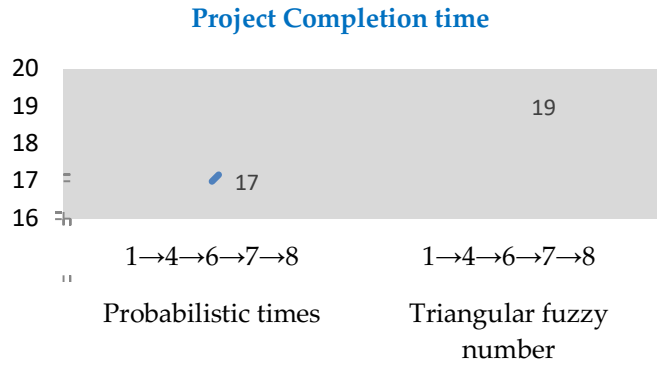
5. Results

Table 7 presents, Critical path and Project duration with Probabilistic and Triangular fuzzy activity times, respectively.

Table 7: *Results*

Activity times	Critical Path	Project Completion time
Probabilistic times	1→4→6→7→8	17
Triangular fuzzy number	1→4→6→7→8	19

Graph 1 presents results that correlate between Probabilistic and Triangular fuzzy mean.



Graph 1: *Correlates the project completion time with Probabilistic mean and Triangular fuzzy mean*

6. Discussion

This article determines the project's earliest and latest times by Modified Dijkstra's algorithm with a triangular fuzzy number and probabilistic times. The network's critical path is identified using project activities earliest and latest times. Moreover, the entire project time is calculated. The project critical path is the same in both cases, but the project completion time is different. Probabilistic mean gives less time compare to fuzzy triangular mean. However, in a non-academic example, this number is affected by various circumstances such as the availability of analysts, the type of activity.

References

[1] Chang and J. A. Abraham, "CHAN: An efficient critical path analysis algorithm," 1993 European Conference on Design Automation with the European Event in ASIC Design, 1993, pp. 444-448.

[2] Cheng X.L., Y. L. Zhang and X. S. Cui (2004), Applied in Resource Constrained Project Scheduling Problem, *Industrial Engineering Journal*, 7(3): 51-54.

[3] Lee, D. (2005), Probability of project completion using stochastic project scheduling simulation. *Journal of Construction Engineering and Management*, 131(3), 310-318.

[4] Li. Q, J. Zhang and R. T. Zhang, Application of Monte Carlo Simulation in Project Schedule, *Journal of Yangtze University (Natural Science Edition)*, 2007, 4(2): 62-66.

[5] Lu, M., & AbouRizk, S. M. (2000), Simplified CPM/PERT simulation model. *Journal of Construction Engineering and Management*, 126(3), 219-226.

[6] MacCrimmon k. and Ryavec, An Analytical Study of the PERT Assumptions. *Operation Research*, 12(1964), 16-37.

[7] Malcon. D.G., J.H. Roseboom, C.E. Clark and W. Fazar (1959), Application of a Technique for Research and Development Program Evaluation.

[8] Salmon Sackey and Byung-Soo Kim (2018), Schedule Risk Analysis using a proposed Modified Variance and Mean of the Original Program Evaluation and Review Technique Model. *KSCE Journal of Civil Engineering*, 23(4), 1484-1492.

[9] Shankar, N. Ravi, and V. Sireesha. "Using modified Dijkstra's algorithm for critical path method in a project network." *International Journal of Computational and Applied Mathematics* 5.2 (2010): 217-225.

[10] Xiaokang Han, Wenzhou Yan, Mei Lu, "Intelligent Critical Path Computation Algorithm Utilising Ant Colony Optimisation for Complex Project Scheduling", *Complexity*, vol. 2021, Article ID 9930113, 8 pages, 2021.

[11] Zadeh L.A. (1965), Fuzzy sets, *Information and Control*, Volume 8, Issue 3, Pages 338-353, ISSN 0019-9958.