DOI: 10.24996/ijs.2022.63.5.34

ISSN: 0067-2904

Developing a Heuristic Algorithm to Solve Uncertainty Problem of Resource Allocation in a Software Project Scheduling

Marwa Adeeb Al-Jawaherry[1]*, Ashraf Abdulmunim Abdulmajeed[2], Tawfeeq Mokdad Tawfeeq 3

[1]*Department of Computer Science, College of Computer Sciences and Mathematics, Tikrit University, Tikrit, Iraq*

[2]*Department of Software, College of Computer Sciences and Mathematics, University of Mosul, Mosul, Iraq*

[3]*Department of Software, College of Computer Sciences and Mathematics, University of Mosul, Mosul, Iraq*

Received: 9/4/2021 Accepted: 5/7/2021 Published: 30/5/2022

Abstract

In project management process, the objective is to define and develop a model for planning, scheduling, controlling, and monitoring different activities of a particular project. Time scheduling plays an important role in successful implementation of various activities and general outcome of project. In practice, various factors cause projects to suffer from time delay in accomplishing the activities. One important reason is imprecise knowledge about time duration of activities. This study addresses the problem of project scheduling in uncertain resource environments, which are defined by uncertain activity durations. The study presents a solution of the levelling and allocation problems for projects that have some uncertain activities. The resources are minimised using resource levelling based on a proposed heuristic algorithm with limited duration. The algorithm performs the resource scheduling (levelling and allocation) for minimum moments. The efficiency of the proposed algorithm is indicated by the resource improvement coefficient and rate of resource utilisation.

Keywords: Scheduling, Resource Allocation, levelling, Floyd Warshall algorithm, Heuristic Algorithm.

تطوير خوارزمية إرشادية لحل مشكلة تخصيص الموارد غير المؤكدة في جدولة مشروع البرمجيات

مروى اديب الجواهري[1]*, اشرف عبدالمنعم عبد المجيد[2], توفيق مقداد توفيق[3]

[1]قسم علوم الحاسوب, كلية علوم الحاسوب والرياضيات, جامعة تكريت, تكريت, العراق

[2]قسم البرمجيات, كلية علوم الحاسوب والرياضيات, جامعة الموصل, الموصل, العراق

[3] قسم البرمجيات, كلية علوم الحاسوب والرياضيات, جامعة الموصل, الموصل, العراق

الخلاصة

الهدف من عملية إدارة المشاريع هو تعريف وتطوير نموذج للتخطيط, الجدولة, السيطرة ومراقبة الأنشطة (Activities) المتنوعة في مشروع معين. ان لعملية جدولة الوقت في المشروع دورًا مهمًا في التنفيذ

*Email: marw.a.aljawaherry@tu.edu.iq

الناجح للأنشطة المختلفة والنتائج العامة للمشروع. ومن الناحية العملية ، تتسبب عوامل مختلفة في معاناة المشاريع من التأخير الزمني في إنجاز تلك الأنشطة، أحد الأسباب المهمة هو عدم الدقة في معرفة المدة الزمنية لإنجاز كل نشاط. تتناول الدراسة الحالية مشكلة جدولة المشروع التي يتم تطويرها في بيئات ذات موارد غير المؤكدة ، والتي يتم تحديدها من خلال فترات غير مؤكدة لإنجاز الانشطة. هذه البحث يقدم حل لمشاكل تسوية وتخصيص المصادر (levelling and allocation problems) للمشاريع ذات الأنشطة غير المؤكدة. اذ يتم تقليل الموارد إلى الحد الأدنى باستخدام تسوية الموارد بناءً على خوارزمية إرشادية مقترحة بمدة محدودة. تقوم الخوارزمية بجدولة الموارد (التسوية والتخصيص) بأقل وقت. كما تمت الإشارة إلى كفاءة الخوارزمية المقترحة من خلال معامل تحسين الموارد ومعدل استخدام الموارد.

1. Introduction

In developing countries, project implementation plays a fundamental and effective role in national development and economic improvement. In these countries, projects are regarded as the wheels of vehicle of growth and development in economic dimensions. Optimal and accurate management of these projects lead to successful implementation, increased revenues, and decreased costs. One of important tasks in project management is scheduling, which has become difficult due to resource constraints along with precedence relationships [1].

The resource-constrained project scheduling problem (RCPSP) is concerned with project scheduling when resources are limited. In this case, which is suitable for projects with a larger number of activities, using heuristics and approximation algorithms is necessary; for small projects, obtaining exact solutions using the branch-and-bound method is necessary [2,3].

However, during project execution, operations are subject to a great deal of ambiguity, which can trigger multiple schedule disturbances. This complexity may stem from a variety of factors. For example, activities may take longer or shorter than expected, resources may become scarce, supplies may arrive late, ready times and due dates may need to be adjusted, new activities may need to be added, or activities may need to be dropped due to project scope changes [2].

Uncertainty is a fact of life in project planning and control. Although this fact is accepted by managers, much of the literature pertaining to project resource allocation/scheduling issues ignores this issue, thereby resulting in disastrous consequences as evidenced by the number of actual projects that are completed late and exceed their budget. Keeping this in mind, engineering project managers seek to develop and create a model that will find an optimal way to assign resources to the various stochastic activities of a project to minimise the stated economic objective function.

In uncertain resource environments, the problem is how to perform a scheduling process, create a mathematical model, and produce a simple algorithm to properly solve the task. A project scheduling problem with uncertain resource constraints consists of two elements. First is the need to maximise the limited capacity for a fixed time duration throughout the project. Second is to maximise the limited resource capacity, which is considered a random variable [4].To approach project scheduling with limitless resources, some project management techniques such as the critical path method (CPM) and program assessment and analysis technique (PERT) were developed in the 1950s and 1960s [5].

The CPM uses activities with known fixed durations and infinite resources to estimate the minimum length of a project [6].

The critical path (CP) is the longest path between the start node (beginning of the project) and the finishing node (end of the project) in the activity network of the project. Various approaches are used to compute the CP (the route between nodes through project activities, and a value that sums the time durations required by each activity); these approaches include dynamic programming, Dijkstra method, and Floyd's algorithm [7].

PERT was founded in 1958 to organize activities related to the construction of the US Navy's Polaris missile system, a project fraught with uncertainty [5].

The PERT approach assumes probabilistic activity durations. Therefore, When determining the durations of project operations, probabilistic assumptions are taken into account. For each activity, three estimates are used: optimistic time t_{opt}, pessimistic time t_{pes}, and most likely time t_m. Usually, the probabilistic data for project activities obey the beta distribution for which the mean μ and variance $\sigma^2$are given by:

$$\mu = \frac{t_{opt}+4t_m+t_{pes}}{6} \quad (1)$$

$$\sigma^2 = (\frac{t_{pes}-t_{opt}}{6})^2 \quad (2)$$

The expected project completion time is$\sum \mu_i$, where μ_i is the expected duration of the critical activity, and the variance of the project completion time is$\sum_i \sigma_i^2$, where σ_i^2 is the variance of the ith critical activity in the CP [8].

This study focuses on determining the best resource allocation to all activities of the network to minimise cost. A proposed heuristic algorithm is developed for resource scheduling based on minimising a certain type of moment. MATLAB software was used to simulate the developed algorithm to solve the uncertainty problem and resource allocation.

The rest of this paper is organised as follows. Section 2 provides a literature review. The resource allocation problem (RAP) is elaborated in Section 3. Then, resource levelling is described in section 4. Floyd–Warshall algorithm is defined in section 5. The proposed algorithm is explained in section 6. The results are provided in section 7. Finally, the conclusion and future work is outlined in section 8, followed by the acknowledgment and a list of references.

2. Literature Review

In this portion, previous work on the RAP is critically evaluated. Between the 1930s and 1950s, project management became a recognized field for use in engineering projects, where standardised methods and tools were used, giving rise to project management models [9]

Many scheduling models have appeared, beginning with the CPM, which can be used to estimate the minimum length of a project using tasks with known set durations and unlimited resources.

PERT was founded in 1958 in the US Navy to organize activities in the production of the Polaris missile system, which was fraught with uncertainty. Many researchers have contributed to the field of project engineering management using unclear RAPs in operation networks [10].

These researchers used various models and solution techniques. For example, they developed an optimization model for resource smoothing depending on genetic algorithm (GA) procedure built on the C++ program to find the optimum solution. The project management program Microsoft Projects is used to conduct resource leveling in order to achieve the best possible result. [11].

Other researchers propose a new methodology that considers fuzzy modeling of the workload inspired by the fuzzy/possibility method to retain ambiguity at all stages of the modeling and solving process. Two project scheduling strategies, resource-constrained scheduling, and resource levelling are considered and generalized to manage fuzzy parameters based on this modelling. The fuzzy-resource-constrained project scheduling problem (FRCPSP) and the fuzzy-resource levelling problem are the names of these problems (FRLP). To solve FRCPSP and FRLP, a greedy algorithm and a GA are given, and they are applied to civil helicopter maintenance as part of a French industrial project called Heli Maintenance [12].

A fuzzy-resource-constrained project-scheduling problem, which is one of the operational research problems that can be described as a combinatorial NP-hard problem of constructing a special plan for performing some precedence-related tasks subject to limited uncertain resources, is presented in another contribution to the same area.

When operation durations are based on human estimates and there is no way to forecast them accurately due to a lack of historical data, fuzzy numbers are useful. The aim is to develop a fuzzy-heuristic-based method for obtaining a near-optimal solution in a reasonable amount of time. [13].

Another recent research on resource allocation used the RCPSP–FRM model to create a scheduling method that lets the project manager imagine the flexibility of resources in a plan. Flexibility is a project manager's partner and a method for properly managing it promotes project performance in terms of time and resource use. The tool shows the resource's versatility in both table and graph formats. It has a straightforward user interface and straightforward performance [5].

Another method for project planning is aim to create an algorithm for project scheduling with fuzzy time and resources. This algorithm calculates the latest start times of activities in a fuzzy environment, and then uses the parallel scheduling method to create a feasible schedule. We solved a project scheduling problem with a certain amount of money to demonstrate the efficacy of the proposed algorithm [14].

For the current study, the proposed algorithm of the RAP is new and has not been implemented previously. Moreover, it can compete with the existing techniques and provide the closest accuracy and real-time implementation.

3. Resource allocation problem in project Management

RAPs in project management are complex. Therefore, the development of efficient algorithms for solving various specific cases is a real problem [15].

RAPs are important in a number of fields, including manufacturing production control (e.g., production scheduling), warehousing (e.g., storage allocation), fleet management (e.g., freight transportation), staff management (e.g., office work), computer software scheduling (e.g., in massively parallel GRID systems), project management, and control. Management science relies heavily on RAPs [16].

Minimizing the period is important in resource allocation strategies for construction projects in project management. The allocation policy used to assess the fractions of resources to be allocated to constituent tasks has a significant impact on project durations. Due to closed-loop workflows that generate complex demand patterns and delays in shifting resources across activities, policies for shorter project durations are difficult to develop and enforce. Resource demand forecasts and resource adjustment periods are two policy features that administrators can easily change to affect project length. The researchers then developed a dynamic systems model to examine how a basic, common framework of project processes and resource allocation policies affects project durations [17].

4. Resource Levelling in Project Management

Delaying activities to solve resource issues is referred to as resource-levelling. It's a form of network analysis in which scheduling decisions are focused on resource management issues (start and finish dates). The primary aim of resource levelling is to create a more equal distribution of resource use. Project managers examine the network diagrams to identify areas of “slack” or “float” as well as resource conflicts. For example, occasionally, over allocations can be removed by delaying non-critical tasks, which does not result in an overall schedule delay. Other times, delaying the project completion date is necessary to reduce or remove over allocations [11]. On the other hand, resource levelling (also known as "resource smoothing") tends to reduce the sharp differences in the resource demand histogram while preserving the original project length. [18].

5. Floyd–Warshall Algorithm

The Floyd–Warshall algorithm is used to find the shortest (longest) paths between all pairs of nodes in a graph with no negative-length loops. The Floyd–Warshall algorithm's key benefit is its programmability and simplicity [19, 20].

Given a network N(V, A) with node set V = {1, 2, . . . , n} and arc set A = {(i, k) : i, k ∈V, i≠ k},where |V| = n, and with at least one cycle on the network, the Floyd–Warshall algorithm is probably the best-known and one of the most effective algorithms for finding the shortest path between every two nodes, i and k, in network N. This algorithm is based on a four-step procedure in which two square matrices, Dj and Rj, for j = 0, . . ., n is calculated, holding the shortest path costs and shortest routes (sources and sinks) between every two arbitrary nodes i and k, respectively. Although the algorithm seems simple, it requires extensive calculations. Given a network with n nodes, the Floyd–Warshall algorithm requires the Dj and Rj matrices to be calculated n + 1 times, starting from D0 and R0, where each has n2 − n entities. Algorithm 1 explains the Floyd–Warshall algorithm. Details can be found in [21, 22], see Figure 1.

6. Proposed Algorithm for Resource Leveling and Resource Allocation

A proposed algorithm is a heuristic procedure that is used to schedule constrained uncertain resources at an uncertain activity network, which is defined as a network that has some (or all) activities with time duration vectors of different lengths. The uncertain activity time duration in the proposed algorithm is given by most likely values and solved as a CP problem. An uncertain vector is assigned for each activity, where vector length represents actual variations around a most likely value. A matrix is assigned for each uncertain network T_m, $m = 1, 2\ldots N$, where N is the total number of networks.

Algorithm 1. The Floyd–Warshall algorithm

Step 1. Set D_j and R_j as two square $n \times n$ matrices, where j is the stage number and n is the total number of nodes of the network.

Step 2. For $j = 0$ calculate D_0 and R_0:

$D_0 = [d_{ik}]$, where

$$d_{ik} = \begin{cases} d_{ik} & \text{if there is a direct route connecting node } i \text{ to the node } k \\ \infty & \text{if there is no direct route connecting the node } i \text{ to the node } k \\ 0 & \text{if } i = k. \end{cases}$$

$R_0 = [r_{ik}]$, where

$$r_{ik} = \begin{cases} k & \text{if there is a direct route connecting node } i \text{to the node } k \\ - & \text{if there is no direct route connecting the node } i \text{to the node } k \\ - & \text{if } i = k. \end{cases}$$

Step 3. For the remaining $j = 1, \ldots, n$ calculate the D_j and the R_j matrices as follows. Note that from now on we derive the entities of the D_j and the R_j matrices on the basis of the entities of the most recent previous matrices, i.e. the D_{j-1} and the R_{j-1} matrices:

$D_j = [d_{ik}]$ where

$$d_{ik} = \begin{cases} d_{ik} & \text{if } i = k,\ i = j, k = j \\ \min(d_{ik},\ d_{ij} + d_{jk}) & \text{otherwise.} \end{cases}$$

$R_j = [r_{ik}]$ where

$$r_{ik} = \begin{cases} k & \text{if } i = k, i = j, k = j \\ k & \text{if } d_{ik} \le d_{ij} + d_{jk} \\ j & \text{if } d_{ik} > d_{ij} + d_{jk}. \end{cases}$$

Step 4. Repeat step 3 until the D_n and the R_n are yielded.

Figure 1-The Floyd–Warshall algorithm [23].

The CP value and critical route through the nodes are computed using Floyd's algorithm, and each network has a vector V_{Mv} that stores its CP values. The not preferred case of project computation time (maximum value) is calculated as:

$$Max\ critical\ value = \max\{V_{Mv}\}$$

The preferred case of project computation time (minimum value) is calculated as:

$$Min\ critical\ value = \min\{V_{Mv}\}$$

Resource levelling is a project management method that ignores resource distribution and resolves potential disputes caused by over allocation. Resource levelling aims to efficiently assign a resource so that the project can be completed within the specified time frame. There are two types of resource levelling projects: those that can be completed by exhausting all available resources and those that can be completed with minimal resources. When resource-levelling fails to produce the desired effects of reduced resources, resource allocation is used.

The proposed program aims to utilise the data calculated by Floyd's algorithm to distribute the resource for each activity. Moreover, the program illustrates the earliest and latest starts. The analysis of the data and results indicate that the resource rate is extremely high and thus, the resource levelling and allocation method is applied to decrease this rate. In this study, the distinguished feature of the proposed resource levelling is a heuristic method that combines the minimum moment and pack methods. The minimum moment algorithm was used to determine a calculation of daily resource demand fluctuations. The following steps describe how the proposed algorithm works:

1. The minimum moment for each uncertain activity is computed as:

$$Mx = \sum_{=1}^{d} \left[(R_i) \times \frac{1}{2} \ R_i \right] \quad (3)$$

Where d represents the duration of the project, and R_i represents the resource demands on day All activity not on the CP should be moved.

2. Non-critical activity with the highest value is shifted, and then a newMx is computed.
3. If Mx decreases, then the next highest value is selected and step 3 is repeated.

Otherwise, activity is shifted by 50% from its value and $Mx$is checked again.

4. If Mxdoes not decrease, the chosen activity is shifted by 25 from its value and Mx is checked again. Then the minimum value of $Mx$is selected, which is fixed for consecutive iterations.
5. The difference between the highest and lowest values of $Mx$is determined, which represents the maximum decrease that can be achieved by resource levelling.
6. The following are computed:

(i) maximum demand of resource,

(ii) utilisation or resource rate (RR) using the following equation:

$$RR = \sum_{i=1}^{d} \frac{(Ri)}{d * Max(Ri)} \quad (4)$$

(iii) Resource improvement coefficient (RIC), which must be decreased to reach a minimum value ≤1 using the following equation:

$$RIC = \frac{d \ \sum(Ri^2)}{\sum(Ri)^2} \quad (5)$$

7. The computed value of $Mx$is assigned in step 5 as project duration.

7. Results and Analysis

When a project completion period cannot be postponed despite the need for additional funding, it is known as time-limited. However, the extra resource consumption should be limited to the absolute minimum. This is referred to as "resource levelling" or "smoothing." It is used to minimize the hiring and firing of resources as well as the fluctuation in a resource's daily demand. The aim, in this case, is to move non-critical activities of the original schedule within their float times so that a better resource can be achieved; resources are not constrained in this analysis, and project length is not required to be delayed; the goal, in this case, is to shift non-critical activities of the original schedule within their float times so that a better resource can be achieved. If the extent of resource availability cannot be met, the project is resource-constrained.

Project delays are acceptable in conditions where resources are scarce, but they should be kept to a minimum. Following the assigning of resources to each task, the next step is to aggregate the resources used by all activities. Resource aggregation is essentially the summation of the resources needed to complete all operations based on the previous resource allocation on a period-by-period basis. The data is commonly represented graphically as a histogram. Depending on the time unit used to assign resources, such aggregation may be performed hourly, regular, or weekly.

The resource aggregation is fairly easy and straightforward when a bar chart is used. In the next section, a numerical example is presented to explain the use of the proposed heuristic algorithm for scheduling activities. Thereafter, the uncertain case is explained for a specific project as a case study. For this purpose, software programs written in MATLAB code are developed. These programs solve the levelling and allocation management, moment calculation, uncertainty analysis, and uncertain allocation management.

7.1 Numerical Example

In this section, a hypothetical example is considered. The project consists of 20 activities from T_1 to T_{20}. The detailed information on these activities, such as their durations, predecessors, and resources, are summarised in Table 1. The data shown in Table 1 have been converted to an activity-on-arrow (AOA) network diagram in Figure 2. This conversion is essential for applying Floyd's algorithm that transacts with the network to solve a major issue in this case.

According to Figure 2, two matrices are deduced. First is the time duration matrix of the network $Time_{20X20}$, where the entries of *Time* indicate the completion time of each activity, for example, $Time_{1,2} = 6$. Second is the identifier matrix *T*, where the entries represent activities as T_1 to T_{20}, and six dummy activities (D_1 to D_6), Where dummy activity is an activity added to a project schedule as a placeholder. It has no activity time associated with it. A dummy activity is intended to show a path of action in a project activity diagram and is employed when a logical relationship between two activities cannot be linked by showing the use of arrows linking one activity to another.

Floyd's algorithm deals with all required calculations aside from computing the CP as maximum time completion of the project. Moreover, the algorithm determines the activities earliest start times and latest completion times, and the duration values of the critical activities along the CP. Therefore, the computation of the total float time of each activity is performed directly. Initially, applying Floyd's algorithm to the project network presents a CP = 32 weeks, and the critical activities are $T_{1,}$ $T_{2,}$ $T_{5,}$ $T_{8,}$ $T_{13,}$ $T_{17,}$ $T_{19,}$ andT_{20}. The time completions of the critical activities are 6, 4, 5, 6, 5, and 6. Table 1 shows the value of the earliest and latest starts for each activity.

Table 1- Time schedule and precedence relation of 20 activities

No.	Activity	Previous Activity	Duration	Resource
1	T_1	-	6	7
2	T_2	-	3	9
3	T_3	T_1	4	3
4	T_4	-	6	5
5	T_5	T_2	7	8
6	T_6	T_3	5	2
7	T_7	T_4	2	9
8	T_8	T_2	2	9
9	T_9	T_8, T_7	2	4
10	T_{10}	T_6	6	7
11	T_{11}	T_3, T_5	1	5
12	T_{12}	T_5, T_8	2	3
13	T_{13}	T_9	4	4
14	T_{14}	T_6	2	4
15	T_{15}	T_{12}	3	2
16	T_{16}	T_{10}, T_{14}, T_{13}	5	7
17	T_{17}	T_{15}	8	3
18	T_{18}	T_{15}, T_4	2	7
19	T_{19}	T_{16}	6	3
20	T_{20}	T_{17}	2	5

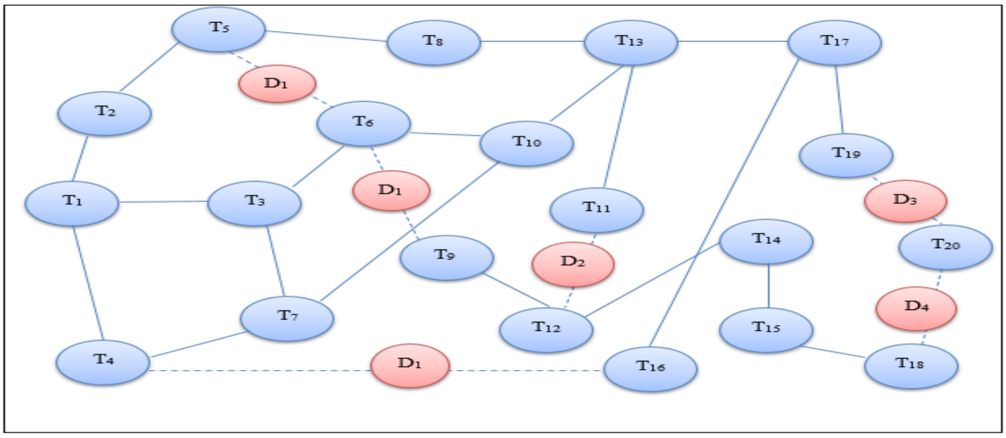

Figure 2-Activity-on-arrow (aoa) network diagram

Table 2- The value of activity earliest start and latest start

Node (Activity)	Earliest time EST	Latest time LCT
T_1	0	0
T_2	6	6
T_3	6	10
T_4	6	13
T_5	10	10
T_6	13	17
T_7	8	15
T_8	15	15
T_9	13	17
T_{10}	10	17
T_{11}	15	19
T_{12}	15	19
T_{13}	21	21
T_{14}	18	22
T_{15}	26	30
T_{16}	18	24
T_{17}	26	26
T_{18}	28	32
T_{19}	32	32
T_{20}	32	32

Consequently, the developed software provides Table 3, which shows the relationship among all the 30 activities (including the dummy activities) with their time duration and total free time. We note that the critical activities have zero total free float FT, and the dummy activities have zero duration time.

Table 3- Activities relationships total free time

Start activity	End activity	Time Duration	Total free time
T_1	T_2	6	0
T_1	T_3	3	7
T_1	T_4	6	7
T_2	T_5	4	0
T_3	T_6	7	4
T_3	T_7	2	7
T_4	T_7	2	7
T_4	T_{16}	0	18
T_5	T_6	0	7
T_5	T_8	5	0
T_6	T_9	0	4
T_6	T_{10}	1	3
T_7	T_{10}	2	7
T_8	T_{13}	6	0
T_9	T_{12}	2	4
T_{10}	T_{13}	4	7
T_{11}	T_{12}	0	4
T_{11}	T_{13}	2	4
T_{12}	T_{14}	3	4
T_{13}	17	5	0
T_{14}	T_{15}	8	4
T_{15}	T_{20}	2	4
T_{16}	T_{17}	2	6
T_{17}	T_{19}	2	6
T_{18}	T_{20}	0	4
T_{19}	T_{20}	0	0

To prepare for resource scheduling, Table 4 is formed by ascending-order sorting according to the earliest estimated start time of activities, and resources are correspondingly associated with each activity, including the dummy one. For dummy activities, zero resources are assigned. The corresponding resource histogram is shown in Figure 3.2, where the resources vary between a maximum value M_v = 29 and a minimum value mv = 4. The values of the moment M_x, rate of resource RR, and resource improvement coefficient RIC are 6457, 43.64%, and 1.2597, respectively.

Table 4- Activity resources sort according earliest time

Activity	Earliest StartTime	Total Float	Activity Duration	Resource
T_1	0	0	6	7
T_2	0	7	3	9
T_4	0	7	6	5
T_3	6	0	4	3
T_5	6	4	7	8
T_8	6	7	2	9
T_7	6	7	2	9
D_1	6	18	0	0
T_9	8	7	2	4
D_2	10	7	0	0
T_6	10	0	5	2
T_{13}	10	7	4	4
D_3	13	4	0	0
T_{11}	13	3	1	5
T_{12}	13	4	2	3
T_{10}	15	0	6	7
D_4	15	4	0	0
T_{14}	15	4	2	4
T_{15}	15	4	3	2
T_{17}	18	4	8	3
T_{18}	18	6	2	7
T_{16}	21	0	5	7
T_{20}	26	4	2	5
T_{19}	26	0	6	3
D_5	28	4	0	0
D_6	32	0	0	0

Another histogram is shown in Figure 3. It depicts the durations of all project activities (rectangular shapes) with their total float (line shapes) over the project time up to the CP value before scheduling. The activities are ordered from T_1 to T_{20}. The critical activities T_1, T_3, T_6, T_9, T_{16}, and T_{19} have no total floats.

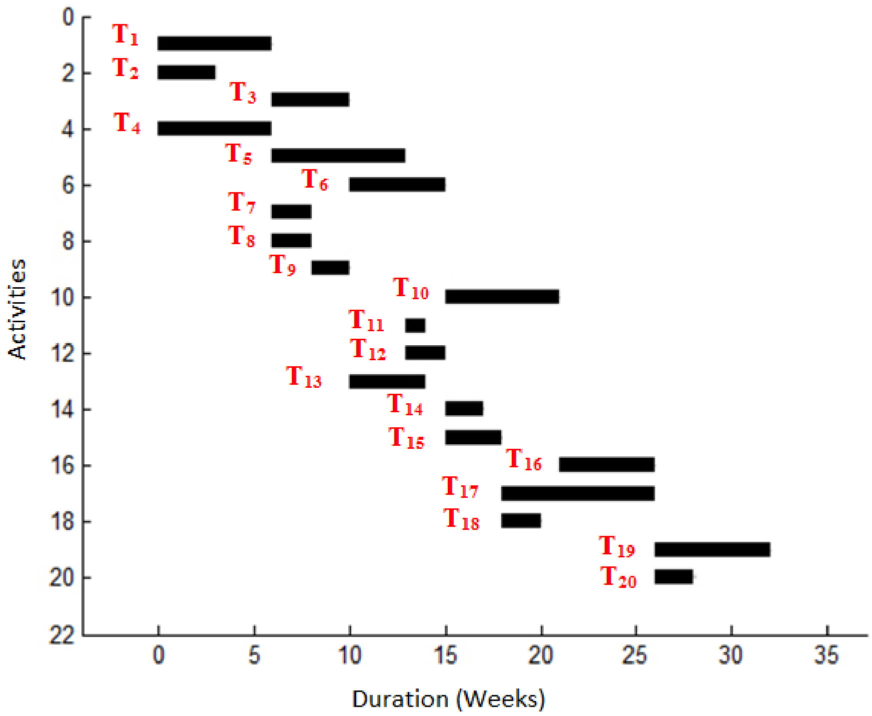

Figure-3 Bar chart before leveling

7.2 Resource Project Levelling

In this section, the proposed heuristic algorithm of the project scheduling is used to minimise M_x. As a result, the activity level should provide a minimisation of the maximum utilizing resources, consequently increasing *RR* and decreasing *RIC*. The developed software makes four iterations to complete the levelling management. Table 5 (a) and (b) show the results of the first and last iterations. The proposed heuristic algorithm provides an improvement of 63.5%. Although this improvement is not optimal, it is significant for a heuristic algorithm. Obviously, the start times of all critical activities are not changed.

Table 5 a- Leveling management (First iteration)

Moment (M_X)	Maximum Resource Demands (M_v)	Resource Rate of Use (*RR*)	Resource improve coefficient (*RIC*)
6458	29	43.644	1.2593
6282	21	57.538	1.2244
6264	21	60.257	1.2212
6206	21	60.269	1.2109
6022	21	60.269	1.1749
6008	21	60.276	1.1723
5948	21	60.258	1.1605

Table 5 b- Leveling management (Last iteration)

Moment (M_X)	Maximum Resource Demands (M_v)	Resource Rate of Use (RR)	Resource improve coefficient (RIC)
6459	29	43.644	1.2596
6283	21	57.538	1.2244
6261	21	60.269	1.2216
6207	21	60.272	1.2106
6022	21	60.272	1.1737
6017	21	60.271	1.1718
5949	21	60.271	1.1601
5921	21	60.271	1.1529
5733	21	60.271	1.1179
5692	21	60.271	1.1104
5621	21	60.271	1.0950

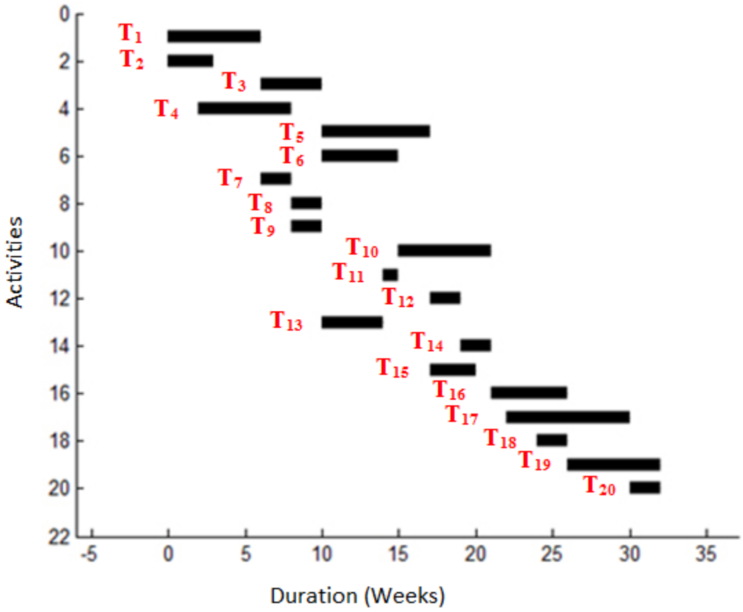

Figure 4-Bar chart after levelling.

Table 6- Summary of project scheduling (levelling)

Activity	Start Time	Finish Time	Resource	Total Float
T_1	0	6	7	0
T_2	0	3	9	7
T_3	6	10	3	0
T_4	2	8	5	0
T_5	10	17	8	0
T_6	10	15	2	0
T_7	6	8	9	0
T_8	8	10	9	7
T_9	8	10	4	0
T_{10}	15	21	7	7
T_{11}	14	15	5	0
T_{12}	17	19	3	0
T_{13}	10	14	4	0
T_{14}	19	21	4	7
T_{15}	17	20	2	0
T_{16}	21	26	7	0
T_{17}	22	30	3	0
T_{18}	24	26	7	0
T_{19}	26	32	3	0
T_{20}	30	32	5	0

7.3 Resource Project Allocation

Using the developed software, we prepare the data to start computing the allocation management, as shown in Table 7. As mentioned, the data are sorted according to the start times for all project activities.

Table 7- Preparing data before allocation

Activity	Start Time	Finish Time	Resource	Total Float
T_1	0	6	7	2
T_2	0	3	9	9
T_3	6	10	3	2
T_4	0	6	5	9
T_5	6	13	8	6
T_6	10	15	2	2
T_7	6	8	9	9
T_8	6	8	9	9
T_9	8	10	4	9
T_{10}	15	21	7	2
T_{11}	13	14	5	5
T_{12}	13	15	3	6
T_{13}	10	14	4	9
T_{14}	15	17	4	6
T_{15}	15	18	2	6
T_{16}	21	26	7	2
T_{17}	18	26	3	6
T_{18}	18	20	7	8
T_{19}	26	32	3	2
T_{20}	26	28	5	6

Using the proposed heuristic algorithm, we conducted five iterations to achieve the results provided in Table 8. These results imply an overall improvement of the activity scheduling (levelling and allocation).

Table 8- Resource Allocation Method (Last Iteration)

Moment (M_X)	**Maximum Resource Demands (M_y)**	**Resource Rate of Use (*RR*)**	**Resource improve coefficient (*RIC*)**
6458	29	41.077	1.3386
6311	26	45.813	1.3084
6170	22	54.147	1.2790
6030	21	56.726	1.2499
5781	21	56.727	1.1978
5659	21	56.725	1.1729
5632	21	56.722	1.1675
5620	21	56.725	1.1649
5591	21	56.725	1.1588
5560	22	54.145	1.1526
5544	26	45.816	1.151
5482	21	56.723	1.1364
5453	21	56.724	1.1300
5420	19	62.692	1.1236
5366	19	62.694	1.1123
5343	18	66.178	1.1073

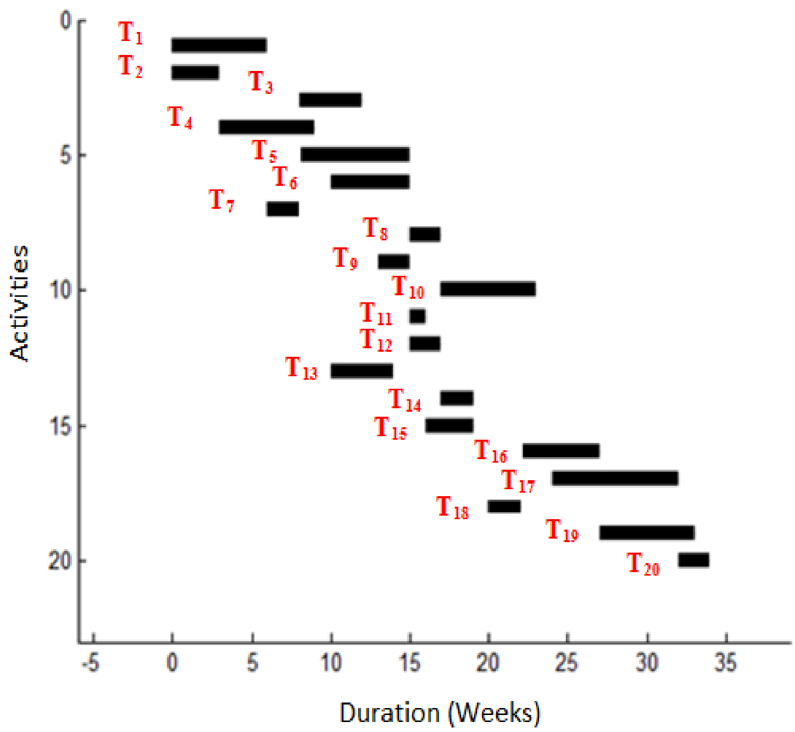

Figure 5-Bar chart after allocation

In the next step, as recommended by the proposed algorithm, the completion time of a project has to be increased by a step of unit of time (in the considered example, by a week). This means adding a dummy activity between the start and finish nodes with duration equal to the CP value plus the step. For this purpose, a step of one week is assumed, and the calculation is performed up to eight weeks. In other words, the developed software will be used to determine the characteristics of the allocation management for eight cases.

The results the allocation management are provided in Table 9. First, because the proposed algorithm is based on the minimisation of the moment M_x, the M_x continues decreasing for all cases. This result indicates the algorithm validity and signifies the possibility of having an optimum result. For instance, the two, four and eight-week give best results; however, these cases compete in Mv, RR, and RIC values. Although, the maximum resource can be 17 and an RIC value of 1.0806.

Table 9- Analysis of Allocation Management

Adding Weeks	CP	M_x	M_v	RR	RIC
0	32	5611	21	60.268	1.0947
1	33	5513	23	53.36	1.1092
2	34	5341	18	66.176	1.1071
3	35	5307	20	57.857	1.1324
4	36	4963	20	62.5	1.0893
5	37	5103	21	54.752	1.1511
6	38	5025	18	50.752	1.1642
7	39	4837	17	61.086	1.1501
8	40	4431	17	59.559	1.0806

8. Conclusions and Future works

This study presents the development of an algorithm to solve the uncertainty problem of resources in a project schedule to assist the project manager in ensuring success in terms of time and resource usage. The work applied Floyd's algorithm on AOA on the case study. The application provided variable data, which were used for resources, and the data obtained from Floyd's algorithm helped to draw all the essential histograms, which were useful in understanding the actual activities and assisted in resource allocation. The heuristic algorithm combines two approaches: minimum moment and Peck methods. This approach can reduce the amount of resources. This number can be further minimised using resource allocation. The results obtained from the resource allocation are introduced to the program of heuristic method to obtain a time matrix, which provides the modified time scheduling. Thus, the original time of the schedule is replaced. In addition, this model solves the uncertain RAP and presents the best solution but not the optimal one. This model is easy to use. The GA can be used to enhance the results. A further application can be introduced, which is a weight index for (M_X, M_y, M_X+M_y), to ensure the best time extension for the schedule.

9. Acknowledgment

The authors would like to thank Tikrit University / College of Computer Science and Mathematics and University of Mosul / College of Computer Science and Mathematics for their facilities, which have helped to enhance the quality of this work.

References

[1] F. Habibi, F. Barzinpour and S. J. Sadjadi, "Resource-Constrained Project Scheduling Problem: Review of Past and Recent Developments," *Journal of Project Management*, vol. 3, no. 2, pp. 55–88, 2018.

[2] W. Herroelen and R. Leus, "Resource-constrained project scheduling: A survey of recent developments," *Journal of Computers & Operations Research Res*, vol. 25, no. 4 , pp. 279–302, 1998.

[3] P. Brucker, S. Knust, A. Schoo and O. Thiele, "A Branch and Bound Algorithm for the Resource-Constrained Project s Scheduling Problem," *European Journal of Operational Research – Elsevier* , vol. 107, no. 2, pp. 272–288, 1998 .

[4] D. A. Dolgov and E. H. Durfee, "Resource Allocation Among Agents with MDP-Induced Preferences," *Journal of Artificial Intelligence Research*, vol. 27, no. 1, pp. 505–549, 2006. .4

[5] R. Lima, A. Tereso and J. Faria, "Project management under uncertainty: resource flexibility visualization in the schedule," *Procedia Computer Science*, vol. 164, no. 12, pp. 381–388, 2019.

[6] J. E. Kelley and M. R. Walker, "Critical-path planning and scheduling," in Proceedings of the eastern joint IRE-AIEE-ACM computer conference on - IRE-AIEE-ACM, 1959.

[7] A. M. Aliyu, "PROJECT MANAGEMENT USING CRITICAL PATH METHOD (CPM): A PRAGMATIC STUDY, "*Global Journal of Pure and Applied Sciences*, vol. 18, no. 3&4, pp. 197-206, 2012.

[8] A. Göksu and S. Ćatović , "Implementation Of Critical Path Method And Project Evaluation And Review Technique," in Proceedings of the 3rd International Symposium on Sustainable Development, , Sarajevo, May 31 - June 01 2012.

[9] G. Garel, "A history of project management models: From pre-models to the standard models," International Journal of Project Management , Elsevier, vol. 31, no. 5, pp. 663– 669, 2013.

[10] K. R. MacCrimmon and C. A. Ryavec, "An Analytical Study of the PERT Assumptions," *Journal of Operational Research*, vol. 12, no. 1, pp. 16–37, 1964 .

[11] S. R. Mohammed and A. A. Abdul Ameer, "Optimization of Resource Allocation and Leveling Using Genetic Algorithms," *Journal of Engineering*, vol. 17, no. 4, pp. 929-947, 2011.

[12] M. Masmoudi and A. Haït, "Project scheduling under uncertainty using fuzzy modeling and solving techniques," *Journal of Engineering Applications of Artificial Intelligence*, Elsevier, Vol. 26, no. 1, pp. 135-149, 2013.

[13] M. Knyazeva, A. Bozhenyuk and I. Rozenberg, "Resource-constrained project scheduling approach under fuzzy Conditions," *Journal of Procedia Computer Science*, Elsevier, vol. 77, no. 12 , pp. 56 – 64, 2015.

[14] M. Khalilzadeh, H. Shakeri, H. Gholami and L. Amini, "A Heuristic Algorithm for Project Scheduling with Fuzzy Parameters", *Procedia Computer Science*, Elsevier, vol. 121, no. 10, pp. 63–71, 2017. Available : https://doi.org/10.1016/j.procs.2017.11.010

[15] Burkova, B. Titarenko, A. Hasnaoui and R. Titarenko, "Resource allocation problem in project management, "in Proceedings of E3S Web Conferences 97, 01003, 2019. Available: https://doi.org/10.1051/e3sconf/20199701003 .

[16] M. A. Eirgash, "Resource Allocation and Leveling in Construction Management Projects with Resource Histogram," *American Journal of Engineering and Technology Management*, vol. 5, no. 6, pp. 91-95, 2020.

[17] Z. W. Lee, D. N. Ford , and N. Joglekar, "Effects of Resource Allocation Policies for Reducing Project Durations: A Systems Modelling Approach," *Wiley interScience, Systems Research and Behavioral Science*, vol. 24, no. 6, pp. 551-566, 2007. Available: https://doi.org/10.1002/sres.809

[18] Moselhi and P. Lorterapong, "Least Impact Algorithm for Resource Allocation, " *Canadian Journal of Civil Engineering*, vol. 20, no. 2, pp. 180-188, 1993.
Available: DOI: 10.1139/l93-023

[19] S. Hougardy, "The Floyd-Warshall algorithm on graphs with negative cycles," *Information Processing Letters*, vol. 110, no. 8, pp. 279-281, 2010.

[20] Ramadiani, D Bukhori, Azainil and N Dengen, "Floyd-Warshall Algorithm to Determine the Shortest Path Based on Android," in Proceedings of 1st International Conference on Tropical Studies and Its Application (ICTROPS), IOP Conf. Series: Earth and Environmental Science, 012019, 2018. Available: doi :10.1088/1755-1315/144/1/012019 .20

[21] R.W. Floyd, "Algorithm 97: Shortest path," *Communications of the ACM*, vol. 5, no. 6, pp.5–6, 1962.

[22] S. Warshall, "A theorem on boolean matrices," *Journal of the ACM* , vol. 9, no. 1, pp. 11–12, 1962.

[23] Aini and A. Salehipour, "Speeding up the Floyd Warshall algorithm for the cycled shortest path problem," *Applied Mathematics Letters*, vol. 25, no. 1, pp.1–5, 2012