STUDIA UNIV. BABEŞ–BOLYAI, INFORMATICA, Volume **LVII**, Number 4, 2012

GENERATING PERT NETWORK WITH TEMPORAL CONSTRAINTS

NASSER EDDINE MOUHOUB(1) AND SAMIR AKROUF(2)

Abstract. A scheduling problem is organizing in time a set of activities, so as to satisfy a set of constraints and optimize the result. The temporal constraint modifies the project scheduling, therefore in loses its characteristics. Our objective is to solve this problem by finding the various types of temporal constraints then modeling them by using graphs. Furthermore we apply a technique for transforming an AoN graph (Activities on Nodes) which is unique and contains a significant number of arcs. This graph is not preferred by practitioners of project management. We transform the AoN graph into an AoA graph (Activities on Arcs) which contains fewer arcs and is preferred by practitioners of project management. In this paper we present some concepts of line graphs and an illustrative example of the proposed method.

1. INTRODUCTION

In project scheduling problems, operational monitoring activities are very important. The project manager draws up the schedule by using graphs. The drawing of AoN (Activities on Nodes) graph also called potential graph or French graph is easy because of its uniqueness despite the large number of arcs it generates. Besides the AoA (Activities on Arcs) graph also called PERT network or American graph is more difficult because of the dummy arcs it generates. However, practitioners prefer to work with the AoA graph because it is easy to read; each activity is represented by an arc. Specialists who insist on using the AoA graph have a number of arguments to justify their choice. This is why according to Fink et al. [1], it is more concise. Furthermore, Hendrickson et al. [2] explains that it is close to the famous Gantt diagram. According to Cohen et al. [3], the structure of the PERT network is much more suitable for certain analytical techniques and optimization formulations.

Received by the editors: June 3, 2012.

2010 *Mathematics Subject Classification.* 90B35, 90B10.

1998 *CR Categories and Descriptors.* G.2.2 [**Discrete Mathematics**]: Graph Theory – *Graph algorithms*.

Key words and phrases. AoA network, AoN network, PERT network, Project scheduling, Temporal constraint.

However, the major disadvantage of this method is in the existence of dummy arcs (see figure 3. (a) and (b)). Their number is likely to be significantly high especially if the size of the network is too large, thus the AoA graph is not unique. In this paper, we focus on finding a method to move from a simple graph (AoN graph) to an AoA graph that will be correct and will respect the scheduling table taking into account prior and temporal constraints. This method will be a draft for the construction of an algorithm that will achieve the transition from the AoN graph to the AoA graph taking into account the temporal constraints.

2. THE PROJECT SCHEDULING

The constraints to which are subjected the various activities, and contributing to the realization of the project, are of various types. We distinguish the potential constraints, disjunctive and cumulative constraints. The potential constraints are the following:

- The constraints of anteriority according to which an activity j cannot start before an activity i is finished, for example, the construction of the pillars follows the foundations
- Temporal constraints which means that a given activity i cannot begin before an imposed date, or that it can be completed after an imposed date.

The problem of scheduling with only potential constraints is called project scheduling problem. Lacomme and al. Paper [4] presents the two conventions which are used in practice for displaying project networks:

2.1. Activity on node graph (AoN).

Each activity is represented by a node in the network. A precedence relationship between two activities is represented by an arc or link between the two (see Figure 1). This graph is called the Activity on Node graph (AoN graph).

FIGURE 1. The activity u, with duration $t(u)$, precedes the activity v.

2.2. **Activity on arc graph (AoA).** Each activity is represented by an arc in the network. If activity u must precede activity v, there is an arc from u to v. Thus, the nodes represent events or "milestones" (e.g., "finished activity u") like in Figure 2. This graph is called the Activity on Arc graph (AoA graph) or PERT graph.

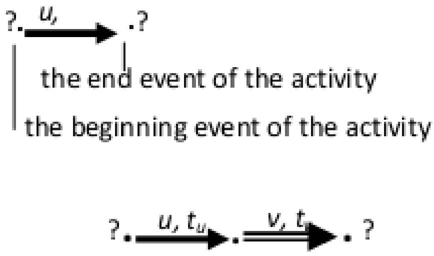

FIGURE 2. The activity u precedes the activity v in PERT graph.

The representation of Table 1 (Figure 3. (a)) in PERT graph, is false; to correct it we introduce an additional activity of duration 0 which does not influence over the total duration of the project. This activity is called a dummy activity. We then modify the table (see Table 2) of scheduling and the PERT graph (figure 3. (b)) the drawing will be easy.
The introduction of the dummy activities gives the possibility to solve certain situations and raise ambiguities. They do not take in consideration any material or financial mean [5].

Code	Predecessors
c	a,b
d	b

TABLE 1. An-under table of precedence of c, d.

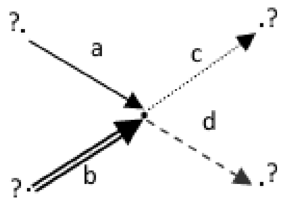

FIGURE 3.(a). The problem of representation in PERT graph.

Code	Predecessors
c	a,f
d	b
f	b

TABLE 2. The new under-table of precedence of C, D and F.

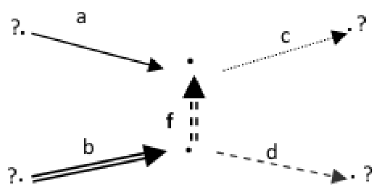

FIGURE 3. (b). Introduction of the dummy activity f and the representation in PERT graph.

For more details for these two methods and their differences, the reader can refer to [6], [7] and [8]. The study of this field is not only in order to facilitate the task to experts, but also for theoretical interests, these are always renewed by the researchers. We can remind that the durations are not mentioned on the different graphs. These durations can be uncertain. For this precise case, there are more details in [9], [10] and [11].

3. THE LINE GRAPH OF GRAPH

Let $G = (X, U)$ a simple or multiple digraph. We build starting from G a graph or line graph noted $L(G)$, called line graph or line digraph of G as follows: The arcs of $L(G)$ are in bijective mapping with the nodes of G for simplicity reasons; we give the same name to the arcs of G and the corresponding nodes of $L(G)$. Two nodes u and v of $L(G)$ are connected by an arc of u towards v if and only if the arcs u and v of G are such as the final end of u matches with the initial end of v, i.e. $T(u) = I(v)$ [12] (see Figure 4).

3.1. **Example.** Let G the following directed acyclic graph be (Figure 4): By definition, any directed graph G admits a unique line graph $L(G)$. On the other hand, two non isomorphs directed acyclic graphs can have the same line graph.

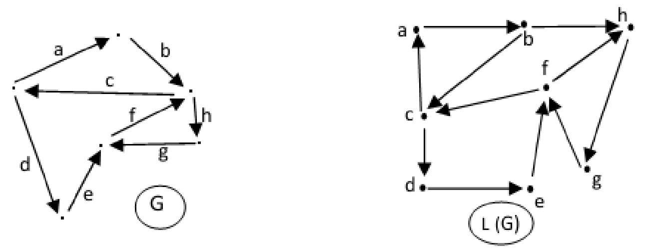

FIGURE 4. A graph G and his line graph $L(G)$.

3.2. **The opposite problem.** We suppose the following opposite problem: Being given a directed acyclic graph H, is it the line graph of any directed acyclic graph? In other words, does there exist a graph G such as $L(G)$ is isomorphs with H, where $H = L(G)$?

G admits a configuration “Z” if G contains four nodes a, b, c and d such as if (a, c), (b, c) and (b, d) are arcs of G, then (a, d) is not an arc of G. With an only aim of simplicity, one will give the name of bar of “Z” to arc (b, c) (see Figure 5) [13].

Configuration “Z” appears when two nodes have common successors and no common successors or by symmetry when two nodes have common predecessors and no common predecessors.

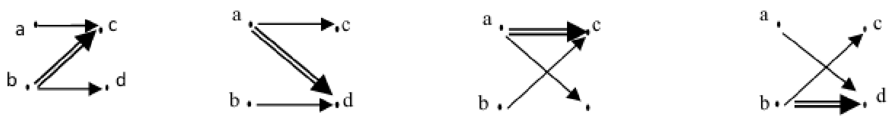

FIGURE 5. The configuration “Z” and his forms.

3.3. **Theorem.** The line graphs have been studied but we will present, in this section, the features in which we are interested and obtained from [12]. H is the line graph of a directed acyclic graph if:

- H does not contain any “Z” configuration.
- Arcs of H can be partitioned in a complete bipartite $B_i = (X_i, Y_i), i = 1, ..., m,$ such as $X_i \cap X_j = \emptyset$ and $Y_i \cap Y_j = \emptyset, \forall i \neq j$.

The bipartite B_i of H are then in a bijection with the nodes also noted B_i which are neither sources nor well, two nodes B_i and B_j of G being connected by an arc from B_i towards B_j if and only if the complete bipartite B_i and B_j of H are such as $Y_i \cap X_j = \emptyset$ (figure 6).

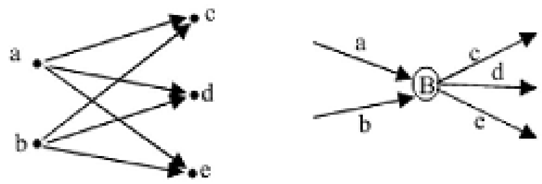

FIGURE 6. A complete bipartite B of G and the star of G.

- *H* does not contain any configuration "Z" and any pair of nodes having common successors has all their common successors.
- Any pair of nodes having common predecessors has all their common predecessors. For more details on this theorem, the reader can refer to [12].

Thus, *H* is not the line graph of any directed acyclic graph if and only if there is a pair of nodes having common successors and no common successors or common predecessors and no common predecessors [5].

4. GENERATING AOA GRAPH

Because of the facility of the use of AoA graph, we must concentrate our efforts on the study of the possibility of transforming the AoN graph (a significant number of arcs) to AoA graph (a reduced number of arcs).
So, we want to know how to transform the graph H (which is an AoN graph) in order to get a new graph which is the line graph (AoA graph). According to [5], the difficulty which arises is to know if H does contain "Z" configurations or not? If it does not, it is a line graph and the transformation is immediate (as in Figure 7).

Code	Predecessors
b_1	$a_1, ..., a_m$
.	..
b_n	$a_1, ..., a_m$

TABLE 3. The sub-table of anteriorities.

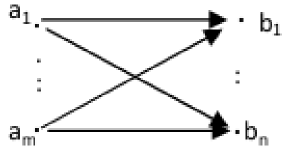

FIGURE 7. (a): The complete bipartite *B* in the AoN graph.

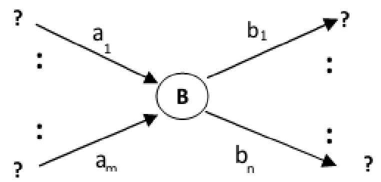

FIGURE 7. (b): The star B of the corresponding AoA graph.

But if it contains "Z" configurations, we have to eliminate the bare from each "Z" preserving the constraints of succession. We then introduce, in the AoN graph, a dummy arc f in every "Z" (Figure 8). The introduction of the

FIGURE 8. "Z" configuration, his corresponding transformation in AoN graph and the partition of the complete bipartite.

dummy arcs aims to eliminate all the "Z" configurations from the AoN graph, the constraints remain unchanged. We should recall that the dummy arcs are not necessary in the AoN graph but are introduced only to build AoA graph. For more details on this transformation from AoN graph to AoA, the reader can refer to [5] and [13].

5. THE TEMPORAL CONSTRAINTS

The temporal constraint is a time allocation constraint. She comes from imperative management constraints such as the supply availability or time delivery, etc.
It specifies the time interval (or semi-interval) during which it is possible to perform or carry out an activity. These constraints are often due to availability of stakeholders (human resources): for example a company which produces frames can only intervene between June 15 and August 31 [1].

The temporal constraint affects the project scheduling and changes. He no longer has the characteristics of the project scheduling. The problem therefore, is to find a way or a technique to normalize the situation and bring it back to the project scheduling. In the following, we will propose an original method which allows us to model the temporal constraints and include them in project scheduling.

We can classify the most important temporal constraints into six types and by adding the precedence constraint they become seven:

C1: Activity A starts t time units before the work begins.
C2: Activity A can only start t time units after the beginning of work.
C3: Activity B must start t time units after the end of activity A.
C4: Activity B starts a fraction of time a/b after the start of activity A (a < b).
C5: Activity B must start t time after the start of activity A ($t < t_A$).
C6: Activity A must start before time t.
C7: Activity B must immediately follow the activity A.

5.1. **Modeling temporal constraints.** In project scheduling, which is a particularly in an AoN graph, incident arcs outside a node (that is to say an activity) have the same value.

The presence of temporal constraints in the graph AoN violates this property, which makes solving the project scheduling impossible. Calculating dates and critical path research ... also become impossible.

Modeling by using graphs can solve this problem. We will present in the following a new technique that allows handling such constraints.

The Figure 9 gives the unique representation of these constraints in the graph AoN. It is clear that the values on the arcs incident to a node outside are different (see the example in Figure 11. (a)). Here we leave the project scheduling.

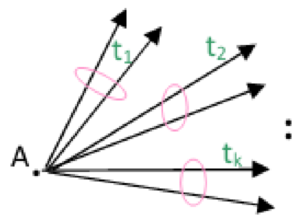

FIGURE 9. Main temporal constraints in AoN graph.

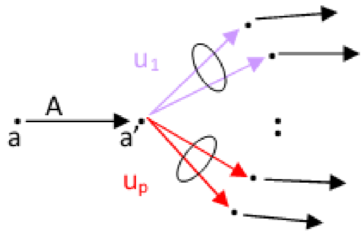

FIGURE 10. (a). Representation of (C2) and (C3) constraints in AoA graph.

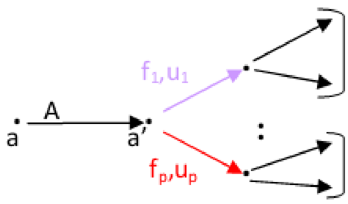

FIGURE 10. (b). Another Representation of (C2) and (C3) constraints in AoA graph with less activities than in (a).

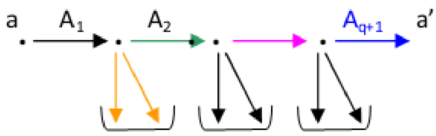

FIGURE 10. (c). Representation of (C4) and (C5) constraints in AoA graph.

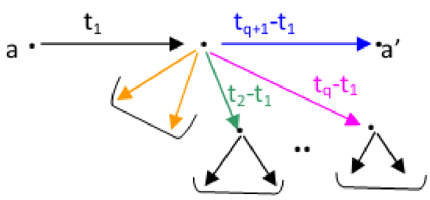

FIGURE 10. (d). Representation of (C4) and (C5) constraints in AoA graph with the same number of activities as in (c).

Figure 10. (a) shows that in the AoA graph, each activity coming after the activity A, has its own dummy arc ui. This representation is poor because the number of dummy arcs may be very important, which clutters the graph.

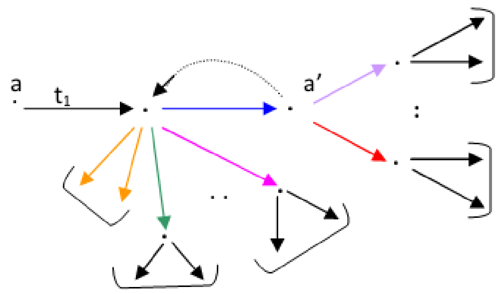

FIGURE 10. (e). type of constraints in PERT graph combining (c) and (d).

A better representation (Figure 10. (b)) consists in gathering several dummy arcs succeeding the real activity A and which have the same value in a single dummy activity. Note that the dummy arc in this context is not of zero duration. She is introduced to solve this problem and introduce the time constraints in the project scheduling.

For constraints of type (C4) and (C5), we notice that both starts after the beginning of activity A. Representation in AoA graph implies the segmentation of A into several tasks, in the general case (A = $A_1 + A_2 + ... + A_{q+1}$). Two models of these constraints are possible (figure 10 (c). and 10. (d)). We note that the representation of figure 10. (d) is more convenient.

Finally, we can combine the figure 10. (b) and 10. (d) keeping in mind the idea of minimizing dummy arcs.

In conclusion, to arrive to figure 10. (e) we must modify in the AoN graph, the arcs incident outside a vertex and who do not have the same value, by introducing dummy arcs of length 0 in order to partition the complete bipartite graph with AoN graph. All these combinations lead us to the changes made in the two graphs AoA and AoN respectively (figure 11)

Correspondence between the representations of temporal constraints in AoN graph and in AoA graph is our goal, we modify figure 9 in figure 11. (a) and figure 10. (e) in figure 11. (b).

The introduction of activities $f_2, ..., f_k$ of times $t_2 - t_1, ..., t_k - t_1$ has the advantage of giving the same value to the arcs of the same initial node in the graph AoN. There is no difficulty to verify that the arcs of the graph (Figure 11. (a)) are partitioned into a complete bipartite graph and that is the associate graph of the graph in Figure 11. (b).

For example, Let A be an activity of duration 5 time units. Suppose that: A precedes B,

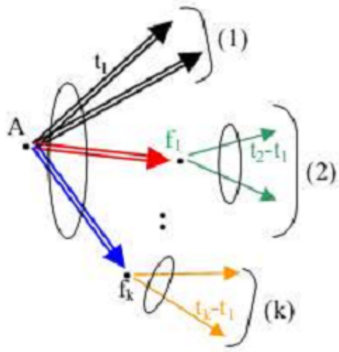

FIGURE 11. (a). modification in AoN graph.

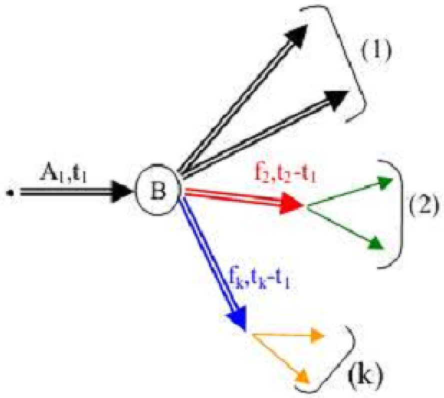

FIGURE 11. (b). Activity A is subdivised as $(\mathrm{A}_1, \mathrm{f}_1)$ in AoN graph. Arcs of the same initial node have the same value.

B1 and B2 can not start a unit of time after the start of activity A,
B3 and B4 begin only 4 time units after the start of A
B5 can not start until A is 3/4 finished,
B6 and B7 begin only 6 time units after the end of A.
In AoN graph, let us draw the arcs leaving the node A (Fig.12.):

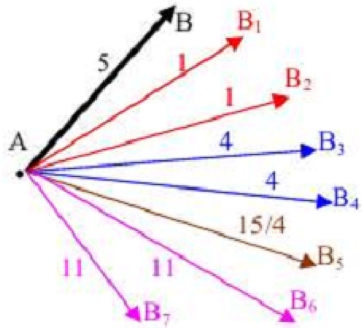

Figure 12. (a). No modification in AoN graph.

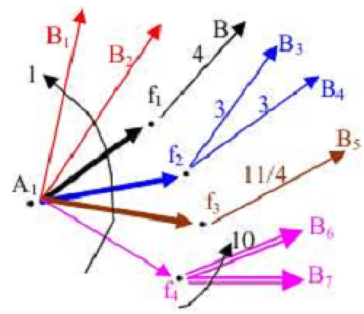

Figure 12. (b). is subdivised as (A1, f1) in AoN graph. Arcs of the same initial node have the same value.

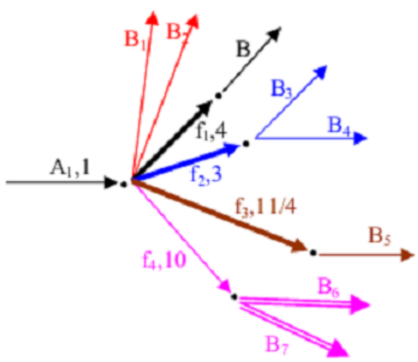

Figure 12. (c). No modification in AoN graph.

To illustrate what we have seen since the beginning of this paper and to construct AoA graph from AoN graph taking into account temporal constraints, we consider the following example:

5.2. **Example.** The Table 4 gives the precedence constraints.

Activity	Description	Predecessors	Duratiion
A	Site clearing	6	-
B	Removal of trees	5	-
C	General excavation	8	-
D	Grading general area	4	A
E	Excavation for trenches	3	A, B, C
F	Placing formwork and reinforcement for concrete	9	C
G	Installing sewer lines	2	D, F
H	Installing other utilities	8	E, F
I	Pouring concrete	5	E, F

Table 4. Precedence relations and durations for a nine activity project example.

Temporal constraints are:

- B can only start 3 time units after the beginning of the work.
- C can begin only after 7 time units the work begins.
- E begins when C is executed to $3/4$
- G starts 4 units of time after the end of E.

The graphs in Figure 13. (a, b, c, d) show the changes in the AoN graph, then the AoA graph construction:

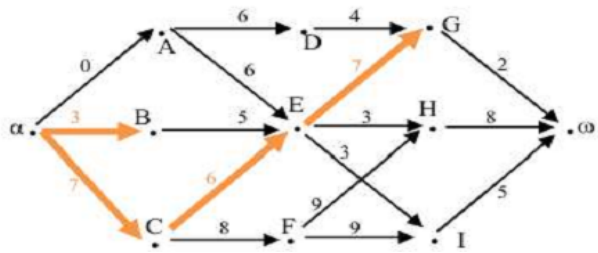

Figure 13. (a). graph from the schedule table (see Table 4.). Arcs in bold represents temporal constraints.

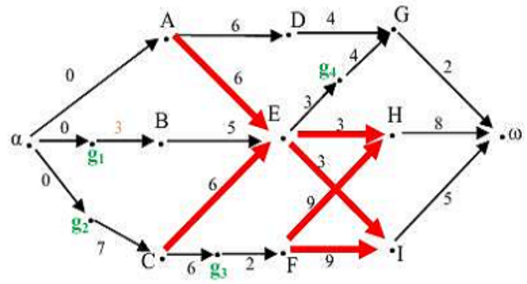

Figure 13. (b). AoN graph whose arcs have the same initial node have the same value. The dummy arcs from temporal constraints g_i: activities α, C, E are divided in two activities. "Z" bars are in bold..

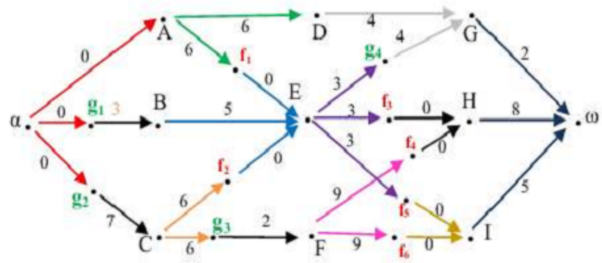

Figure 13. (c). The AoN graph with no "Z" configuration and whose nodes are reorganized into levels. We can verify that the arcs can be partitioned into complete bipartite.

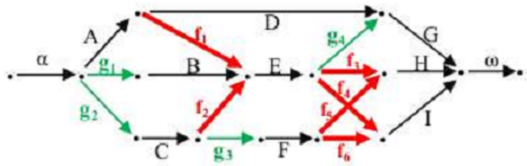

Figure 13. (d). AoA graph of Table 4. Activities duration not included (the fi "in bold" have duration zero).

5.3. **Discussion.** The algorithm inspired from this method finishes since the loop is carried out only when there is a 'Z' sub graph or a temporal constraint. The number of 'Z' in AoN graph is known and finite. Also, the number of temporal constraints is known. The rest of the algorithm is a succession of simple instructions. The complexity of the algorithm is polynomial ($O(n^4)$).

6. CONCLUSION

This work has introduced graphs associates in project scheduling problems, with or without the presence of 'Z' in the graph for AoN, for AoA graph construction. He also used the modeling of temporal constraints that can be included in the project scheduling when the resolution becomes easier thus the calculation of dates at the earliest, at the latest, free margins, the critical path, etc becomes possible by applying Bellman algorithm.

This work opens up perspectives, such as searching the minimal PERT graph network in terms of dummy arcs is NP-hard or in terms of nodes. The project scheduling with limited resources can be viewed by using modeling with graphs.

References

[1] Fink, G.: Recherche oprationnelle et rseaux, Lavoisier, Paris,(2002).

[2] Henderckson, C., Project Management for Construction, Department of Civil and Environmental Engineering, Carnegie Mellon University, Pittsburgh, PA l52l3, Version 2.2,08.

[3] Cohen, Y., Sadeh, A.: A New Approach for Constructing and Generating AoA Networks, Journal of computer science, 1-1, 2007. *http://www.scientificjournals.org/journals2007/articles/1049.htm*

[4] Esquirol, P. and P. Lopez, P .: l'ordonnancement, ECONOMICA, Paris, France, ISBN 2-7178-3798-1, 1999.

[5] Mouhoub, N. E., Belouadah, H. and Boubetra A.: Algorithme de construction d'un graphe Pert partir d'un graphe des potentiels donn, STUDIA UNIV. BABES BOLYAI, INFORMATICA, Volume LI, Number 2, 2006.

[6] Bernard ROY, Algbre moderne et thorie des graphes, tome 2, Fascicule 3, Problmes d'ordonnancement et ensembles de potentiels sur un graphe, DUNOD, Paris, France, 1970.

[7] A. Haga, Tim O'keefe, Crashing PERT networks: a simulation approach, 4th International conference of the Academy of Business and Administrative Sciences, Quebec City, Canada, July 12-14, 2001.

[8] F. Bacchus and F. Kabanza, Planning for Temporally Extended Goals. Annals of Mathematics and Artificial Intelligence, 22(1-2) :5-27, 1998.

[9] P. Morris, N. Muscettola, and T. Vidal, Dynamic Control of Plans with Temporal Uncertainty, In Proceedings of the 17th International Joint Conference on Artificial Intelligence, IJCAI-01, pages 494-502. Morgan Kaufmann, 2001.

[10] I. Tsamardinos, T. Vidal, and M. E. Pollack, CTP: A New Constrained-based Formalism for Conditional Temporal Planning, Constraints journal, special issue on Planning, 8(4):365-388, 2003.

[11] Neal SAMPLE, Pedram KEYANI, Gio WIEDERHOLD, Scheduling Under Uncertainty: Planning for the Ubiquitous Grid, Computer Science Department, Stanford University, Stanford CA 94305, 2002.

[12] C. Heuchenne, Sur une certaine correspondance entre graphes, Bull. Soc. Roy. Sci. Liege, 743-753, 33, 1964.

[13] Mouhoub, N. E., Benhocine, A. and Belouadah, H.: A new method for constructing a minimal PERT network, (APM) Applied Mathematical modeling, Elsevier ISSN: 0307904X, Vol. 35, Issue: 9, 4575-4588, 2011.

(1) Department of Computer Science, Faculty of Mathematics and Informatics, Bordj Bou Arreridj University, El Anasser, 34030, Algeria
E-mail address: mouhoub_n@yahoo.fr

(2) Department of Computer Science, Faculty of Mathematics and Informatics, LMSE, Bordj Bou Arreridj University, El Anasser, 34030, Algeria
E-mail address: samir.akrouf@gmail.com