HAL
open science

The PERT Problem with Alternatives: Modelisation and Optimisation

Fabrice Chauvet, Jean-Marie Proth

To cite this version:

Fabrice Chauvet, Jean-Marie Proth. The PERT Problem with Alternatives: Modelisation and Optimisation. [Research Report] RR-3651, INRIA. 1999, pp.27. inria-00073021

HAL Id: inria-00073021

https://inria.hal.science/inria-00073021v1

Submitted on 24 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

The PERT Problem with Alternatives: Modelisation and Optimisation

Fabrice Chauvet - Jean-Marie Proth

N° 3651
Mars 1999

THÈME 4

Rapport de recherche

Les rapports de recherche de l'INRIA
sont disponibles en format postscript sous
ftp.inria.fr (192.93.2.54)

si vous n'avez pas d'accès ftp
la forme papier peut être commandée par mail :
e-mail : dif.gesdif@inria.fr
(n'oubliez pas de mentionner votre adresse postale).

par courrier :
Centre de Diffusion
INRIA
BP 105 - 78153 Le Chesnay Cedex (FRANCE)

INRIA research reports
are available in postscript format
ftp.inria.fr (192.93.2.54)

if you haven't access by ftp
we recommend ordering them by e-mail :
e-mail : dif.gesdif@inria.fr
(don't forget to mention your postal address).

by mail :
Centre de Diffusion
INRIA
BP 105 - 78153 Le Chesnay Cedex (FRANCE)

The PERT Problem with Alternatives: Modelisation and Optimisation

Fabrice CHAUVET* and Jean-Marie PROTH* and **

ABSTRACT

Management of projects often requires decisions concerning the choice of alternative activities. The completion time of the whole project (i.e. the makespan) is computed subsequently. In this paper, we aim at selecting the activities and computing the makespan simultaneously. This problem is referred to as PERT Problem with Alternatives (PPA). The corresponding model is similar to a conventional PERT graph, except that two types of nodes are introduced to represent either the choice between activities, or the fact that a set of activities should be completed before starting a subsequent set of activities. In this paper, we analyse the PPA and we propose a pseudo-polynomial algorithm to solve it.

KEYWORDS

PERT Network, Shortest Path, Alternative Activities, Makespan Optimization.

* INRIA Lorraine, Technopôle Metz 2000, 4 rue Marconi, 57070 Metz, France
** Institute for Systems Research, University of Maryland, College Park, MD 20742, USA

This work has been supported by the INTAS Project 96-820.

Problème PERT avec Alternatives : Modélisation et Optimisation

Fabrice CHAUVET* et Jean-Marie PROTH* et **

RESUME

La gestion de projet demande souvent que soient choisies les activités à exécuter dans un ensemble d'activités équivalentes. La durée requise pour exécuter le projet, c'est-à-dire le "makespan", est calculée à partir de ce choix. Dans cet article, nous cherchons à sélectionner les activités et à calculer le makespan simultanément. Nous appelons ce problème Problème PERT avec Alternatives (PPA). Le modèle correspondant à ce problème est similaire au graphe PERT conventionnel, excepté que deux types de nœuds sont utilisés pour représenter soit le choix entre plusieurs activités alternatives, soit le fait qu'un ensemble d'activités doit être terminé avant qu'un autre ensemble d'activités puisse commencer. Dans cet article, nous analysons le PPA et proposons un algorithme pseudo-polynomial pour le résoudre.

MOTS-CLEFS

Modèle PERT, Plus court chemin, Activités alternatives, Makespan.

* INRIA Lorraine, Technopôle Metz 2000, 4 rue Marconi, 57070 Metz, France
** Institute for Systems Research, University of Maryland, College Park, MD 20742, USA

Ce travail a été financé par le projet INTAS n° 96-820.

1. INTRODUCTION

In this paper we introduce and analyze PERT networks modeling projects with alternative operations. Recall that any PERT network is a finite directed graph G in which arcs represent *activities*, or *operations*, and nodes represent *events*, that is, starting and ending points of activities. j and i denote respectively the origin and the ending node of arc (j,i). The length of each arc is given; it is assumed to be either a positive or negative rational number, or zero.

In the traditional PERT models (see, for instance, [8, 9, 12, 13]), all activities are assumed to be of the *and*-type when the goal is to find the longest path. An arc (j,i) represents an *and*-type activity if *all* the activities represented by an arc (j',i) (whose ending node is i) must be completed before any activity represented by an arc (i,k) (whose starting node is i) can begin. However, in many practical cases, a project may include, along with the *and*-type activities, another type of activities known as *alternatives*, or *or*-type activities. An arc (j,i) represents an *or*-type activity if *at least one* activity represented by an arc (j',i) (whose ending node is i) must be completed before any activity represented by an arc (i,k) (whose starting node is i) can begin. There are many real-life applications in manufacturing, communication and logistic that are modeled by means of networks with alternative operations (see, for example, [1, 4-7, 10-11]).

The type of project management problem presented in this paper is referred to as *PERT Problem with Alternatives* (*PPA*). Given a project containing both *and*-type and *or*-type activities, with two fixed nodes denoted respectively by s (*start*) and f (*final*), the problem is to find the earliest starting time of each one of the activities so that the completion time (i.e. the makespan) of the entire project is minimum. The mathematical model of this problem is a combination of the classical PERT (longest path) problem and the routing (shortest path) problem.

Not much have been done to date for efficiently solving scheduling problems with alternative activities. Dinic [4] described a polynomial-time algorithm for solving the PERT Problem with Alternatives (PPA) on a bipartite graph with arcs of (strictly) positive lengths. This algorithm has been extended (see [2]) to find the minimum makespan for any graph (not necessary bipartite) with (strictly) positive arc lengths. In [2], several well-solvable cases of the problem and a powerful decomposition algorithm running in polynomial time are presented. This algorithm is applicable for solving many real-life problems. Other approaches as integer-programming, heuristic and branch-and-bound have been studied to treat alternative activities in network systems. Kusiak and Finke [7] were apparently the first who studied a problem with alternative process plans. They proposed an integer mathematical programming model to measure the total cost. Nasr and Elsayed [10] considered the problem of minimizing the mean flow time in a machining system with alternative routings and developed a decomposition method for solving the mixed integer formulation of the problem. Wilhelm and Shin [15]

effectively applied a linear programming model and examined the influence of alternative operations on the performance of flexible manufacturing systems. Ahn, He and Kusiak [1] and Gere [5] demonstrated that scheduling manufacturing systems taking into account simultaneously alternative process plans and alternative operations permit to increase the throughput rate by a better utilization of scarce resources; these authors developed heuristic algorithms for scheduling alternative operations and studied the effect of alternative operations on the performance of schedules generated by different heuristic rules. Iwata, Murotsu and Oba [6] proposed a branch-and-bound technique to determine a minimum total production time in a flexible system with alternative operations. A branch-and-bound algorithm was developed also by Pan and Chen [11] to minimize the makespan in a two-machine flowshop with alternative operations. In an early paper by Stecke and Solberg [14], a simulation experiment is conducted for the real-time control of an FMS with alternative operations.

The present paper is devoted to PERT Problem with Alternatives (PPA) represented by graphs whose *arc lengths are rational and of any sign*. We will focus on the analysis of its properties. Notice that the optimal solution is neither defined by the longest path (as in the PERT/CPM models with *and*-type activities) nor by the shortest one (as in the routing problem), but rather recursively as a combination of paths of the both types.

We introduce the formalisation of the problem in Section 2. Some basic definitions are given in Section 3. In Section 4, we describe the properties of a feasible solution and an algorithm to obtain such a feasible solution. In Section 5, we give a necessary and sufficient condition for a solution to be optimal and derive an algorithm to obtain the optimal solution. Section 6 is the conclusion.

2. PROBLEM FORMULATION

Notations

Let $G=(V, E)$ be a PERT network, that is, a finite directed graph, where V is the set of nodes and E the set of arcs. The network describes a project in which arcs represent activities and nodes represent events (i.e. starting and ending points of the project activities). Without loss of generality, we assume that graph G is connected (otherwise, we could process each one of its connected component independently). The lengths $l(e)$ of arcs $e \in E$ are given. Each arc length is assumed to be a rational constant (positive, negative or zero). A positive arc length denotes the duration of the activity corresponding to the arc. An arc of positive (respectively negative) length $l(j,i)$ ending with an *and*-node i (respectively *or*-node i) means that event i will occur not sooner than $l(j,i)$ units of time after event i occurs. Notice that events i and j may belong to different activities. An arc of positive (respectively negative) length $l(j,i)$ ending with an *or*-node i (respectively *and*-node i) means that event i must occur at most $l(j,i)$ units of time after event j.

As mentioned in introduction, we consider PERT networks with two types of activities, that is *or*- and *and*-activities. Corresponding to these two types of activities, two kinds of nodes are introduced in the corresponding network model. In the remainder of this paper, the ending nodes of *or*-type activities are called *or-nodes*, and the ending nodes of *and*-type activities, *and-nodes* (see Figure 1).

(a) Since T_a and T_b are *or*-activities, at least one of the activities T_a or T_b must be completed before T_c and T_d start. The ending node of T_a and T_b is an *or*-node.

t_a, t_b, t_c and t_d being the starting times of activities T_a, T_b, T_c and T_d respectively, the following inequalities must be satisfied:

$$t_c \geq \min\{t_a+l(T_a);\ t_b+l(T_b)\}, \text{ and}$$
$$t_d \geq \min\{t_a+l(T_a);\ t_b+l(T_b)\}.$$

(b) Since T_a and T_b are *and*-activities, both activities T_a and T_b must be completed before T_c and T_d start. The ending node of T_a and T_b is an *and*-node.

t_a, t_b, t_c and t_d being the starting times of activities T_a, T_b, T_c and T_d respectively, the following inequalities must be satisfied:

$$t_c \geq \max\{t_a+l(T_a);\ t_b+l(T_b)\}, \text{ and}$$
$$t_d \geq \max\{t_a+l(T_a);\ t_b+l(T_b)\}.$$

Figure 1. An *or*-node and an *and*-node.

We assume that there is only one node without predecessors, called the *start* node of the project and denoted by s. (Notice that if there are several nodes having no predecessor, it is always possible to add one *new_start* node and connect this node with the nodes having no predecessors with arcs of zero length). Using a similar argument, we assume that there is only one node without successors, called the *final* node of the project and denoted by f. Node s represents the beginning of the project while node f represents its end. We also assume that there exits a (directed) path from node s to any node in the graph.

The set of *or*-nodes (respectively *and*-nodes) is denoted by R (respectively A); $V=\{s\}\cup R\cup A$. Notice that node f belongs to either R or A. Let us denote by $Pred(i)$ the set of predecessors of node i, where $i\in R\cup A$: $Pred(i)=\{j/\ j\in V,\ (j,i)\in E\}$. If there are several arcs starting from the same node and ending at an *or*-node (respectively an *and*-node), the arc of the lowest (respectively greatest) value is the only one which will be considered. Finally, we can assume, without loss of generality, that there exists at most one arc between any two nodes.

A parameter t_i is associated with each node i. If i is an *and*-node, t_i is an upper bound of the completion times of *all* the activities represented by arcs whose ending node is i. If i is an *or*-

node, t_i is an upper bound of the completion time of *at least one* activity represented by arcs whose ending node is i. Without loss of generality, we assume that the project starts no sooner than time 0, i.e. $t_s \geq 0$.

Modelisation

Our goal is to find, for each node i, the times t_i such that:

$$\begin{Bmatrix} t_i \geq \min\limits_{j \in Pred(i)} \{t_j + l(j,i)\} & \text{if } i \in R \\ t_i \geq \max\limits_{j \in Pred(i)} \{t_j + l(j,i)\} & \text{if } i \in A \end{Bmatrix}.$$

and such that the completion time of the entire project (i.e. the time t_f assigned to node f) is minimum. This optimal solution is provided (see [2]) by solving the following problem P_{INEQ}:

Problem P_{INEQ} : minimize $\sum\limits_{i \in V} t_i$

subject to

$$\begin{Bmatrix} t_i \geq 0 & \text{if } i = s \\ t_i \geq \min\limits_{j \in Pred(i)} \{t_j + l(j,i)\} & \text{if } i \in R \\ t_i \geq \max\limits_{j \in Pred(i)} \{t_j + l(j,i)\} & \text{if } i \in A \end{Bmatrix} \qquad (S_{\text{INEQ}})$$

A set $\{t_i\}_{i \in V}$ which satisfies (S_{INEQ}) is a feasible solution of P_{INEQ}. The optimal solution of P_{INEQ}, called *earliest times* solution, is unique (see [2]) and is also the unique optimal solution of the associated problem P_{EQ} defined as follows:

Problem P_{EQ} : minimize $\sum\limits_{i \in V} t_i$

subject to

$$\begin{Bmatrix} t_i = 0 & \text{if } i = s \\ t_i = \min\limits_{j \in Pred(i)} \{t_j + l(j,i)\} & \text{if } i \in R \\ t_i = \max\limits_{j \in Pred(i)} \{t_j + l(j,i)\} & \text{if } i \in A \end{Bmatrix} \qquad (S_{\text{EQ}})$$

Since the two problems P_{INEQ} and P_{EQ} have the same optimal solutions if one of them has an optimal solution, we focus on P_{EQ} and we present the proprieties of the feasible and the optimal solutions of P_{EQ} and some algorithms to obtain these solutions.

At the end of the computation of the optimal solutions, some arcs ending with an *or*-node can be deleted since they represent an activity wich will not be performed. For example, assume that $\{t_i\}_{i \in V}$ is the optimal solution of problems P_{INEQ} and P_{EQ}. In this case, if node f is

an *and*-node then all the operations (i,f) ending with f must be performed. Otherwise, f is an *or*-node and at least one operation (j,f) such that $t_f = \min_{j \in Pred(f)} \{t_j + l(j,f)\}$ must be performed.

3. SATURATED ARCS, NODES AND CIRCUITS

Definitions 1

Recall that a (directed) path is a sequence of different arcs $\{e_1;e_2;\ldots;e_q\}$ of E such that the end node of e_k is the origin node of e_{k+1} for each $k=1, 2, \ldots, q-1$. A path is a *circuit* if it begins and ends with the same node (i.e. the end node of e_q is the origin node of e_1).

Since there exists only one arc between two nodes, it is possible to define a path (resp. a circuit) either by a sequence of arcs, or by a sequence of nodes. A circuit whose length is equal to zero, is called a *null-circuit*. A circuit whose length is strictly positive (resp. strictly negative) is called a *positive circuit* (resp. *negative circuit*).

An arc (j_0,i_0), where j_0 and i_0 denote respectively the origin and the ending node of the arc, is called *saturated* by a solution $\{t_i\}_{i \in V}$

if $t_{i_0} = t_{j_0} + l(j_0,i_0)$ and if $\begin{cases} t_{i_0} = \min_{j \in Pred(i_0)} \{t_j + l(j,i_0)\} & \text{if } i_0 \in R \\ t_{i_0} = \max_{j \in Pred(i_0)} \{t_j + l(j,i_0)\} & \text{if } i_0 \in A. \end{cases}$

A node is called *saturated* by a solution $\{t_i\}_{i \in V}$ if there exists a saturated arc which ends with this node. A solution feasible to P_{EQ} saturates all the nodes of the graph.

A path is called *saturated* if all the arcs belonging to this path are saturated.

By adding sequentially the latter equations along the path, we obtain the following property: for any path $\mathbf{P}_{i_1 \to i_{q+1}} = \{(i_1,i_2);(i_2,i_3);\ldots;(i_q,i_{q+1})\}$ saturated by a solution $\{t_i\}_{i \in V}$, we have $t_{i_{q+1}} - t_{i_1} = \sum_{k=1}^{q} l(i_k,i_{k+1})$. If the saturated path is a circuit , the length of a such saturated circuit must be equal to zero since, in this case, $t_{i_{q+1}} - t_{i_1} = t_{i_1} - t_{i_1} = 0$.

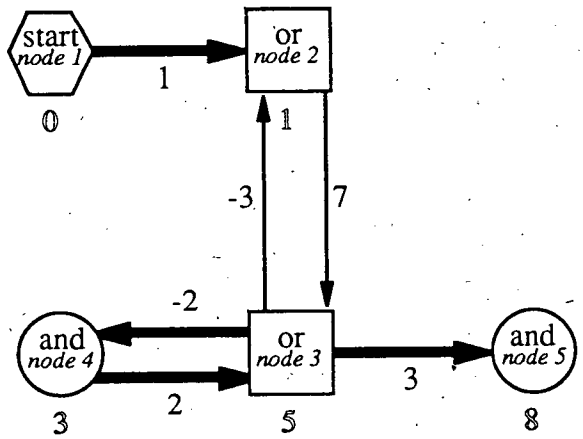
Lemma 1

If there exists a feasible solution of P_{EQ} such that there is no saturated path from node s to some other nodes, then there exists a saturated null-circuit in G.

Proof

Let $\{t_i\}_{i \in V}$ be a feasible solution of P_{EQ}, for which there is no saturated path from node s to some other nodes. Let us denote by $I \subset V$ the set of all such nodes. Since $\{t_i\}_{i \in V}$ is a feasible solution of P_{EQ}, for any node i of V, there exists a node $j \in Pred(i)$ such that the arc (j,i) is saturated. Since there is no path saturated by solution $\{t_i\}_{i \in V}$ from node s to any node of I, there is no saturated arc from a node of V/I to a node of I. Thus, for any node i of I, there

exists $j \in Pred(i) \cap I$ such that the arc (j,i) is saturated. Beginning with node $i_1 \in I$, and scanning the nodes of I one after another in this way, we obtain a saturated path whose nodes belong to I, say $\{(i_p,i_{p-1});(i_{p-1},i_{p-2});\ldots;(i_{q+1},i_q);\ldots;(i_2,i_1)\}$ in which $i_q=i_p$, because the set I is finite. The sequence $\mathscr{C}=\{(i_p,i_{p-1});(i_{p-1},i_{p-2});\ldots;(i_{q+1},i_q)\}$ is a circuit in which all arcs are saturated. Thus, the length of this saturated circuit is equal to zero (see above). QED



The lengths are indicated near the corresponding arcs and the saturated arcs are thick-lined. A feasible solution is given by the numbers located under the nodes. For the given feasible solution of P_{EQ}, there is no saturated path from node s to nodes 3, 4 and 5, while two nodes, 3 and 4 constitute a null-circuit.

Figure 2. Illustration of Lemma 1.

Theorem 1 *(Importance of null-circuit)*

If there exists a feasible solution of P_{EQ} which does not saturate any null-circuit in G, then:

(i) this feasible solution is the unique feasible solution of P_{EQ},

(ii) it is optimal for P_{EQ} and P_{INEQ}, and

(iii) there exists a path, saturated by this solution, from node s to any other node of G.

Remind that the length of any saturated path from node s to node f is equal to the makespan.

Proof

(i) To prove that a feasible solution of P_{EQ} which does not saturate any null-circuit in G is the unique feasible solution of P_{EQ}, we assume that two different solutions $\{t_i\}_{i\in V}$ and $\{x_i\}_{i\in V}$, feasible to P_{EQ}, exist. Thus, there exists $i_0 \in V$ such that $x_{i_0} \neq t_{i_0}$. We assume that:

$$x_{i_0} > t_{i_0} \tag{1}$$

(if not, it is always possible to rename the two solutions so that $x_{i_0} > t_{i_0}$). We then construct backward a directed path whose ending node is i_0. For k=0, -1, -2, ..., we select i_{k-1} such that:

$$t_{i_k} = t_{i_{k-1}} + l(i_{k-1},i_k) \text{ if } i_k \text{ is an } or\text{-node, and} \tag{2}$$

$$x_{i_k} = x_{i_{k-1}} + l(i_{k-1},i_k) \text{ if } i_k \text{ is an } and\text{-node.} \tag{3}$$

Since $\{t_i\}_{i\in V}$ and $\{x_i\}_{i\in V}$ are feasible solutions of P_{EQ}, all the nodes are saturated and i_{k-1} always exists.

The properties of this path are the following:

- We have $x_{i_k} > t_{i_k}$ for any $k \leq 0$. (4)
- The difference $x_{i_k} - t_{i_k}$ increases as k decreases,

that is $x_{i_{k-1}} - t_{i_{k-1}} \geq x_{i_k} - t_{i_k}$ for any $k \leq 0$. (5)

These properties are proved hereafter. For k=0, we have $x_{i_0} > t_{i_0}$. Assume that property (4) holds for $k \leq 0$. We prove that, in this case, property (4) also holds for k-1 and $x_{i_{k-1}} - t_{i_{k-1}} \geq x_{i_k} - t_{i_k}$.

If i_k is an *or*-node, then:

$t_{i_k} = \min_{j \in Pred(i_k)} \left\{ t_j + l(j, i_k) \right\} = t_{i_{k-1}} + l(i_{k-1}, i_k)$ (see relation (2)), and

$x_{i_k} = \min_{j \in Pred(i_k)} \left\{ x_j + l(j, i_k) \right\} \leq x_{i_{k-1}} + l(i_{k-1}, i_k)$ (since $\{x_i\}_{i \in V}$ is feasible for P_{EQ}).

Thus, since $x_{i_k} > t_{i_k}$, we have $x_{i_{k-1}} > t_{i_{k-1}}$ and $x_{i_{k-1}} - t_{i_{k-1}} \geq x_{i_k} - t_{i_k}$.

If i_k is an *and*-node, then:

$x_{i_k} = \max_{j \in Pred(i_k)} \left\{ x_j + l(j, i_k) \right\} = x_{i_{k-1}} + l(i_{k-1}, i_k)$ (see relation (3)), and

$t_{i_k} = \max_{j \in Pred(i_k)} \left\{ t_j + l(j, i_k) \right\} \geq t_{i_{k-1}} + l(i_{k-1}, i_k)$ (since $\{t_i\}_{i \in V}$ is feasible for P_{EQ}).

Thus, since $x_{i_k} > t_{i_k}$, we have $x_{i_{k-1}} > t_{i_{k-1}}$ and $x_{i_{k-1}} - t_{i_{k-1}} \geq x_{i_k} - t_{i_k}$.

Finally, properties (4) and (5) hold, whatever the type of node i_k, for $k \leq 0$. Hereafter, we use the above properties to derive the existence of at least one null-circuit which is saturated by both solutions.

The path under consideration is constructed on the finite set of nodes V. As a consequence, there exist two indexes $k_2 < k_1 \leq 0$ such that $i_{k_1} = i_{k_2}$. In other words, the backward construction of the path leads out onto a circuit. In this circuit, according to property (5), we have: $0 < x_{i_{k_1}} - t_{i_{k_1}} \leq x_{i_{k_1-1}} - t_{i_{k_1-1}} \leq \ldots \leq x_{i_{k_2+1}} - t_{i_{k_2+1}} \leq x_{i_{k_2}} - t_{i_{k_2}} = x_{i_{k_1}} - t_{i_{k_1}}$. As a consequence, these differences are equal to each other, and:

$$0 < x_{i_{k_1}} - t_{i_{k_1}} = x_{i_{k_1-1}} - t_{i_{k_1-1}} = \ldots = x_{i_{k_2+1}} - t_{i_{k_2+1}} = x_{i_{k_2}} - t_{i_{k_2}} = \Delta. \quad (6)$$

In other words, the restriction of the solutions to the nodes of the circuit are derived from each other by adding the same value to each component of the solution.

Let us consider two consecutive nodes of the circuit, say i_{k-1} and i_k. We have just proved that: $x_{i_{k-1}} - t_{i_{k-1}} = x_{i_k} - t_{i_k}$. This leads to: $t_{i_k} - t_{i_{k-1}} = x_{i_k} - x_{i_{k-1}}$.

If i_k is an *or*-node, then: $t_{i_k} - t_{i_{k-1}} = l(i_{k-1}, i_k)$ (see relation (2)) and thus $x_{i_k} - x_{i_{k-1}} = l(i_{k-1}, i_k)$. If i_k is an *and*-node, then: $x_{i_k} - x_{i_{k-1}} = l(i_{k-1}, i_k)$ (see relation (3)) and thus $t_{i_k} - t_{i_{k-1}} = l(i_{k-1}, i_k)$. In both cases, the arc (i_{k-1}, i_k) is saturated by boths solutions. Such a saturated circuit is a null-circuit.

Thus, this null-circuit is saturated by boths solutions. (7)

This contradicts the hypothesis of the Theorem and (i) is proved: if a feasible solution of P_{EQ} does not saturate any null-circuit, then this solution is unique.

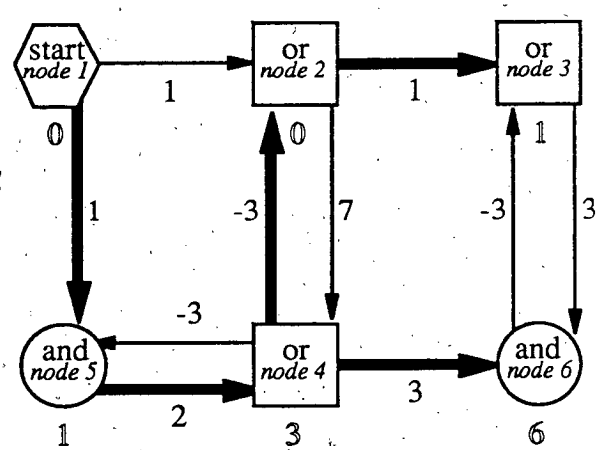
(ii) According to our premise, graph G does not contain any saturated null-circuit, and the feasible solution of P_{EQ} is unique. Thus, it is optimal to P_{EQ}. It is also optimal to P_{INEQ} as shown in [2].

(iii) According to Lemma 1, there exists a saturated path in graph G from node s to any other node. QED

As a consequence of Theorem 1, if the graph does not contain any null-circuit, a feasible solution of P_{EQ} is the unique feasible solution of P_{EQ} and is optimal for P_{EQ} and P_{INEQ}.

Definition 2

There exists a saturated path from the node s to each node. This set of saturated paths from node s is called a *saturated arborescence*. This concept generalizes the concept of the shortest path tree for the shortest path problem [3].



The solution (indicated by the numbers located under the nodes) is feasible to P_{EQ}. No null-circuit are saturated by this solution. According to Theorem 1, it is optimal.

Figure 3. Illustration of Theorem 1.

Corollary 1

If there exists two feasible solutions $\{t_i\}_{i\in V}$ and $\{x_i\}_{i\in V}$ of P_{EQ}, then:

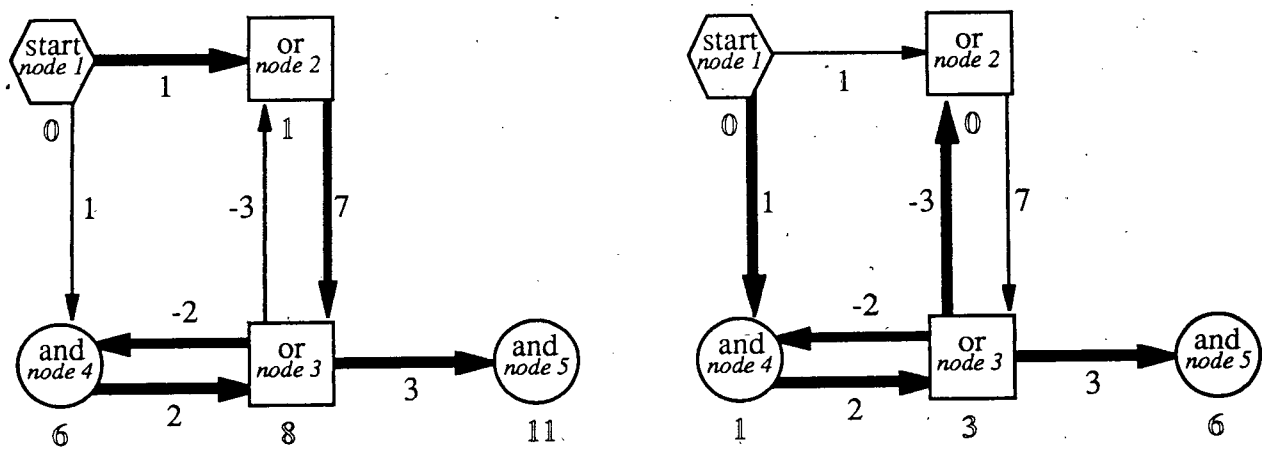
(i) the graph G contains at least one null-circuit $\mathcal{C}_0$ saturated by the two solutions, and

(ii) there exists $\Delta_{\mathcal{C}_0} > 0$ such that any node i of $\mathcal{C}_0$ satisfies the condition $|x_i - t_i| = \Delta_{\mathcal{C}_0}$.

Proof

Point (i) is mentioned in (7) while point (ii) is obtained by inequality (5) completed by (6).

QED



(a) The solution {0;1;8;6;11} given by the numbers located under the nodes is feasible for P_{EQ}.

(b) The solution {0;0;3;1;6} given by the numbers located under the nodes is feasible for P_{EQ}.

The two solutions (a) and (b) are different. According to Corollary 1, there exists a null-circuit saturated by the two solutions. This circuit is the one made with nodes 3 and 4.

Figure 4. Illustration of Corollary 1.

4. PROPERTIES OF THE FEASIBLE SOLUTION OF P_{EQ}

The goal of this section is to establish properties which lead to:

- a characterization of a feasible solution of P_{EQ},
- an algorithm to reach a feasible solution in the most general cases.

Theorem 2 *(Characterization of a feasible solution)*

Assume that one of the two following conditions is satisfied:

(i) There exists a set of negative circuits $C^-=\{\mathcal{C}_1^-,\mathcal{C}_2^-,\ldots,\mathcal{C}_q^-\}$ in the graph such that any arcs (j,i) which ends with an *and*-node i of one circuit of C^-, belongs to a circuit of C^-,

(ii) There exists a set of positive circuits $C^+=\{\mathcal{C}_1^+,\mathcal{C}_2^+,\ldots,\mathcal{C}_p^+\}$ in the graph such that any arcs (j,i) which ends with an *or*-node i of one circuit of C^+, belongs to a circuit of C^+.

In this case, the problem P_{EQ} has no feasible solution.

Proof

(i) Assume that condition (i) is satisfied and that there exists a feasible solution $\{t_i\}_{i\in V}$ to problem P_{EQ}. Thus, for any *and*-node i of one circuit of C^-, there exists $j\in Pred(i)$ such that the arc (j,i) is saturated by $\{t_i\}_{i\in V}$ and (j,i) belongs to a circuit of C^-. As a consequence, we can select backward a circuit $\mathcal{C}_r^-\in C^-$ such that, if $\mathcal{C}_r^-=\{i_0,i_1,\ldots,i_n=i_0\}$ then, for any $k\in\{1,2,\ldots,n\}$:

$t_{i_k}=t_{i_{k-1}}+l(i_{k-1},i_k)$ if i_k is an *and*-node (see condition (i)), and

$t_{i_k}\le t_{i_{k-1}}+l(i_{k-1},i_k)$ if i_k is an *or*-node (by definition of P_{EQ}).

By summing left-hand sides and right-hand sides of these equalities and inequalities for the

nodes of $\mathcal{C}_r^-$, we obtain:

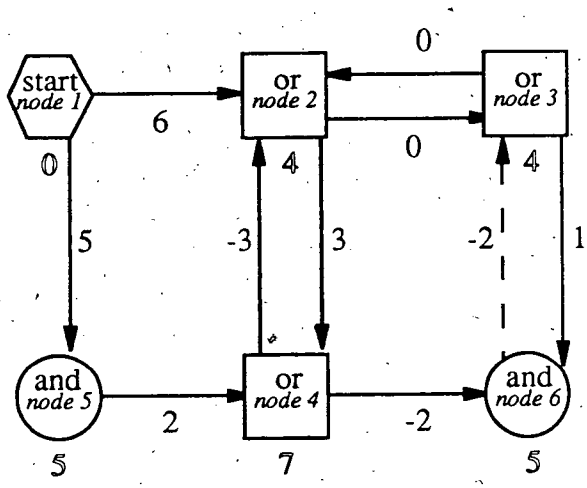
$t_{i_0} = t_{i_n} \leq t_{i_0} + \sum_{k=1}^{n} l(i_{k-1}, i_k)$ which is impossible since the length of $\mathcal{C}_r^-$ is strictly negative.

(ii) Using the same arguments as in (i), we can show that (ii) holds. QED

Remarks

Algorithm 1 which is presented at the end of this section, ends either by finding a feasible solution of P_{EQ}, or by finding a set of circuits for which at least one of the conditions of the Theorem 2 is satisfied. It follows that if the problem P_{EQ} has no feasible solution, at least one of the conditions of the Theorem 2 is satisfied.

These two conditions are the generalisation of the existence of negative circuit in routing problems (which included only *or*-nodes), and of the existence of positive circuit in the PERT/CPM models (which included only *and*-nodes).



Consider $\mathcal{C}_1^-$ the negative circuit made with nodes 2, 4, 6 and 3, and $\mathcal{C}_2^-$ the negative circuit made with nodes 3 and 6. Arcs (3, 6) and (4, 6) are the only arcs ending with an *and*-node belonging to $\mathcal{C}_1^-$ and $\mathcal{C}_2^-$. Both of these arcs belong to one of the circuits $\mathcal{C}_1^-$ and $\mathcal{C}_2^-$. According to Theorem 2, the problem P_{EQ} has no feasible solution.

The solution (indicated by the numbers located under the nodes) is not feasible to P_{EQ} since node 3 is a non-saturated node (because of the doted arc).

Figure 5. Illustration of Theorem 2.

Construction of an initial solution

We denote by $G=(V,E')$, the subgraph composed with the set of arcs $E' \subseteq E$, but which contains all the nodes of the initial graph $G=(V,E)$.

Let E_0 be a subset of E such that $G=(V,E_0)$ does not contain any directed cycle, and such that there exits a (directed) path from node s to any node in $G(V,E_0)$. Such a subset of E exists since we assumed that there exits a (directed) path from node s to any node in $G(V,E)$.

To obtain the optimal solution of problems P_{EQ} associated with $G(V,E_0)$, it is possible to use the breadth first search (BFS) which marks the nodes of the network progressively step by step (see [3]). Nodes are labelled using relations (S_{EQ}). Since the lengths of arcs are rational, all the

components of $\{t_i^0\}_{i\in V}$ are rational. Let q be the greatest common divisor of the lengths of arcs. Then, q is also a common divisor of the components of $\{t_i^0\}_{i\in V}$. The optimal solution $\{t_i^0\}_{i\in V}$ of problems P_{EQ} associated with $G(V,E_0)$ is not necessary feasible to P_{EQ} associated with $G(V,E)$. This solution $\{t_i^0\}_{i\in V}$ is feasible to the following problem P_{INEQ2}:

$$\text{Problem } P_{\mathrm{INEQ2}} : \text{minimize} \left[\sum_{i\in R}\left(t_i - \min_{j\in Pred(i)}\{t_j + l(j,i)\}\right) + \sum_{i\in A}\left(\max_{j\in Pred(i)}\{t_j + l(j,i)\} - t_i\right)\right]$$

subject to

$$\left\{\begin{array}{ll} t_i = 0 & \text{if } i = s \\ t_i \geq \min\limits_{j\in Pred(i)}\{t_j + l(j,i)\} & \text{if } i \in R \\ t_i \leq \max\limits_{j\in Pred(i)}\{t_j + l(j,i)\} & \text{if } i \in A \end{array}\right\} \qquad (S_{\mathrm{INEQ2}})$$

Algorithm 1, which will be presented at the end of this section, consists in reducing the value of the *or*-nodes i for which $t_i^0 > \min\limits_{j\in Pred(i)}\{t_j^0 + l(j,i)\}$ and increasing the value of the *and*-nodes i for which $t_i^0 < \max\limits_{j\in Pred(i)}\{t_j^0 + l(j,i)\}$. These values are be modified step by step and, at each step, the solution obtained still satisfies (S_{INEQ2}). Lemma 2 describes some specific properties of a subset of nodes $K\subset V$ and of a solution satisfying S_{INEQ2}. These properties allow us to improve the solution under consideration.

Lemma 2

We consider a rational solution $\{t_i\}_{i\in V}$ satisfying S_{INEQ2} and we assume that there exists $K\subset V$ (K is strictly included in V) such that:

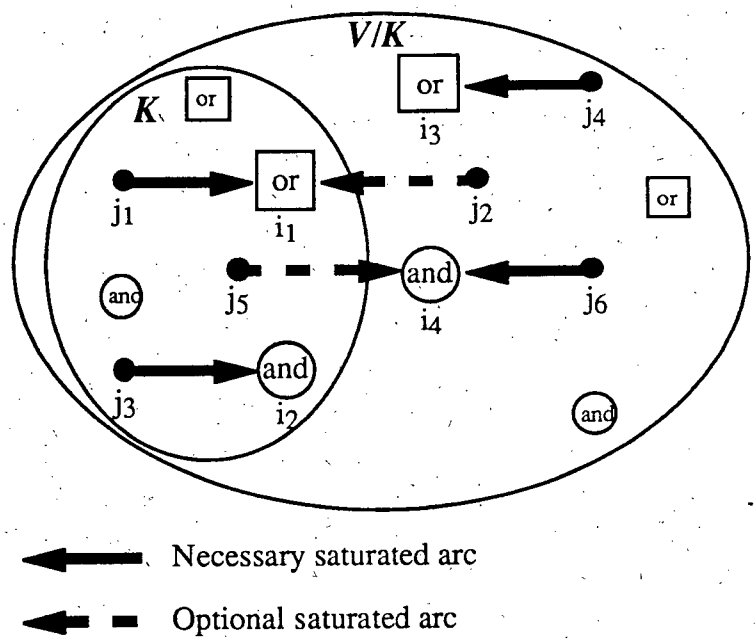
(i) for any saturated *or*-node $i\in K$, there exists a node $j\in K$ such that the arc (j,i) is saturated, i.e. $t_i = \min\limits_{j\in Pred(i)}\{t_j + l(j,i)\}$,

(ii) for any saturated *and*-node $i\in K$, it does not exist a node $j\in V/K$ such that the arc (j,i) is saturated, i.e. $t_i > \max\limits_{j\in Pred(i)}\{t_j + l(j,i)\}$,

(iii) for any saturated *or*-node $i\in V/K$, it does not exist a node $j\in K$ such that the arc (j,i) is saturated, i.e. $t_i < \min\limits_{j\in Pred(i)}\{t_j + l(j,i)\}$, and

(iv) for any saturated *and*-node $i\in V/K$, there exists a node $j\in V/K$ such that the arc (j,i) is saturated, i.e. $t_i = \max\limits_{j\in Pred(i)}\{t_j + l(j,i)\}$.

If these conditions are satisfied, then there exist an infinite number of rational solutions satisfying S_{INEQ2} which saturate all the nodes saturated by $\{t_i\}_{i\in V}$.



All non-saturated arcs are optional for the definition of K and are not represented in the figure. Properties (i) to (iv) can be summarized as follows:

- any saturated node is the end point of a saturated arc whose origin is in the same subset,
- the only saturated arcs which cross the boundaries of K are those which either ends in K with an *or*-node, or ends in V/K with an *and*-node (these saturated arcs are optional).

Figure 6. Definition of the subset K.

Proof

Let $\{t_i\}_{i\in V}$ the solution under consideration which satisfies S_{INEQ2}. The basic idea behind the proof presented hereafter is to use the relative independence between the restriction of the solution to the nodes belonging to K, and the restriction of the solution to the nodes belonging to V/K. We denote by:

A_S (respectively R_S) the set of the saturated *and*-nodes (respectively *or*-nodes) and A_{NS} (respectively R_{NS}) the set of the non-saturated *and*-nodes (respectively *or*-nodes).

We consider:

$$\Delta_{A_S\cap K} = \begin{cases} +\infty & \text{if } A_S \cap K = \varnothing \text{ or } Pred(i \text{ s.t. } i \in A_S \cap K)/K = \varnothing \\ \min\limits_{i\in A_S\cap K}\left\{\max\limits_{j\in Pred(i)/K}\{t_i - t_j - l(j,i)\}\right\} & \text{otherwise.} \end{cases}$$

According to property (ii), we have: $t_j + l(j,i) < t_i \quad \forall i \in A_S \cap K,\ \forall j \in Pred(i)/K$.

Thus, $\Delta_{A_S\cap K} > 0$.

We also consider:

$$\Delta_{R_S/K} = \begin{cases} +\infty & \text{if } R_S/K = \varnothing \text{ or } Pred(i \text{ s.t. } i \in R_S/K) \cap K = \varnothing \\ \min\limits_{i\in R_S/K}\left\{\min\limits_{j\in Pred(i)\cap K}\{t_j + l(j,i) - t_i\}\right\} & \text{otherwise.} \end{cases}$$

According to property (iii), we have: $t_j + l(j,i) > t_i \quad \forall i \in R_S/K,\ \forall j \in Pred(i) \cap K$.

Thus, $\Delta_{R_S/K} > 0$.

We then consider:

$$\Delta_{R_{NS}\cap K} = \begin{cases} +\infty & \text{if } \exists j \in K \text{ s.t. } t_i > t_j + l(j,i) \\ \min\limits_{i\in R_{NS}\cap K}\left\{\min\limits_{j\in Pred(i)}\{t_i - t_j - l(j,i)\}\right\} & \text{otherwise.} \end{cases}$$

By definition of the non-saturated *or*-nodes, we have: $\Delta_{R_{NS}\cap K} > 0$.

Finally, we consider:

$$\Delta_{A_{NS}/K} = \begin{cases} +\infty & \text{if } \exists j \in V/K \text{ s.t. } t_i < t_j + l(j,i) \\ \min\limits_{i\in A_{NS}/K}\left\{\max\limits_{j\in Pred(i)}\{t_j + l(j,i) - t_i\}\right\} & \text{otherwise.} \end{cases}$$

By definition of the non-saturated *and*-nodes, we have: $\Delta_{A_{NS}/K} > 0$.

We set $\Delta = \min\left\{\Delta_{A_S\cap K}; \Delta_{R_S/K}; \Delta_{R_{NS}\cap K}; \Delta_{A_{NS}/K}\right\} > 0$. Δ is rational since the components of $\{t_i\}_{i\in V}$ and the lengths of arcs are rational.

If $s\in V/K$, we claim that, for any rational $\alpha\in]0;\Delta]$ and $\alpha\neq+\infty$, the solution $\{t_i(\alpha)\}_{i\in V}$, defined as follows, satisfies S_{INEQ2} and saturates all the nodes saturated by $\{t_i\}_{i\in V}$:

$$\begin{cases} t_i(\alpha) = t_i - \alpha & \text{if } i \in K \\ t_i(\alpha) = t_i & \text{if } i \in V/K. \end{cases}$$

The explanation is straightforward.

Similarly, if $s\in K$, we claim that, for any rational $\beta\in]0;\Delta]$ and $\beta\neq+\infty$, the solution $\{t_i(\beta)\}_{i\in V}$, defined as follows, satisfies S_{INEQ2} and saturates all the nodes saturated by $\{t_i\}_{i\in V}$:

$$\begin{cases} t_i(\beta) = t_i & \text{if } i \in K \\ t_i(\beta) = t_i + \beta & \text{if } i \in V/K. \end{cases}$$

QED

Remark

As a consequence of the Lemma 2, if there exists $\{t_i\}_{i\in V}$, feasible solution of P_{EQ}, and a set of nodes $K\subset V$ such that properties (i) to (iv) hold, then P_{EQ} has an infinite number of feasible solutions. Moreover, according to the definitions of Δ and $\{t_i(\alpha)\}_{i\in V}$ given in the proof, if $s\notin K$, then $\{t_i\}_{i\in V}$ is not optimal to P_{EQ} since the solution $\{t_i(\alpha)\}_{i\in V}$ is better. Furthermore, if $\Delta = +\infty$ then there is no optimal solution of P_{EQ} since it is possible to decrease $\{t_i\}_{i\in K}$ to $-\infty$.

If q, the greatest common divisor of the lengths of arcs, is a common divisor of the components of $\{t_i\}_{i\in V}$, then it is also a common divisor of Δ and of the components of the solutions $\{t_i(\alpha{=}\Delta)\}_{i\in V}$ and $\{t_i(\beta{=}\Delta)\}_{i\in V}$.

Assume that $\{t_i\}_{i\in V}$ is a solution satisfying S_{INEQ2}. If we want to reduce (resp. increase) the

values $\{t_i\}_{i\in L}$ of the nodes of a set $L\subset V$ in order to obtain another solution satisfying S_{INEQ2} which saturate all the nodes already saturated by $\{t_i\}_{i\in V}$, we will show that we must reduce (resp. increase) the values of all the nodes of a larger set, called L_1 (resp. L_2). These new sets are introduced hereafter.

Definitions 3

For any set of nodes L, we define L_1 as the smallest set such that:

(a1) If $i\in L$, then $i\in L_1$, i.e. $L_1\supseteq L$.

(b1) If $i\in R$ and if the starting node j of *at least one* saturated arc (j,i) belongs to L_1, then $i\in L_1$.

(c1) If $i\in A$ and if the starting node j of *all the* saturated arcs (j,i) belong to L_1, then $i\in L_1$.

For a set of nodes L, we define L_2 as the smallest set such that:

(a2) If $i\in V/L$, then $i\in L_2$, i.e. $L_2\supseteq V/L$.

(b2) If $i\in R$ and if the starting node j of *all the* saturated arcs (j,i) belong to L_2, then $i\in L_2$.

(c2) If $i\in A$ and if the starting node j of *at least one* saturated arc (j,i) belongs to L_2, then $i\in L_2$.

We can obtain L_1 (resp. L_2) from L (resp. V/L), using a breadth first search (BFS) on the subgraph composed with the saturated arcs of E, knowing the feasible solution at hand. Starting from L (resp. V/L), the nodes satisfying conditions (b1) and (c1) (resp. (b2) and (c2)) are added one after the other until no more node can be added. The resulting subset is L_1 (resp. L_2). Lemma 3 precises the properties that a set L must satisfy in order to be able to modify the values of the nodes of L_1 (or L_2) derived from L.

Lemma 3

Assume that there exists a solution $\{t_i\}_{i\in V}$ satisfying S_{INEQ2}.

(a) If L is a set of nodes satisfying properties (i) and (ii) of Lemma 2 by replacing K by L, then L_1 (where L_1 is defined by (a1), (b1), (c1)) satisfies properties (i) to (iv) of Lemma 2 by replacing K by L_1.

(b) If V/L is a set of nodes satisfying properties (iii) and (iv) of Lemma 2 by replacing V/K by V/L, then V/L_2 (where L_2 is defined by (a2), (b2), (c2)) satisfies properties (i) to (iv) of Lemma 2 by replacing K by V/L_2.

As a consequence, if $V/L_1\neq\emptyset$ or $L_2\neq\emptyset$, then there exist an infinite number of solutions satisfying S_{INEQ2} which saturate all the nodes saturated by $\{t_i\}_{i\in V}$ (see Lemma 2).

Proof

(a) Let L be a set satisfying properties (i) to (ii) of Lemma 2 and L_1 the subset derived from L.

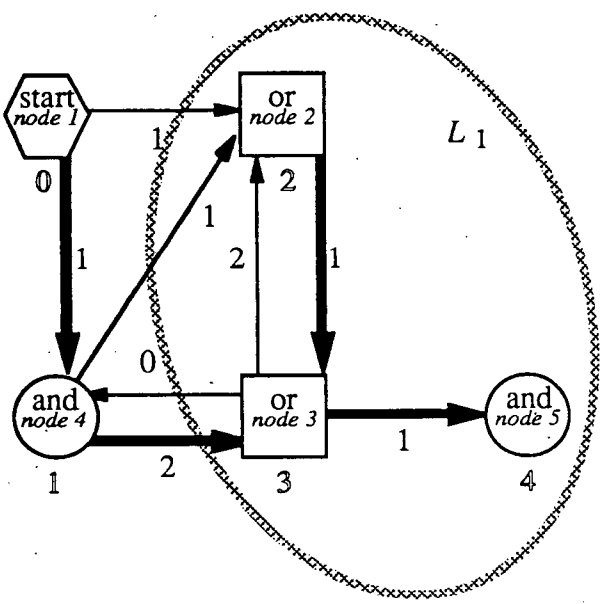
If an *or*-node $i \notin L_1$ is such that the starting node j of a saturated arc (j,i) belongs to L_1, node i would be added to L_1 according to (b1) of Definitions 3. Thus, the nodes of L_1 satisfy property (iii) of Lemma 2.

Moreover, if an *and*-node $i \notin L_1$ is such that the starting node j of any saturated arc (j,i) belongs to L_1, node i would be added to L_1 according to (c1) of Definitions 3. Thus, the nodes of L_1 satisfy property (iv) of Lemma 2. By construction (see (b1) and (c1)), the nodes of L_1/L satisfy properties (i) and (ii) of Lemma 2. Finally, the nodes of $L_1=L \cup L_1/L$ satisfy properties (i) to (iv) of Lemma 2.

(b) Using the same arguments as in (a), we can show that (b) holds. QED

Remark

If $L_1=V$ (or $L_2=V$), we cannot apply Lemma 2 since L_1 (resp. L_2) should be strictly included in V for Lemma 2 to apply.



The solution (indicated by the numbers located under the nodes) satisfies (S_{INEQ2}). Node 2 is non-saturated because of arc (1, 2).

The set of nodes $L=\{2\}$ satisfies the two first properties of Lemma 2. The set of nodes $L_1=K=\{2, 3, 5\}$ derived from L, satisfies the four properties of Lemma 2. Thus, for any $\alpha \in [-1; 0]$, the solutions $\{t_i(\alpha)\}_{i=1,\dots,5}$ defined by $\{t_1=0; t_2=2+\alpha; t_3=3+\alpha; t_4=1; t_5=4+\alpha\}$ satisfies (S_{INEQ2}) and saturates all the nodes saturated by $\{t_i\}_{i=1,\dots,5}$.

Figure 7. Illustration of Lemma 2 and 3.

Definitions

We consider a solution $\{t_i\}_{i \in V}$ satisfying S_{INEQ2}. Let R_{NS} (respectively R_S) be the set of non-saturated (respectively saturated) *or*-nodes. Let $R_{NS,1}$ be the set derived from R_{NS} by applying (a1) to (c1) of Definitions 3.

Since R_{NS} satisfies properties (i) and (ii) of Lemma 2, $R_{NS,1}$ satisfies properties (i) to (iv) of

Lemma 2 (see Lemma 3). Furthermore, $s \notin R_{NS}$. As a consequence of the definition of $R_{NS,1}$ and since node s has no predecessor, $s \notin R_{NS,1}$. Thus, we can apply Lemma 2 to decrease the values of the nodes belonging to $R_{NS,1}$ using the solution $\{t_i(\alpha=\Delta)\}_{i \in V}$ (see proof of Lemma 2).

Let A_{NS} (respectively A_S) be the set of non-saturated (respectively saturated) *and*-nodes non-saturated. Let $A_{NS,2}$ be the set derived from A_{NS} by applying (a2) to (c2) of Definitions 3.

Since A_{NS} satisfies properties (i) and (ii) of Lemma 2, $A_{NS,2}$ satisfies properties (i) to (iv) of Lemma 2 (see Lemma 3). Furthermore, $s \notin A_{NS,2}$. As a consequence of the definition of $A_{NS,2}$ and since node s has no predecessor, $s \notin A_{NS,2}$. Thus, we can apply Lemma 2 to increase the values of the nodes belonging to $A_{NS,2}$ using the solution $\{t_i(\beta=\Delta)\}_{i \in V}$ (see proof of Lemma 2).

Lemma 4 shows that if applying Lemma 2 to $R_{NS,1}$ (respectively $A_{NS,2}$) we obtain $\Delta = +\infty$, then the problem P_{EQ} under consideration has no feasible solution.

Lemma 4

Assume that there exists a solution $\{t_i\}_{i \in V}$ satisfying S_{INEQ2}.

(a) If there exists a set $R_{NS,1}$ derived from R_{NS} satisfying the following properties:

- no arc (j,i) starts from $j \notin R_{NS,1}$ and ends with *and*-node $i \in R_{NS,1}$,
- for any $i \in R_{NS}$, there exists (j,i) starting from $j \in R_{NS,1}$ such that $t_i > t_j + l(j,i)$,

then there exists in $R_{NS,1}$ a set of negative circuits $C^- = \{\mathcal{C}_1^-, \mathcal{C}_2^-, \ldots, \mathcal{C}_q^-\}$ such that any arcs (j,i) which ends with an *and*-node i of one circuit of C^-, belongs to a circuit of C^-.

(b) If there exists a set $A_{NS,2}$ derived from A_{NS} satisfying the following properties:

- no arc (j,i) starts from $j \in A_{NS,2}$ and ends with *and*-node $i \notin A_{NS,2}$,
- for any $i \in A_{NS}$, there exists (j,i) starting from $j \in A_{NS,2}$ such that $t_i < t_j + l(j,i)$,

then there exists a set of positive circuits $C^+ = \{\mathcal{C}_1^+, \mathcal{C}_2^+, \ldots, \mathcal{C}_p^+\}$ in the graph such that any arcs (j,i) which ends with an *or*-node i of one circuit of C^+, belongs to a circuit of C^+.

In both cases, according to Theorem 2, the problem P_{EQ} has no feasible solution.

Proof

(a) Assume that there exists $\{t_i\}_{i \in V}$ satisfying S_{INEQ2} and a set $R_{NS,1}$ derived from R_{NS} satisfying the two properties (a) of Lemma 4.

Let $R_{NS}^1, R_{NS}^2, \ldots, R_{NS}^n$ be the smallest disjoint subsets of R_{NS} such that $R_{NS} = R_{NS}^1 \cup R_{NS}^2 \cup \ldots \cup R_{NS}^n$ and that, for any $1 \le r \le n$ and any $i \in R_{NS}^r$, there exists (j,i) starting from $j \in R_{NS,1}^r$ which satisfies $t_i > t_j + l(j,i)$, where $R_{NS,1}^r$ is derived from R_{NS}^r. According to the second condition of (a), we know that this partition is possible.

Consider a subset R_{NS}^r. For any $i \in R_{NS}^r$, there exists (j,i) starting from $j \in R_{NS,1}^r$ such that $t_i > t_j + l(j,i)$. By construction of $R_{NS,1}^r$, there exists in $R_{NS,1}^r$ a saturated path from a node of R_{NS}^r to

$j \in R^r_{NS,1}$. Thus, there exists, in $R^r_{NS,1}$, a circuit which includes arcs saturated by $\{t_i\}_{i\in V}$, and at least one arc (j,i) satisfying $t_i > t_j + l(j,i)$. Let $\mathcal{C}^r$ be such a circuit. For any arc (j,i) of this circuit $\mathcal{C}^r$, we have $t_j - t_i \leq l(j,i)$ and there exists at least one arc for which $t_j - t_i < l(j,i)$. It follows that $R^r_{NS,1}$ contains a negative circuit $\mathcal{C}^r$ which includes an arc (j,i) satisfying $t_i > t_j + l(j,i)$. Moreover, there exists a path $\mathbf{P}_{\to k}$ from $\mathcal{C}^r$ to any *or*-node $k \in R^r_{NS}$ which includes either arcs (j,i) satisfying $t_i > t_j + l(j,i)$, or saturated arcs.

Two cases are possible:

First case. There exists $R^r_{NS,1}$ such that the predecessors of any *and*-node i of the negative circuit $\mathcal{C}^r$, belong to $R^r_{NS,1}$. Let C^- be the set of the negative circuits in $R^r_{NS,1}$ intersecting $\mathcal{C}^r$. In this case, for any predecessor j of an *and*-node i of a circuit $\mathcal{C}^-$ in C^-, there exists a saturated path $\mathbf{P}_{k\to j}$ from $k \in R^r_{NS}$ to j. Thus, arc (j,i) belongs to a negative circuit in C^- made with paths $\mathbf{P}_{\to k}$ and $\mathbf{P}_{k\to j}$, arc (j,i), and a path included in $\mathcal{C}^-$.

Second case. For any subset $R^r_{NS,1}$, there exists a predecessor $j_q \notin R^r_{NS,1}$ of an *and*-node i_r $(r \neq q)$ of the negative circuit $\mathcal{C}^r$. In this case, there exists a negative circuit made with arcs similar to (j_q, i_r) and paths included in $\mathcal{C}^r$ and joining i_r to j_r. Finally, it is possible to obtain in $R_{NS,1}$ a set of negative circuits C^- such that any arcs (j,i) which ends with an *and*-node i of one circuit of C^-, belongs to a circuit of C^-.

(b) Using the same arguments as in (a), we can claim that (b) holds. QED

Construction of a feasible solution of P_{EQ}

First step. We start building the solution $\{t_i\}_{i\in V}$ optimal to the problem P_{EQ} associated with $G(V,E_0)$, where E_0 is a subset of E such that $G=(V,E_0)$ does not contain any directed cycle, and such that there exits a (directed) path from node s to any node in $G(V,E_0)$. The solution $\{t_i\}_{i\in V}$ satisfies (S_{INEQ2}). Then, the algorithm proceeds in two other steps.

Second step. The algorithm modifies the solution $\{t_i\}_{i\in V}$ in order to saturate the non-saturated *or*-nodes. The idea of the algorithm is to decrease the values $\{t_i\}$ of the nodes i which belong to $R_{NS,1}$ derived from R_{NS}, and to adjust $R_{NS,1}$ by adding or removing nodes adequately.

If $R_{NS}=\varnothing$ then all the *or*-nodes are saturated by the solution. Otherwise, since R_{NS} satisfies the properties (i) and (ii) of Lemma 2, $R_{NS,1}$ satisfies the properties (i) to (iv) of Lemma 2 (see Lemma 3). Furthermore, $s \notin R_{NS}$. As a consequence of the definition of $R_{NS,1}$ and since node s has no predecessor, $s \notin R_{NS,1}$. As in the proof of Lemma 2, we define Δ and, for $\alpha \in]0;\Delta]$, $\{t_i(\alpha)\}_{i\in V}$. According Lemma 4, if $\Delta = +\infty$ then there is no feasible solution to P_{EQ}.

If $\Delta \neq +\infty$, by definition of Δ, the solution $\{t_i(\alpha=\Delta)\}_{i\in V}$ saturates one arc (j,i) which was not saturated by solution $\{t_i\}_{i\in V}$. Four cases are possible to adjust $R_{NS,1}$, according to its definition:

- this arc (j,i) starts from $j \notin R_{NS,1}$ and ends with *and*-node $i \in R_{NS,1}$: this *and*-node i must be removed from $R_{NS,1}$,
- this arc (j,i) starts from $j \in R_{NS,1}$ and ends with *or*-node $i \notin R_{NS}$: this *or*-node i must be added

to $R_{NS,1}$,

- this arc (j,i) starts from $j \notin R_{NS,1}$ and ends with $i \in R_{NS}$: this *or*-node i must be removed from R_{NS} and $R_{NS,1}$, or
- this arc (j,i) starts from $j \in R_{NS,1}$ and ends with $i \notin A_{NS}$: this *and*-node i must be added to $R_{NS,1}$ (and removed from A_{NS}).

In both cases, we derive a new set $R_{NS,1}$ from R_{NS} by applying (a1) to (c1) of Definitions 3 taking into account the arcs saturated by solution $\{t_i(\alpha=\Delta)\}_{i\in V}$.

For each set $R_{NS,1}$, we define a solution $\{t_i\}_{i\in V}$. Since the number of non-saturated *or*-nodes decreases and since the value $\sum_{i\in R_{NS}} \min_{j\in Pred(i)} \{t_i - t_j - l(j,i)\}$ strictly decreases at each modification of the solution, the algorithm converges to a feasible solution of P_{EQ}, if any.

Since all the non-saturated *or*-nodes belong to $R_{NS} \subseteq R_{NS,1}$ and since all the *and*-nodes of $R_{NS,1}$ are saturated, we have: $A_S \cap R_{NS,1} = A \cap R_{NS,1}$, $R_S/R_{NS,1} = R/R_{NS,1}$, $R_{NS} \cap R_{NS,1} = R_{NS}$ and $A_{NS}/R_{NS,1} = A_{NS}$. These simplifications will be used in the notations of steps 2.4 to 2.8 in Algorithm 1.

Third step. The algorithm modifies the solution in order to saturate all the *and*-nodes. The idea is to increase the values $\{t_i\}$ of the nodes i which belong to $A_{NS,2}$ (where $A_{NS,2}$ is derived from A_{NS} by applying (a2) to (c2), to the set of the non-saturated *and*-nodes) . Then, $A_{NS,2}$ is adjusted by adding or removing nodes adequately, similarly as in the second step.

Algorithm 1 - Construction of a feasible solution of P_{EQ}

Input data:

$G=(V,E)$ is the graph under consideration.

Output data:

$\{t_i\}_{i\in V}$, is a feasible solution to P_{EQ}, if any.

1. Construction of an initial solution satisfying (S_{INEQ2})

1.1. Select E_0, a subset of E, such that $G=(V,E_0)$ does not contain any directed cycle, and such that there exits a (directed) path from node s to any node in $G(V,E_0)$.

1.2. Compute $\{t_i\}_{i\in V}$, the optimal solution of problems P_{EQ} associated with $G(V,E_0)$, using a breadth first search.

2. Saturation of the *or*-nodes

2.1. Let R_{NS} be the set of the *or*-nodes which are not saturated by $\{t_i\}_{i\in V}$.

2.2. If $R_{NS}=\varnothing$ then goto step 3.

2.3. Derive $R_{NS,1}$ from R_{NS} by applying:

(a1) If $i \in R_{NS}$, then $i \in R_{NS,1}$.

(b1) If $i \in R$ and if the starting node j of *at least one* saturated arc (j,i) belongs to $R_{NS,1}$, then $i \in R_{NS,1}$.

(c1) If $i \in A$ and if the starting node j of *all the* saturated arcs (j,i) belong to $R_{NS,1}$, then $i \in R_{NS,1}$.

2.4. Set $\Delta_{A \cap R_{NS,1}} = \begin{cases} +\infty & \text{if } A \cap R_{NS,1} = \varnothing \text{ or } Pred(i \text{ s.t. } i \in A \cap R_{NS,1}) / R_{NS,1} = \varnothing \\ \min\limits_{i \in A \cap R_{NS,1}} \left\{ \max\limits_{j \in Pred(i)/R_{NS,1}} \{t_i - t_j - l(j,i)\} \right\} & \text{otherwise.} \end{cases}$

2.5. Set $\Delta_{R/R_{NS,1}} = \begin{cases} +\infty & \text{if } R/R_{NS,1} = \varnothing \text{ or } Pred(i \text{ s.t. } i \in R/R_{NS,1}) \cap R_{NS,1} = \varnothing \\ \min\limits_{i \in R/R_{NS,1}} \left\{ \min\limits_{j \in Pred(i) \cap R_{NS,1}} \{t_j + l(j,i) - t_i\} \right\} & \text{otherwise.} \end{cases}$

2.6. Set $\Delta_{R_{NS}} = \begin{cases} +\infty & \text{if } \exists j \in R_{NS,1} \text{ s.t. } t_i > t_j + l(j,i) \\ \min\limits_{i \in R_{NS}} \left\{ \min\limits_{j \in Pred(i)} \{t_i - t_j - l(j,i)\} \right\} & \text{otherwise.} \end{cases}$

2.7. Set $\Delta_{A_{NS}} = \begin{cases} +\infty & \text{if } \exists j \in V/R_{NS,1} \text{ s.t. } t_i < t_j + l(j,i) \\ \min\limits_{i \in A_{NS}} \left\{ \max\limits_{j \in Pred(i)} \{t_j + l(j,i) - t_i\} \right\} & \text{otherwise.} \end{cases}$

2.8. Set $\Delta = \min\left\{\Delta_{A \cap R_{NS,1}}; \Delta_{R/R_{NS,1}}; \Delta_{R_{NS}}; \Delta_{A_{NS}}\right\} > 0$.

2.9. If $\Delta = +\infty$ then there is no feasible solution of P_{EQ}. End.

2.10. Set $t_i = t_i - \Delta$ for $i \in R_{NS,1}$.

2.11. Goto step 2.

3. Saturation of the *and*-nodes

3.1. Let A_{NS} be the set of the *and*-nodes which are not saturated by $\{t_i\}_{i \in V}$.

3.2. If $A_{NS}=\varnothing$ then $\{t_i\}_{i \in V}$ is feasible to P_{EQ}. End.

3.3. Derive $A_{NS,2}$ from A_{NS} by applying:

(a2) If $i \in A_{NS}$, then $i \in A_{NS,2}$.

(b2) If $i \in R$ and if the starting node j of *all the* saturated arcs (j,i) belong to $A_{NS,2}$, then $i \in A_{NS,2}$.

(c2) If $i \in A$ and if the starting node j of *at least one* saturated arc (j,i) belongs to $A_{NS,2}$, then $i \in A_{NS,2}$.

3.4. Set $\Delta_{A/A_{NS,2}} = \begin{cases} +\infty & \text{if } A/A_{NS,2} = \varnothing \text{ or } Pred(i \text{ s.t. } i \in A/A_{NS,2}) \cap A_{NS,2} = \varnothing \\ \min\limits_{i \in A/A_{NS,2}} \left\{ \max\limits_{j \in Pred(i) \cap A_{NS,2}} \{t_i - t_j - l(j,i)\} \right\} & \text{otherwise.} \end{cases}$

3.5. Set $\Delta_{R \cap A_{NS,2}} = \begin{cases} +\infty & \text{if } R \cap A_{NS,2} = \varnothing \text{ or } Pred(i \text{ s.t. } i \in R \cap A_{NS,2}) / A_{NS,2} = \varnothing \\ \min\limits_{i \in R \cap A_{NS,2}} \left\{ \min\limits_{j \in Pred(i)/A_{NS,2}} \{t_j + l(j,i) - t_i\} \right\} & \text{otherwise.} \end{cases}$

3.6. Set $\Delta_{R_{NS}} = \begin{cases} +\infty & \text{if } \exists j \notin A_{NS,2} \text{ s.t. } t_i > t_j + l(j,i) \\ \min\limits_{i \in R_{NS}} \left\{ \min\limits_{j \in Pred(i)} \{t_i - t_j - l(j,i)\} \right\} & \text{otherwise.} \end{cases}$

3.7. Set $\Delta_{A_{NS}} = \begin{cases} +\infty & \text{if } \exists j \in A_{NS,2} \text{ s.t. } t_i < t_j + l(j,i) \\ \min\limits_{i \in A_{NS}} \left\{ \max\limits_{j \in Pred(i)} \{t_j + l(j,i) - t_i\} \right\} & \text{otherwise.} \end{cases}$

3.8. Set $\Delta = \min\left\{\Delta_{A/A_{NS,2}}; \Delta_{R \cap A_{NS,2}}; \Delta_{R_{NS}}; \Delta_{A_{NS}}\right\} > 0$.

3.9. If $\Delta = +\infty$ then there is no feasible solution of P_{EQ}. End.

3.10. Set $t_i = t_i + \Delta$ for $i \in A_{NS,2}$.

3.11. Goto step 3.

Proposition

The complexity of the previous algorithm is $O[(|V|+|E|)|V|.Lq]$, where $|V|$ and $|E|$ represent respectively the number of nodes and arcs, q is the greatest common divisor of the lengths of arcs and $L=\sum_{i_0\in V}\sum_{j_0\in Pred(i_0)}|l(j_0,i_0)|$ is equal to the sum of the lengths of the arcs.

Proof

Using a breadth first search (BFS), it is possible to find a subset E_0 of E such that it does not contain any directed cycle, and such that there exits a (directed) path from node s to any node. Thus, step 1.1 is computed in $O(|V|+|E|)$.

The breadth first search (BFS) marks the nodes of the network step by step (see [3]). Nodes are labelled using the relations (S_{EQ}). Thus, step 1.2 is computed in $O(|V|+|E_0|)\leq O(|V|+|E|)$.

It is possible to find the non-saturated nodes by looking at all the arcs. Thus, steps 2.1 and 3.1 are computed in $O(|V|+|E|)$.

Steps 2.2 and 3.2 are computed in O(1).

Applying (a1) to (c1) to the subgraph limited to the saturated arcs, we derive $R_{NS,1}$ from R_{NS}. Applying (a2) to (c2) to the subgraph limited to the saturated arcs, we derive $A_{NS,2}$ from A_{NS}. Thus, steps 2.3 and 3.3 are computed in $O(|V|+|E|)$.

Steps 2.4 to 2.7, and 3.4 to 3.7 are computed in $O(|V|+|E|)$, since it is sufficient to look at the arcs one end of which belongs to $R_{NS,1}$ or $A_{NS,2}$.

Steps 2.8 to 2.9, and 3.8 to 3.9 are computed in O(1).

Steps 2.10 and 3.10 are computed in $O(|R_{NS,1}|)+O(|A_{NS,2}|)\leq O(|V|)$.

Let q be the greatest common divisor of the lengths of arcs. q is a common divisor of the components of the initial solution $\{t_i\}_{i\in V}$, computed at step 1.2, and is also a common divisor of the components of all the new solutions $\{t_i\}_{i\in V}$ computed at steps 2.10 and 3.10 (see Lemma 2). Since the criterium of P_{INEQ2}, which is $\sum_{i\in R}\left(t_i-\min_{j\in Pred(i)}\{t_j+l(j,i)\}\right)+\sum_{i\in A}\left(\max_{j\in Pred(i)}\{t_j+l(j,i)\}-t_i\right)$, strictly decreases, it decreases by a value greater or equal to $1/q$, at each step. If there exists a feasible solution to P_{EQ}, the criterium is lower bounded by 0. Thus, steps 2.11 and 3.11 are performed at most $O(Zq)$-times, where Z is the value of the criterium for the initial solution computed at step 1.2. Since each component t_i of the initial solution is equal to the length of a path from s to node i, we have for any $(j,i)\in E$, $|t_i-t_j-l(j,i)|\leq 2\sum_{i_0\in V}\sum_{j_0\in Pred(i_0)}|l(j_0,i_0)|$ and $Z\leq 2|V|\sum_{i_0\in V}\sum_{j_0\in Pred(i_0)}|l(j_0,i_0)|$. It follows that steps 2.11 and 3.11 are performed at most $O(|V|Lq)$-times, where L is equal to the sum of the lengths of the arcs of G.

Finally, the algorithm requires $O[(|V|+|E|)|V|.Lq]$ operations. QED

Remark

According to Theorem 1, if there is no null-circuit in the graph, the feasible solution to P_{EQ} provided by Algorithm 1 is optimal.

5. PROPERTIES OF THE OPTIMAL SOLUTION

The goal of this section is to establish properties which lead to:

- a characterization of the optimal solution,
- an algorithm to reach the optimal solution in the most general cases.

Definition 4

A null-circuit $\mathscr{C}$ is said to be ***and*-saturated** if $\mathscr{C}$ is saturated and if there exists a saturated path ending with an *and*-node of $\mathscr{C}$ which do not contain arcs belonging to a saturated null-circuit and which starts from either node s, or a node of another *and*-saturated null-circuit.

The set of *and*-saturated null-circuits can be obtained iteratively from the above definition.

Note that there exists a saturated path from node s to any node of an *and*-saturated null-circuit.

Theorem 3 *(Necessary and sufficient conditions for optimality)*

A feasible solution of P_{EQ} is optimal if and only if the following conditions are satisfied:

(i) there is a saturated path from node s to any node of G, and

(ii) any saturated null-circuit is *and*-saturated.

Proof

(a) *A feasible solution of P_{EQ} is not optimal if one of the two conditions* (i) *and* (ii) *is not satisfied.*

Assume that there exists a feasible solution of P_{EQ} for which condition (i) and/or (ii) is not satisfied. Let N be the set nodes such that it does not exist a saturated path either from node s or from a node belonging to an *and*-saturated null-circuit, to such a node. By definition of N, the only saturated arcs (j,i) such that $j \in V/N$ and $i \in N$, end with an *or*-node i which belongs to a null-circuit. According to the definition of P_{EQ} (see Section 2), whatever $i \in N$, there exists $j \in N$ such that (j,i) is saturated. As a consequence, N satisfies the properties (i) and (ii) of Lemma 2. Thus, if N_l is derived from N by applying (a1) to (c1) of Definitions 3, N_l satisfies the properties (i) to (iv) of Lemma 2 (see Lemma 3). Furthermore, $s \notin N$. As a consequence of the definition of N_l and since node s has no predecessor, $s \notin N_l$. According to Lemma 2, we conclude that a feasible solution of P_{EQ} for which condition (i) and/or (ii) of Theorem 1 is not satisfied is not optimal.

(b) *If a feasible solution of P_{EQ} satisfies the two conditions* (i) *and* (ii), *it is optimal.*

Assume that there exists a feasible solution $\{t_i\}_{i\in V}$ of P_{EQ} for which the two conditions (i) and (ii) are satisfied. According to condition (ii), we can number the p null-circuits saturated by this solution from 1 to p such that:

for any saturated null-circuit $\mathcal{C}_n$ ($1\leq n\leq p$), there exists a saturated path $\mathcal{P}_n$ ending with an *and*-node of $\mathcal{C}_n$ which does not contain arcs from another saturated null-circuit and which starts from either node s, or from a node of *and*-saturated null-circuit belonging to $\mathcal{C}_m$, where $1\leq m<n$.

We set $O_0=\{s\}$. The solution $\{t_s=0\}=\{t_i\}_{i\in O_0}$ is equal to the optimal solution to problem P_{EQ} restricted to the set O_0. Let also O_n be $O_n=O_{n-1}\cup\mathcal{P}_n\cup\mathcal{C}_n$, for $1\leq n\leq p$.

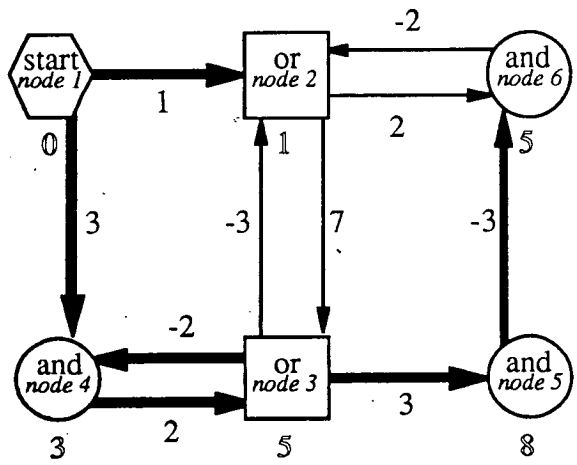
Assume that the solution $\{t_i\}_{i\in O_{n-1}}$ is equal to to the optimal solution to problem P_{EQ} restricted to O_{n-1}. Another solution to the restriction of problem P_{EQ} to O_n, better than $\{t_i\}_{i\in O_n}$, will saturate a null-circuit $\mathcal{C}$, according to Corollary 1. If we consider the restriction of this new solution to $\mathcal{C}$, it will be obtained by decreasing the values $\{t_i\}_{i\in\mathcal{C}}$ by $\Delta_{\mathcal{C}}>0$ (see Corollary 1). Since the solution $\{t_i\}_{i\in O_{n-1}}$ cannot be decreased and since the only null-circuit which appears in O_n/O_{n-1} is $\mathcal{C}_n$, the values $\{t_i\}_{i\in\mathcal{C}_n}$ of the nodes of $\mathcal{C}_n$ would be decreased by $\Delta_{\mathcal{C}_n}>0$. It is the case in particular for the value of the *and*-node belonging to $\mathcal{C}_n$ at which the saturated path $\mathcal{P}_n$ ends. It follows that the values $\{t_i\}_{i\in\mathcal{P}_n}$ of all the nodes of the saturated path $\mathcal{P}_n$ would be decreased by a positive value. Since the first node of the path $\mathcal{P}_n$ belongs to O_{n-1} and since we assume that $\{t_i\}_{i\in O_{n-1}}$ is the optimal solution to problem P_{EQ} restricted to O_{n-1}, it is impossible to decrease its value. Hence, the solution $\{t_i\}_{i\in O_n}$ is equal to the optimal solution to problem P_{EQ} restricted to O_n.

Thus, recursively, we prove that the solution $\{t_i\}_{i\in O_n}$ is equal to the optimal solution to problem P_{EQ} restricted to O_n, for $1\leq n\leq p$. Since there is no null-circuit which belongs to $G(V/O_p)$ and since there is a saturated path from s to any node of V/O_p (see condition (i)), according to Theorem 1, the solution $\{t_i\}_{i\in V}$ is equal to the the optimal solution to problem P_{EQ}.

Remarks

As a consequence of the Theorem 3, if there is no path in the graph from node s to some nodes in G, there is no optimal solution to P_{EQ}. This condition is assumed in this paper.

In the optimal solution, the length of the saturated path between node s to node f is equal to the makespan.



The feasible solution (indicated by the numbers located under the nodes) of P_{EQ} is optimal, according Theorem 3.

Figure 8. Illustration of Theorem 3.

Construction of the optimal solution of P_{EQ} starting from a feasible solution

Assume that $\{t_i\}_{i\in V}$ is a feasible solution of P_{EQ}. It can be obtained by applying Algorithm 1. Let N be the set of nodes such that it does not exist a saturated path from either node s or a node belonging to an *and*-saturated null-circuit, to any node of N. Let N_1 be the set derived from from N by applying (a1) to (c1) of Definitions 3 taking into account the arcs saturated by solution $\{t_i\}_{i\in V}$. Let $N_2=V/N_1$. The idea of the algorithm is to decrease the values $\{t_i\}$ of the nodes i which belong to N_1 and to adjust N_1 by adding or removing nodes adequately.

If $N_1=\varnothing$ then $N=\varnothing$ and, according to Theorem 3, $\{t_i\}_{i\in V}$ is the optimal solution to P_{EQ} and also to P_{INEQ}. Otherwise, as seen in the proof of Theorem 3, N_1 satisfies the properties (i) to (iv) of Lemma 2 and $s\notin N_1$. We define both Δ and, for $\alpha\in\,]0;\Delta]$, $\{t_i(\alpha)\}_{i\in V}$, as in the proof of Lemma 2. Any of these solutions is better than $\{t_i\}_{i\in V}$ regarding problem P_{EQ}. Moreover, if $\Delta=+\infty$ then there is no optimal solution.

If $\Delta\neq+\infty$, by definition of Δ, the solution $\{t_i(\alpha=\Delta)\}_{i\in V}$ saturates one arc which was not saturated by solution $\{t_i\}_{i\in V}$. This arc either starts from a node of N_1 and ends with an *or*-node of V/N_1, or starts from a node of V/N_1 and ends with an *and*-node of N_1. In the first case, we derive a new set N_1 from N by applying (a1) to (c1) of Definitions 3 taking into account the arcs saturated by solution $\{t_i(\alpha=\Delta)\}_{i\in V}$. In the second case, for at least one *and*-node of N_1, there exists a saturated path starting either from node s or from a node belonging to an *and*-saturated null-circuit, and ending with this *and*-node. These *and*-nodes can be removed from N_1. Moreover, the set of nodes obtained by applying (a2) to (c2) of Definitions 3, on the set V/N_1 taking into account the arcs saturated by solution $\{t_i(\alpha=\Delta)\}_{i\in V}$ can be removed from N_1.

For each set N_1 obtained along the algorithm, we consider the previous solution $\{t_i(\alpha=\Delta)\}_{i\in V}$ as the new initial solution $\{t_i\}_{i\in V}$. Since the solution is strictly better at each modification of the set N_1, the algorithm converges to the optimal solution, if any.

Since all the nodes are saturated, we have: $A_{NS}=\varnothing$ and $A_S\cap N_1=A\cap N_1$. We have also

$R_{NS}=\varnothing$ and $R_S/N_1=R/N_1$. These simplifications will be used in the notations of steps 4 to 6 in Algorithm 2.

Algorithm 2 - Construction of the optimal solution of P_{EQ}

Input data:

$G=(V,E)$ is the graph under consideration

$\{t_i\}_{i\in V}$ is a feasible solution of P_{EQ}, obtained by applying Algorithm 1.

Output data:

$\{t_i\}_{i\in V}$ is the optimal solution of P_{EQ}, if any.

1. Let N be the set nodes such that it does not exist a saturated path from either node s or a node belonging to an *and*-saturated null-circuit, to such a node.
2. If $N=\varnothing$ then $\{t_i\}_{i\in V}$ is optimal for P_{EQ} and also to P_{INEQ}. End.
3. Derive N_1 from N by applying:
   - (a1) If $i\in N$, then $i\in N_1$, i.e. $N_1\supseteq N$.
   - (b1) If $i\in R$ and if the starting node j of *at least one* saturated arc (j,i) belongs to N_1, then $i\in N_1$.
   - (c1) If $i\in A$ and if the starting node j of *all the* saturated arcs (j,i) belong to N_1, then $i\in N_1$.
3. If $N_1=\varnothing$ then $\{t_i\}_{i\in V}$ is optimal for P_{EQ} and also to P_{INEQ}. End.
4. Set $\Delta_{A\cap N_1}=\begin{cases}+\infty & \text{if } A\cap N_1 \text{ or } Pred(i \text{ s.t. } i\in A\cap N_1)/N_1 \text{ is empty}\\ \min\limits_{i\in A\cap N_1}\left\{\max\limits_{j\in Pred(i)/N_1}\{t_i-t_j-l(j,i)\}\right\} & \text{otherwise.}\end{cases}$
5. Set $\Delta_{R/N_1}=\begin{cases}+\infty & \text{if } R/N_1 \text{ or } Pred(i \text{ s.t. } i\in R/N_1)\cap N_1 \text{ is empty}\\ \min\limits_{i\in R/N_1}\left\{\min\limits_{j\in Pred(i)\cap N_1}\{t_j+l(j,i)-t_i\}\right\} & \text{otherwise.}\end{cases}$
6. Set $\Delta=\min\{\Delta_{A\cap N_1};\Delta_{R/N_1}\}>0$.
7. If $\Delta=+\infty$ then there is no optimal solution. End.
8. Set $t_i=t_i-\Delta$ for $i\in N_1$.
9. Goto step 1.

Proposition

The complexity of the previous algorithm is $O[(|V|+|E|)|V|.Lq]$, where $|V|$ and $|E|$ represent respectively the number of nodes and arcs, q is the greatest common divisor of the lengths of arcs and $L=\sum_{i_0\in V}\sum_{j_0\in Pred(i_0)}|l(j_0,i_0)|$ is equal to the sum of the lengths of the arcs.

Proof

It is possible to find in $O(|V|+|E|)$ the fully connected components of the subgraph limited to the saturated arcs (see [3]). Each component corresponds to either an unique node which

does not belong to a saturated null-circuit, or a set of nodes which belong to a saturated null-circuit. Using a breadth first search (BFS) from node s (see [3]) in the subgraph restricted to the saturated arcs, we find the set V/N of nodes such that it exists a saturated path from either node s or a node belonging to an *and*-saturated null-circuit, to such a node. Note that if a saturated path, starting from either node s or a node belonging to an *and*-saturated null-circuit, ends with an *and*-node i, any saturated null-circuit which contains i is *and*-saturated. Thus, step 1 is computed in $O(|V|+|E|)$.

Steps 2 to 5 of Algorithm 2 are similar to steps 2.2 to 2.5 of Algorithm 1. Steps 6 to 8 of Algorithm 2 are similar to steps 2.8 to 2.10 of Algorithm 1. Thus, steps 2 to 8 of Algorithm 2 can be computed in $O(|V|+|E|)$.

Let we denote by q the greatest common divisor of the lengths of arcs. q is a common divisor of the components of the feasible solution $\{t_i\}_{i\in V}$ computed by Algorithm 1, and is also a common divisor of the components of all the new solutions $\{t_i\}_{i\in V}$ computed at step 8 (see Lemma 2). The criterium of P_{EQ}, which is $\sum_{i\in V} t_i$, strictly decreases by a value greater or equal to $1/q$ at each change of the solution. Thus, step 9 is performed at most $O[(Z_0-\underline{Z})q]$-times, where Z_0 is the value of the criterium for the solution at the beginning of Algorithm 2 and $\underline{Z}$ is the value of the criterium for the optimal solution, if any. Since each component t_i of a solution computed by Algorithm 2 is equal to the length of a path from s to node i, we have for any $i\in V$, $|t_i| \le \sum_{i_0\in V} \sum_{j_0\in Pred(i_0)} |l(j_0,i_0)|$ and $(Z_0-\underline{Z}) \le 2|V| \sum_{i_0\in V} \sum_{j_0\in Pred(i_0)} |l(j_0,i_0)|$. It follows that step 9 is performed at most $O(|V|Lq)$-times, where L is equal to the sum of the lengths of the arcs of G.

Finally, the algorithm requires $O[(|V|+|E|)|V|.Lq]$ operations. QED

6. CONCLUSION

In this paper, we proposed a new type of graphs to handle project management where a new type of activities have to be taken into account, that is alternative activities. We showed that a possible approach consists in finding a feasible solution which is then improved step by step until the optimal solution (if any) is reached.

The most important aspect of this paper, in terms of research, is that it provides results which permits to handle graphs including arcs of any rational value, and null-circuits. Further research will consist in working on the complexity, the goal being to define if a strongly polynomial algorithm exists for this problem.

7. REFERENCES

1. Ahn, J., He, W., and Kusiak, A., Scheduling with alternative operations, *IEEE Journal of Robotics and Automation* (1993) 9, 297-303.

2. Chauvet, F., Levner E., and Proth, J.-M., On PERT Networks with Alternatives, *INRIA Research Report* (1998).

3. Cormen, T.H., Leiserson C.E., and Rivest, R.L., *Introduction to Algorithms*, The MIT Press, Cambridge, Massachussetts (1994).

4. Dinic, E.A., The fastest algorithm for the PERT problem with AND- and OR-nodes. (The New-Product-New-Technology Problem), *Proceedings of the Workshop on Combinatorial Optimization*, Waterloo, 1990, University of Waterloo Press, Waterloo, 1990, 185-187.

5. Gere, W.S., Heuristics in job shop scheduling, *Management Science* (1996) 13, 167-190.

6. Iwata, K., Murotsu, Y., and Oba, F., Solution of large-scale scheduling problems for job-shop machining systems with alternative machine tools, *Annals of CIRP* (1980) 29, 335-338.

7. Kusiak, A., and Finke, G., Selection of process plans in automated manufacturing systems, *IEEE Journal of Robotics and Automation* (1988) 4, 397-402.

8. Levner, E.V., and Nemirovsky, A.S., A network flow algorithm for just-in-time project scheduling, *European Journal of Operational Research* (1994) 79, 167-175.

9. Moder, J.J., Phillips, C.R., and Davis, E.W., *Project Management with CPM, PERT and Precedence Diagramming*, Van Nostrand Reinhold, New York, (1983).

10. Nasr, N., and Elsayed, E.A., Job shop scheduling with alternative machines, *International Journal of Production Research* (1990) 28, 1595-1609.

11. Pan, C.H., and Chen, J.S., Scheduling alternative operations in two-machine flow-shops, *Journal of the Operational Research Society* (1997) 48, 553-540.

12. Shtub, A., Bard, J., and Globerson, C., *Project Management: Engineering Technology and Implementation*, Prentice Hall (1994).

13. Slowinski, R., Weglarz, J., (eds.) *Recent Advances in Project Scheduling*, Elsevier, Amsterdam (1989).

14. Stecke, K.E., and Solberg, J.J., Loading and control policies for a flexible manufacturing system, *International Journal of Production Research* (1981) 19, 481-490.

15. Wilhelm, W.E., and Shin, H., Effectiveness of alternate operations in a flexible manufacturing system, *International Journal of Production Research* (1985) 23, 65-79.

Unité de recherche INRIA Lorraine
Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - B.P. 101 - 54602 Villers lès Nancy Cedex (France)

Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38330 Montbonnot St Martin (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - B.P. 105 - 78153 Le Chesnay Cedex (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - B.P. 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, B.P. 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399

RR_3651