

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE MOHAMED BOUDIAF - M'SILA

FACULTE Mathématique et Informatique

DEPARTEMENT Informatique

N° :



DOMAINE : Mathématique et Informatique

FILIERE : Informatique

OPTION : OMID

Mémoire présenté pour l'obtention
Du diplôme de Master Académique

Par: ZAZGAD AHLEM

Intitulé

Le Chemin Critique Dans L'ordonnancement De
Projet (Etude Comparative)

Soutenu devant le jury composé de :

N.Ould Mehamedi

Université de M'sila

Président

Mouhoub Nasser Eddine

Université de M'sila

Rapporteur

A.Guerna

Université de M'sila

Examineur

Année universitaire : 2016 /2017

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE MOHAMED BOUDIAF - M'SILA

FACULTE Mathématique et Informatique

DEPARTEMENT Informatique

N° :



DOMAINE : Mathématique et Informatique

FILIERE : Informatique

OPTION : OMID

Mémoire présenté pour l'obtention
Du diplôme de Master Académique

Par: ZAZGAD AHLEM

Intitulé

Le Chemin Critique Dans L'ordonnancement De
Projet (Etude Comparative)

Soutenu devant le jury composé de :

N.Ould Mehamedi

Université de M'sila

Président

Mouhoub Nasser Eddine

Université de M'sila

Rapporteur

A.Guerna

Université de M'sila

Examineur

Année universitaire : 2016 /2017

Remerciements

Je remercie avant tous, le dieu tout puissant, de m'avoir aide et de m'avoir donné autant de courage, et de volonté pour la réalisation de ce travail.

Je tiens tout d'abord à remercier Monsieur MOUHOUB NASSER EDDINE pour m'avoir encadré, conseillé, orienté et soutenu tout au long de ma thèse et pour m'avoir guidé et aidé lors de l'écriture de ce manuscrit.

Ce travail n'aurait pas pu d'aboutir à des résultats sans l'aide et les encouragements de plusieurs personnes que nous remercions.

Merci à tous les membres du jury pour avoir évalué mon travail.

Je remercie tous les enseignants du département d'informatique.

Enfin je remercie tous les personnes qui ont contribué de proche ou de loin et sans oublier monsieur ElAihar Abd Alhakim pour l'aide qu'ils m'ont apporté pour la réalisation de ce travail.

Table des matières

Liste des figures

Liste des tableaux

INTRODUCTION GENERALE

CHAPITRE I : L'ORDONNANCEMENT

I.1. Introduction	2
I.2. L'ordonnancement	2
I.2.1. Les étapes de l'ordonnancement	2
I.2.2. Les objectifs de l'ordonnancement	3
I.3. les tâches	3
I.4. les ressources	3
I.5. les contraintes	4
I.5.1. Les contraintes potentielles	4
I.5.2. Les contraintes disjonctives	4
I.5.3. Les contraintes cumulatives	4
I.6. La fonction économique	5
I.7. Caractéristiques générales des ordonnancements	6
I.7.1. Ordonnements statique et dynamique	6
I.7.2. Ordonnements admissibles	6
I.7.3. Ordonnements actifs et semi-actifs	6
I.7.4. Ordonnements sans retard	6
I.7.5. Ordonnement préemptif et non préemptif	6
I.8. Classification des problèmes d'ordonnement	6
I.9. Méthode de résolution des problèmes d'ordonnement	7
I.10. les problèmes d'ordonnement d'atelier	8
I.10.1. Les opérations	9
I.10.2. Les gammes	9
I.10.3. Les machines	9
I.11. conclusion	11

CHAPITRE II : LE PROBLEME CENTRAL DE L'ORDONNACEMENT

II.1. introduction	13
II.2.l'ordonnancement de projet	13
II.3.le problème central de l'ordonnancement	13
II.3.1.Modelisation du problème central	13
II.3.1.1.Le diagramme de Gantt	13
A) Le principe	14
B) La réalisation	14
II.3.1.2.Le graphe des potentiels	15
A) Le principe	15
B) La construction	16
II.3.1.3 La méthode PERT /CPM	17
A) Principe	18
B) Réalisation	18
C) Dessin du graphe PERT	18
D) Notion de tâche fictive	20
II.4. Comparaison	21
II.5.La résolution du problème central	21
II.5.1.Dates et marges associées à une tâche	21
a) Date de début au plus tôt	21
b) Date de début au plus tard	21
c) Date de fin au plus tôt	21
d) Date de fin au plus tard	21
e) Marge totale	22
f) Marge libre	22
II.5.2. Calcul des dates et marges	22
a) Calcul des dates	22
b) Calcul des marges	23
II.5.3.Le chemin critique	23

II.6. Méthode PERT stochastique	24
II.6.1. Le principe du PERT	25
II.7. Conclusion	29

CHAPITRE III : LA LOGIQUE FLOUE

III.1. Introduction	31
III.2. Historique de la logique floue	31
III.3. L'idée de la logique floue	32
III.4. L'utilisation de la logique floue	32
III.5. Logique classique et logique floue	32
III.6. les ensembles flous	33
III.6.1. Définitions	33
III.6.1.1. Sous-ensemble flou	34
III.6.2. Propriétés des ensembles flous	34
A) Support	34
B) Noyau	34
C) Hauteur	35
D) α-coupe	35
III.6.3. Opérations sur les sous-ensembles flous	35
A) Egalité	36
B) Inclusion	36
C) Intersection	36
D) Union	36
E) Complément	36
F) Produit cartésien	36
III.6.4. Normes et conormes triangulaires	37
III.7. Variables linguistiques	37
III.8. Fonctions d'appartenance	38
III.9. Structure générale d'un système flou	39
III.9.1. Fuzzificateur	40
a) Fuzzification singleton	40
b) Fuzzification non – singleton	40
III.9.2. Règles d'Inférence floue	41
III.9.3. Defuzzification	41

a) Défuzzification par centre de gravité	41
b) Défuzzification par centre maximum	42
c) Défuzzification par valeur maximum	42
III.10. Avantages et inconvénients de réglage par logique floue	42
III.10.1. Les Avantages	42
III.10.2. Les Inconvénients	43
III.11. Conclusion	43

CHAPITRE IV : LE PERT FLOU

IV.1. Introduction	45
IV.2. Notions d'arithmétique floue	45
IV.2.1.Ensembles flous	45
IV.2.2.Nombres flous	45
IV.2.3.Alpha-Cuts	46
IV.2.4.Principe d'extension	47
IV.2.5.Théorie des possibilités	48
IV.3.Le PERT flou	49
IV.3.1.Exemple	50
IV.3.2.Difficultés inhérentes au PERT flou	52
IV.4.Conclusion	52

CHAPITRE V: LA COMPARAISON ENTRE LES TROIS

METHODES

V.1. Introduction	54
V.2. Les outils de développement	54
V.2.1.Système d'exploitation	54
V.2.2.Delphi 2007	54
V.2. Le projet	55
V.2.1. L'architecture de projet	55
V.3. Résultats et comparaison	55
V.3.1. Le chemin critique	56
➤ par la méthode CPM	56
➤ par la méthode PERT stochastique	56

➤ par la méthode PERT flou	57
V.3.2. La comparaison	57
V.4. Quelques interfaces de l'application	58
V.4. 1.Fenêtre principale	58
V.4. 2.Fenêtre de travail	58
a) Les méthodes	58
• Nouveau graphe	58
• Ouvrir graphe	59
❖ Calcul par étape	59
❖ Calcul direct	60
b) Quitter	60
V.5. Conclusion	60
CONCLUSION GENERALE	
BIBLIOGRAPHIE	
RESUME	

Liste Des Figures

Figure	Titre	Page
Figure.I.1	les ressources	4
Figure.I.2	les contraintes	5
Figure.I.3	typologies des problèmes d'ordonnancement par les ressources	11
Figure.II.1	Diagramme de Gantt	14
Figure.II.2	Le Diagramme de Gantt	15
Figure.II.3	Le graphe des potentiels	17
Figure.II.4	Le graphe PERT construit par ajout successif des tâches, les durées n'étant pas inscrites.	19
Figure.II.5	sous-exemple pour la tâche fictive	20
Figure.II.6	Exemple complet pour la tâche fictive	20
Figure.II.7	Calcul de do_A dans le cas d'un ou plusieurs arcs.	22
Figure.II.8	Calcul de da_A dans le cas d'un ou plusieurs arcs.	23
Figure.II.9	Les dates de début au plus tôt et au plus tard dans le graphe des potentiels	24
Figure.II.10	Les dates de début au plus tôt et au plus tard dans le graphe PERT	24
Figure.II.11	Graphe associé au projet, avec les durées moyennes	28
Figure.III.1	classification des températures d'une pièce en deux ensembles.	33
Figure.III.2	Ensemble flou	33
Figure.III.3	Les concepts flous décrivant une température tiède.	35
Figure.III.4	Variable linguistique $\{x, T(x), U, G, M\}$ pour décrire la température	38
Figure.III.5	Différentes formes de fonctions d'appartenance	39
Figure.III.6	Structure générale d'un système basé sur la logique floue	40
Figure.IV.1	Représentation d'un nombre flou triangulaire	46
Figure.IV.2	Représentation d'un nombre flou trapézoïdal	46

Figure.IV.3	Alpha-cut	47
Figure.IV.4	Projet à traiter par le PERT flou	50
Figure.IV.5	Nombre flou obtenu pour la durée totale du projet	51
Figure.V.1	l'interface de Delphi7	55
Figure.V.2	Chemin Critique par CPM	56
Figure.V.3	Chemin Critique par PERT stochastique	56
Figure.V.4	Chemin Critique par PERT flou	57
Figure.V.5	Fenêtre principale	58
Figure.V.6	Menu des méthodes	58
Figure.V.7	Saisir un nouveau graphe	58
Figure.V.8	Enregistrer ou abandonner le nouveau graphe	59
Figure.V.9	Chargement du graphe enregistré	59
Figure.V.10	Le calcul par étape	60

Liste Des Tableaux

Tableau	Titre	Page
Table.I.1	Exemple de contraintes cumulatives	5
Table.II.1	Exemple d'ordonnancement (construction d'une voiture)	15
Table.II.2	Exemple d'ordonnancement (construction d'un nouveau bâtiment).	16
Table.II.3	Table de niveaux	16
Table.II.4	Sous-exemple d'ordonnancement	20
Table.II.5	Exemple complet d'ordonnancement	20
Table.II.6	Tableau des données et des résultats.	24
Table.II.7	Tâches intervenant dans la construction d'une maison	27
Table.II.8	Estimations temporelles pour chaque activité	27
Table.II.9	Moyenne et variance pour chaque tâche	28
Table.II.10	Temps au plus tôt attendu et variance pour chaque évènement	28
Table.III.1	Définitions des t-normes et t-conormes les plus utilisées.	37
Table.IV.1	Estimations des durées associées au projet du graphe	50

INTRODUCTION GENERALE

INTRODUCTION GENERALE

La recherche opérationnelle est une discipline dont le but est de fournir des méthodes pour répondre à un type précis de problème. Sa vocation scientifique est donc de construire des modèles formels d'aide à la décision, en particulier les modèles liés à des problèmes d'optimisation, et de proposer des méthodes de résolution efficace de ces modèles.

Ainsi, la théorie de l'ordonnancement est une branche de la recherche opérationnelle qui s'intéresse au calcul de dates d'exécution optimales des tâches. Pour cela, il est très souvent nécessaire d'affecter en même temps les ressources nécessaires à l'exécution de ces tâches.

Dans les dernières années, les problèmes d'ordonnancement ont fait l'objet de nombreuses études. En effet, de nombreuses méthodes de résolutions ont été développées pour résoudre des problèmes de plus en plus complexes et proches de la réalité, nous nous intéressons plus particulièrement aux problèmes d'ordonnancement de projet.

En revanche, la gestion de projet dispose souvent d'un ensemble d'informations toujours entachées d'incertitudes ou d'imprécisions. C'est la raison pour laquelle, il est important de pouvoir exprimer des informations non sûres et imprécises dans le cadre de notre étude.

Plusieurs théories ont été présentées pour modéliser le concept d'incertitude, notamment la plus ancienne est celle de la théorie des probabilités, ainsi que la théorie des ensembles flous présentée par Zadeh Lorfi de l'Université de California Berkeley dans les années soixante. Cette théorie se présente comme un outil privilégié pour la modélisation des situations présentant des imprécisions.

Tenant compte de l'imprécision, la logique floue a été proposée comme méthode rationnelle de manipulation de l'incertain et du qualitatif, présentant un bon compromis entre la souplesse d'emploi et la puissance de représentation, elle permet la construction de modèles simples et efficaces dans les problèmes de prise de décisions, de manipulation de l'information, de gestion de projet ...

L'objectif de ce travail se situe sur le calcul de la durée totale d'un projet dans un problème d'ordonnancement qui se fait par les méthodes MPM, CPM/PERT qu'on verra plus tard.

Dans ce travail on a voulu introduire la logique floue dans le calcul de la durée du projet, faire la comparaison avec les autres méthodes et voir qui est la meilleure.

Nous commençons notre mémoire par une introduction générale qui fixe l'objectif assigné à ce travail.

Ce mémoire s'articule autour de cinq chapitres qui nous permettront de présenter les différents aspects de notre travail.

Dans le **premier chapitre**, on donnera des définitions, les objectifs et les différentes étapes de l'ordonnancement, les caractéristiques générales, puis, la classification des problèmes d'ordonnancement et les problèmes d'ordonnancement d'atelier.

Le **deuxième chapitre**, nous donnerons une brève définition de l'ordonnancement de projet. Par la suite nous étudierons le problème central de l'ordonnancement et on présente les méthodes courantes appliquées pour le résoudre : la méthode des potentiels, le diagramme de Gantt et la méthode PERT/CPM nous verrons comment calculer les dates de début au plus tôt et au plus tard des tâches ainsi que la manière de calculer le chemin critique. Le chapitre sera clôturé par le problème du PERT stochastique Il s'agit de chercher la détermination des durées des tâches du projet qui ne sont pas connues d'une manière exacte par ses trois paramètres : optimistes, pessimistes et les plus probables.

Dans le troisième chapitre, on présentera la théorie de la logique floue. On s'intéressera à l'ensemble flou et ses principes de base. Ensuite, on expliquera la structure générale d'un système d'inférence flou.

Le **quatrième chapitre** présentera la méthode PERT flou, on s'intéresse à l'utilisation des ensembles flous dans le cadre de problèmes pouvant s'apparenter aux problèmes d'ordonnancement de projet, on peut distinguer différentes utilisations des ensembles flous qui nécessitent-elles mêmes des traitements différents. Nous terminerons par les difficultés inhérentes au PERT flou.

Dans le **dernier chapitre**, nous présenterons la comparaison entre les trois méthodes CPM, PERT stochastique, PERT flou et nous verrons quelle est la méthode la plus avantageuse. Ensuite, on présentera l'implémentation de ces trois méthodes.

Finalement, nous clôturons ce mémoire par une conclusion générale contenant un résumé de ce qui a été fait et les perspectives de ce travail.

CHAPITRE I
L'ORDONNANCEMENT

CHAPITRE I

L'ORDONNANCEMENT

I.1. Introduction

La théorie de l'ordonnancement est une branche de la recherche opérationnelle qui s'intéresse au calcul de dates d'exécution optimales de tâches. Pour cela, il est très souvent nécessaire d'affecter en même temps les ressources nécessaires à l'exécution de ces tâches. Un problème d'ordonnancement peut être considéré comme un sous problème de planification dans lequel il s'agit de décider de l'exécution opérationnelle des tâches planifiées.

À cet effet, les problèmes d'ordonnancement consistent à organiser dans le temps l'exécution d'un ensemble de tâches au moyen d'un ensemble de ressources tout en satisfaisant un ensemble de contraintes et/ou en optimisant un ou plusieurs objectifs.

I.2. L'ordonnancement

L'ordonnancement est défini de différentes manières dans plusieurs ouvrages, nous reprenons la définition proposée dans [1] :

" Ordonnancer c'est programmer l'exécution d'une réalisation en attribuant des ressources aux tâches et en fixant leurs dates d'exécution."

Donc l'ordonnancement constitue une solution au problème d'ordonnancement. Il décrit l'exécution des tâches et l'allocation des ressources au cours du temps et vise à satisfaire un ou plusieurs objectifs.

La théorie de l'ordonnancement joue un rôle essentiel dans de nombreux secteurs d'activités : la conception (de bâtiments, de produits, de systèmes, . . .), l'administration (gestion d'emplois du temps, gestion du personnel), l'industrie (gestion de la production), l'informatique (ordonnancement de processus, ordonnancement de réseaux).

On parle d'ordonnancement lorsqu'on fixe les dates de début ou de fin de chacune des tâches alors Un ordonnancement est une solution au problème d'ordonnancement.

I.2.1. Les étapes de l'ordonnancement :

L'ordonnancement se déroule en trois étapes :

- ❖ **La planification** : qui vise à déterminer les différentes opérations à réaliser, les dates correspondantes, et les moyens matériels et humains à y affecter.
- ❖ **L'exécution** : qui consiste à la mise en œuvre des différentes opérations définies dans la phase de planification.

- ❖ **Le contrôle** : qui consiste à effectuer une comparaison entre planification et exécution, soit au niveau des couts, soit au niveau des dates de réalisation.

I.2.2. Les objectifs de l'ordonnancement :

Le traitement de l'ordonnancement dans la littérature s'est tout d'abord orienté vers une optimisation monocritère. L'environnement manufacturier évoluant rapidement et la concurrence devenant de plus en plus acharnée, les objectifs des entreprises se sont diversifiés et le processus d'ordonnancement est devenu de plus en plus multicritère. Les critères que doit satisfaire un ordonnancement sont variés. D'une manière générale, on distingue plusieurs classes d'objectifs concernant un ordonnancement [2] :

- Les objectifs liés au temps: on trouve par exemple la minimisation du temps total d'exécution, du temps moyen d'achèvement, des durées totales de réglage ou des retards par rapport aux dates de livraison,
- Les objectifs liés aux ressources: maximiser la charge d'une ressource ou minimiser le nombre de ressources nécessaires pour réaliser un ensemble de tâches sont des objectifs de ce type,
- Les objectifs liés au coût: ces objectifs sont généralement de minimiser les coûts de lancement, de production, de stockage, de transport, etc.

I.3. les tâches :

Une tâche est une entité élémentaire localisée dans le temps par une date de début et/ou de fin, dont la réalisation nécessite une durée, et qui consomme un moyen selon une certaine intensité.

I.4. les ressources :

La ressource est un moyen technique ou humain destiné à être utilisé pour la réalisation d'une tâche.

On distingue deux types de ressources: les ressources renouvelables et les ressources consommables.

Une ressource est consommable si, après avoir été allouée à une tâche, elle n'est plus disponible pour les tâches suivantes. Le cas pour l'argent, la matière première, etc.

Une ressource est renouvelable si, après avoir été allouée à une tâche, elle redevient disponible après la fin de cette tâche pour les tâches suivantes. C'est le cas pour les machines, les processeurs, les fichiers, le personnel, etc.

Les ressources disjonctives qui ne peuvent exécuter qu'une tâche à la fois (machine –outil, robot manipulateur).

Les ressources cumulatives qui peuvent être utilisées par plusieurs tâches simultanément mais en nombre limite (équipe d'ouvriers, poste de travail).

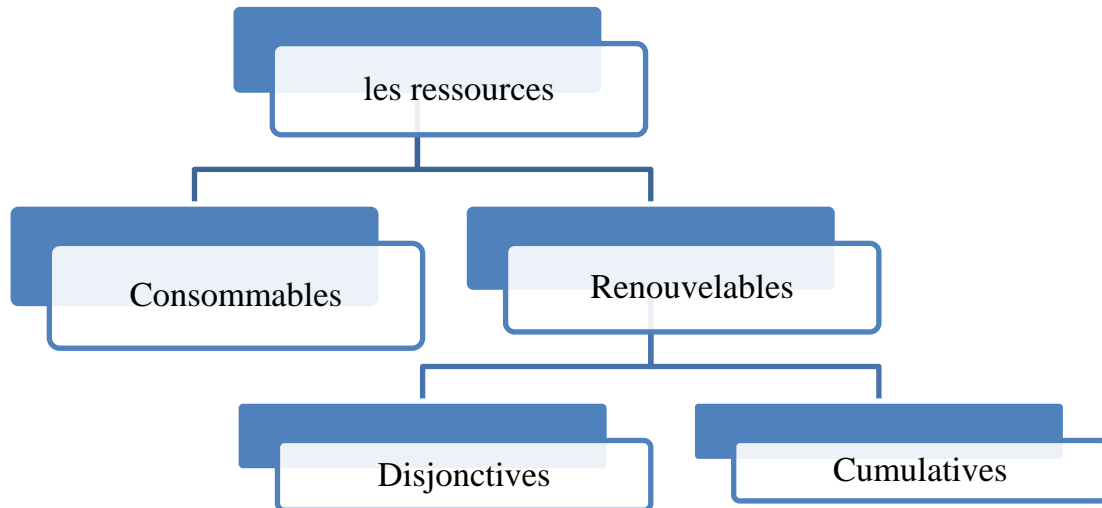


Figure.I.1.les ressources

I.5. les contraintes

Une contrainte exprime une restriction sur les valeurs que peuvent prendre conjointement une ou plusieurs variables de décision sur le temps ou bien sur les ressources ,les contraintes auxquelles sont soumises les diverses tâches qui concourent à la réalisation du projet sont de divers types. On distingue :

I.5.1. Les contraintes potentielles

Elles peuvent être de deux sortes:

- Les *contraintes d'antériorité* selon laquelle une tâche j ne peut commencer avant une tâche i ne soit terminée, par exemple, la construction des piliers suit les fondations.
- Les *contraintes de localisation temporelle* impliquant qu'une tâche donnée i ne peut débiter avant une date imposée, ou qu'elle peut s'achever après une date imposée.

I.5.2. Les contraintes disjonctives

Une contrainte disjonctive impose la non-réalisation simultanée de deux tâches A et B. On trouve de telles contraintes dans le cas d'utilisation d'une ressource présente en un seul exemplaire (une grue, une équipe, etc.) ou pour formuler des interdictions de réalisation simultanée pour des raisons de sécurité ou des problèmes de place. Arbitrer une contrainte disjonctive consiste à décider si A sera fait avant B ou l'inverse. [3]

I.5.3. Les contraintes cumulatives

On parle de contraintes cumulatives lorsque les tâches demandent une partie d'une ou plusieurs ressources présentes en quantité limitée. Le problème est beaucoup plus combinatoire que pour les contraintes disjonctives. Considérons l'exemple où nous avons cinq intervenants et cinq tâches à

effectuer. Chaque tâche demande la présence d'un certain nombre de ces intervenants (voir le tableau) [4]

Tâche	A	B	C	D	E
Nombre d'intervenants	4	3	2	1	1

Table.I.1. Exemple de contraintes cumulatives

Pour que l'ordonnancement soit réalisable, il faut qu'on utilise, à tout moment, au plus cinq intervenants. Cette contrainte va interdire les ordonnancements réalisant en parallèle : (A//B),(A //C), (A//D//E), (B//C//D), (B//C//E). Ces configurations sont minimales au sens que, si (A //B) est interdit, toute configuration contenant (A//B) l'est aussi (par exemple : (A//B//D), (A//B//C//D) etc.

Lorsqu'une configuration minimale porte sur deux tâches (ici A//B ou A//C), on peut la remplacer par une contrainte disjonctive entre ces deux tâches. [3]

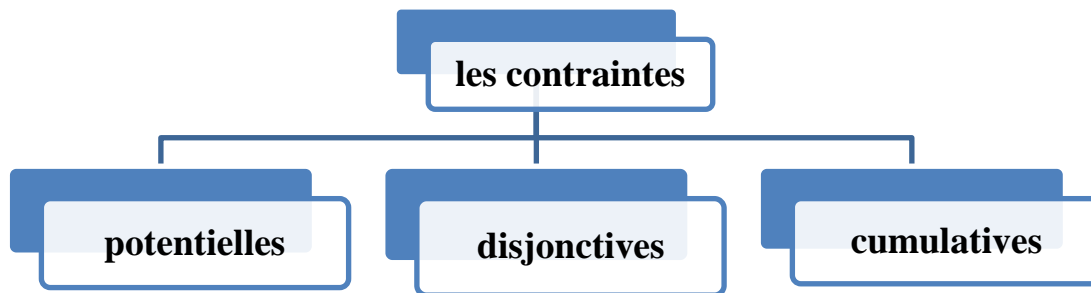


Figure.I.2. les contraintes

I.6. La fonction économique

Lorsqu'on cherche la résolution d'un problème d'ordonnancement, on est amené à décider entre deux familles de principes. Le premier vise à chercher l'optimalité de la solution par rapport à un critère ou même à plusieurs critères, et le second cherche l'admissibilité de cette solution par rapport aux contraintes.

L'approche d'optimisation suppose que les solutions proposées sont classées dans un certain ordre par rapport à un ou plusieurs critères d'évaluation. Ce classement permet de mesurer la qualité des solutions.

Donc, selon le domaine d'application, la fonction ordonnancement peut avoir d'autres objectifs que celui de veiller au simple respect des contraintes de temps et de ressource. Il peut s'agir de satisfaire des objectifs économiques, de respecter une législation du travail en vigueur dans une entreprise ou de gérer au mieux le risque en présence d'incertitudes.

I.7. Caractéristiques générales des ordonnancements

I.7.1. Ordonnements statique et dynamique

Si l'ensemble des informations nécessaires à la résolution d'un ordonnancement est fixé au départ (ensembles des tâches, des contraintes, des ressources, etc.), on est alors devant un problème d'ordonnement *statique*.

Il y a une nuance entre la solution proposée qui est, généralement accompagnée d'un plan prévisionnel d'exécution des tâches, et l'exécution réelle de ces tâches. Si le plan n'est pas respecté et les objectifs sont modifiés, on est devant un problème d'ordonnement *dynamique* qui nécessite une résolution d'une série de problèmes statiques et chaque étape doit débiter par une prise d'informations permettant d'actualiser le modèle à résoudre.

I.7.2. Ordonnements admissibles

Un ordonnancement est dit admissible s'il respecte toutes les contraintes du problème (dates de début, dates de fin, précédence, contraintes de ressources, etc.).

I.7.3. Ordonnements actifs et semi-actifs

Dans un ordonnancement actif, aucune tâche ne peut commencer au plus tôt et qui entraîne l'ordre relatif entre au moins deux tâches. Dans l'ordonnement semi actif, on ne peut pas avancer une tâche sans modifier la séquence sur la ressource.

I.7.4. Ordonnements sans retard

Dans un ordonnancement sans retard, on doit exécuter la tâche qui est en attente à condition que la ressource soit disponible.

I.7.5. Ordonnement préemptif et non préemptif

Selon les problèmes, les tâches peuvent être exécutées sans interruption, c'est-à-dire si on commence l'exécution d'une tâche elle n'est pas interrompu jusqu'à son achèvement. On parle alors d'un ordonnancement préemptif. Par contre, si les tâches sont exécutées par morceaux, l'ordonnement est appelé non préemptif.[5]

I.8. Classification des problèmes d'ordonnement

Selon la nature des variables mises en jeu, la nature des contraintes, ou encore la structure particulière du graphe non-conjonctif associé à un problème, plusieurs classifications des problèmes d'ordonnement sont proposées dans la littérature.

Dans le domaine de l'ordonnement d'atelier, la notion structurante de gamme opératoire est souvent utilisée comme critère de classification. En effet, une gamme opératoire impose un ordre de passage des produits sur les machines, et donc un ordre des opérations associées à chaque travail. De ce fait, le graphe potentiels-tâches non conjonctif associé au problème possède parfois une

structure particulière, qu'il est possible d'exploiter pour élaborer des méthodes d'ordonnancement efficaces. Dans [6], sept types de problèmes d'ordonnancement d'atelier sont ainsi distingués :

- 1- le problème job shop où chaque travail à sa gamme opératoire propre.
- 2- le problème flow shop où chaque travail à une gamme identique ; le problème open shop où l'ordre de passage sur les machines est libre pour chaque travail ;
- 3- le problème open shop où l'ordre de passage sur les machines est libre pour chaque travail ;
- 4- le problème flow shop de permutation où chaque travail à une gamme identique et où l'ordre de passage des travaux est le même pour chaque machine ;
- 5- le problème à une machine où chaque travail est assimilé à une opération unique, exécutée par une seule et même machine m de capacité $Q_m = 1$;
- 6- le problème à machines parallèles où chaque travail est assimilé à une opération unique, exécutée par une machine à sélectionner dans un ensemble ;
- 7- le problème job shop à machines dupliquées où chaque travail à sa gamme opératoire propre et où chaque opération est réalisée par une machine à sélectionner dans un ensemble.

I.9. Méthode de résolution des problèmes d'ordonnancement

Plusieurs auteurs [6], [7] partitionnent les méthodes de résolution en trois classes : les méthodes optimales efficaces, les méthodes énumératives et les méthodes heuristiques. Les méthodes optimales efficaces garantissent, pour un problème et un critère donnés, la détermination d'une solution optimale en un temps de calcul polynomial, i.e. le temps maximal consommé pour trouver une solution optimale est une fonction polynomiale des données du problème (nombre de ressources, de tâches). De telles méthodes ne sont évidemment disponibles que pour des classes réduites de problèmes d'ordonnancement. Parmi les plus connues, citons :

- la règle SPT (Shortest Processing Time) pour le problème $1//\Sigma C_j$;
- la règle WSPT (Weighted Shortest Processing Time) pour le problème $1//\Sigma w_j C_j$;
- la règle EDD (Earliest Due Date) pour le problème $1/d_j/ L_{\max}$;
- l'algorithme de Moore et Hodgson pour le problème $1/d_j/\Sigma U_j$;
- l'algorithme de Johnson pour le problème $F2/prmu/C_{\max}$.

Selon [8], les méthodes optimales énumératives procèdent quant à elles à une énumération partielle de l'espace de recherche. Leur complexité temporelle théorique est généralement exponentielle, mais elles fournissent en pratique, sur des problèmes de taille moyenne, des solutions optimales en un temps raisonnable. Dans cette classe, on peut distinguer :

- les Procédures par Séparation et Évaluation (PSE) qui énumèrent par une recherche arborescente un ensemble de solutions, en éliminant les branches de l'arbre de recherche montrées non-optimales (utilisation de bornes inférieure et supérieure du critère) ;
- les méthodes de Programmation Linéaire (PL), modélisant les critères et les contraintes comme des fonctions linéaires de variables mixtes (réelles, entières) ;
- les méthodes basées sur la Programmation Dynamique (PD) qui procèdent à une décomposition en sous-problèmes, que l'on résout optimalement à rebours, en tenant compte à chaque étape des informations issues de la résolution du sous-problème précédent.

Les méthodes heuristiques sont souvent utilisées pour traiter des problèmes que les méthodes optimales sont incapables de résoudre en un temps acceptable. Elles produisent généralement une solution faisable de bonne qualité en un temps relativement court. La qualité d'une heuristique doit être évaluée sur plusieurs jeux d'instance de taille variable afin de mesurer d'une part, la déviation moyenne du critère par rapport à sa valeur optimale, et d'autre part, l'évolution du temps de calcul en fonction de la taille ou de la structure du problème. Ces heuristiques sont classiquement classifiées en trois grands types :

- les algorithmes gloutons dans lesquels les décisions d'ordonnancement sont prises progressivement, à temps croissant, au fur et à mesure que les ressources se libèrent, grâce à des règles de priorité simples de type SPT, EDD, etc. Et pour lesquels on ne remet jamais en cause une décision qui a été prise.
- les méthodes de recherche locale (tabou, recuit simulé, algorithmes génétiques, etc.) qui, partant d'une solution initiale, définissent un voisinage, qui est ensuite exploré pour trouver des solutions meilleures ; (en s'autorisant parfois à dégrader la solution courante pour augmenter les chances d'obtenir une solution meilleure).
- les méthodes de recherche arborescente tronquée, proches des PSE, excepté que l'arbre de recherche est volontairement restreint, quitte à perdre des solutions optimales, afin de gagner en temps de calcul [9].

I.10. les problèmes d'ordonnancement d'atelier

Un problème d'ordonnancement d'atelier consiste à trouver une séquence de passage d'un certain nombre de tâche ou travaux à exécuter sur différentes machines de façon à satisfaire des contraintes technologiques et à optimiser un ou plusieurs critères de performances. Il s'agit de prévoir le travail à exécuter, de façon à coordonner l'utilisation des matières premières et des moyens de production, c'est-à-dire à faire en sorte que tout soit prêt au moment voulu.

Un problème d'ordonnancement consiste à effectuer des tâches à des moyens de fabrication au cours du temps pour effectuer un ensemble de travaux de manière à optimiser certain critère, tous en respectant les contraintes techniques de fabrication.

La solution d'un problème d'ordonnancement général doit répondre à deux questions :

- Quand ?
- Avec quels moyens ?

L'énoncé général d'un problème d'ordonnancement d'atelier comporte quatre types d'information :

- 1- Les produits et les opérations à effectuer.
- 2- Les quantités et les type de ressources.
- 3- Les contraintes endogènes (gamme, capacités) et exogènes (délais).
- 4- Les critères pour évaluer la solution

I.10.1. Les opérations :

Une opération est une succession d'actions élémentaires réalisables de manière consécutive sur une machine. C'est une phase de fabrication du produit que l'on ne souhaite pas détailler plus finement au niveau du problème d'ordonnancement considéré. Un produit est terminé si et seulement si toutes les opérations de la gamme sont réalisées.

On distingue trois types d'opérations :

1. **Les opérations non interruptibles** : une fois qu'une opération commence à être exécutée sur une machine, l'exécution ne peut pas être interrompue avant qu'elle ne soit terminée.
2. **Les opérations interruptibles** : l'interruption est autorisée. Pour finir cette opération, il faut qu'elle se poursuive à partir de l'état où elle a été interrompue.
3. **Les opérations préemptives répétées** : l'interruption est autorisée, mais l'exécution d'une opération doit être recommencée depuis le début. C'est-à-dire que le travail déjà fait avant l'interruption a été perdu.

I.10.2. Les gammes :

Si le choix de types de machines pour les opérations n'est pas imposé, on dit que la gamme est une gamme logique, sinon on dit qu'une gamme technique.

I.10.3. Les machines :

On distingue trois types de machines :

- 1) Machines identiques

Par exemple, un ordinateur biprocesseur, toute opération peut être effectuée sur n'importe quelle machine parmi les machines parallèles.

2) Machines uniformes ou a vitesses proportionnelles

Dans ce cas, les machines peuvent effectuer les mêmes types d'opérations avec des performances différentes : les temps nécessaires pour effectuer une opération sont différents d'une machine à l'autre.

3) Machines non reliées :

Il s'agit de machines polyvalentes. Les machines sont partiellement interchangeable. Par exemple, une machine peut être conçue spécialement pour certains types d'opérations, mais elle peut aussi en effectuer d'autres avec une performance moins bonne que les machines dédiées à ces opérations.

On distingue les trois types d'ateliers suivants :

- Les ateliers de type flow-shop

Appelés également ateliers a cheminement unique, ce sont des ateliers ou une ligne de fabrication est constitué de plusieurs machines en série ; toutes les opérations de toutes les tâches passent par les machines dans le même ordre.

- Les ateliers de type job shop :

Appelés également ateliers a cheminement multiple, ce sont des ateliers ou les opérations sont réalisées selon un ordre bien déterminé, variant selon la tâche a exécuter, le job shop flexible est une extension du modèle job shop classique ;sa particularité réside dans le fait que plusieurs machines sont potentiellement capable de réaliser un sous ensemble d'opérations.

- Les ateliers de type open shop

Ce type d'atelier est moins contraint que celui de type flow shop ou de type job shop. Ainsi, l'ordre des opérations n'est pas fixe à priori ; le problème d'ordonnancement consiste, d'une part, a déterminer le cheminement de chaque produit et, d'autre part, à ordonnancer les produit en tenant compte des gammes trouvées, ces deux problèmes pouvant être résolus simultanément. Compare aux autres modèles d'ateliers, l'open shop n'est pas couramment utilise dans les entreprises.

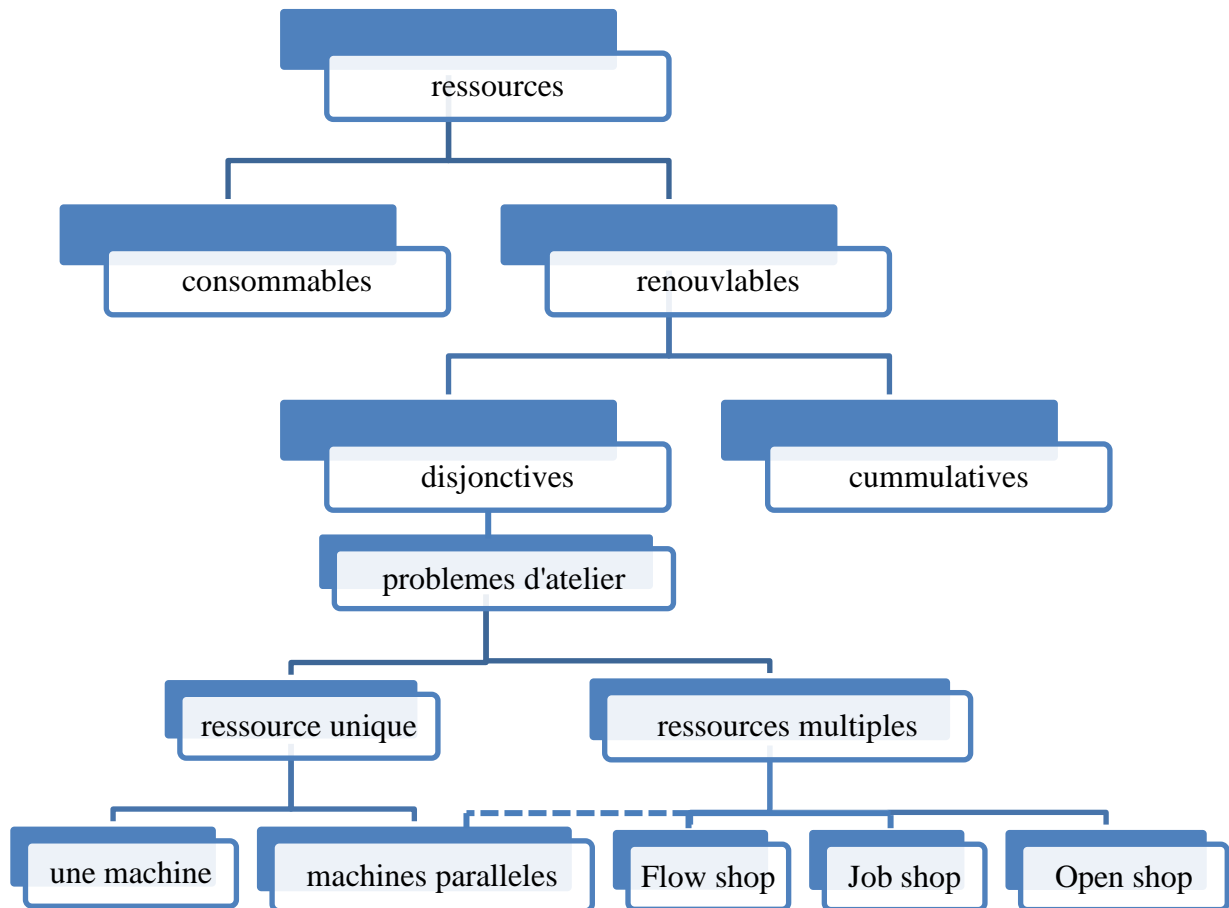


Figure.I.3.typologies des problèmes d'ordonnancement par les ressources

I.11. conclusion

L'ordonnancement est généralement décrit comme une fonction particulière de décision au sein d'un système de gestion du travail concernant la production de bien, d'ouvrages ou de services. Le problème d'ordonnancement consiste à effectuer des tâches à des moyens de fabrication au cours du temps pour effectuer un ensemble de travaux de manière à optimiser certains critères, tout en respectant les contraintes techniques de fabrication.

Le problème d'ordonnancement consiste à trouver une séquence de passage d'un certain nombre de tâches ou travaux à exécuter sur différentes machines de façon à satisfaire des contraintes technologiques et à optimiser un ou plusieurs critères de performances.

CHAPITRE II

LE PROBLEME CENTRAL DE L'ORDONNACEMENT

CHAPITRE II

LE PROBLEME CENTRAL DE L'ORDONNACEMENT

II.1. introduction

Dans ce chapitre, on s'intéresse à l'ordonnancement de projet, nous donnerons une définition de la gestion de projets, le projet (les caractéristiques, les acteurs, cycle de vie...).

Nous présenterons les trois méthodes de modélisation du problème central : le diagramme de GANTT, la méthode des potentiels et la méthode PERT.

Avant de présenter ces méthodes de résolution il faut d'abords donner une définition précise le problème central. Pour chacune de ces méthodes, nous verrons comment représenter le problème a l'aide d'un graphe, comment calculer les dates de début au plus tôt et au plus tard des tâches ainsi que la manière de calculer le chemin critique et comment l'analyse du chemin critique peut être utile pour réduire la durée du projet.

II.2.l'ordonnancement de projet

Ordonnancer un projet consiste à programmer dans le temps l'exécution des tâches et des ressources, tout en respectant les contraintes de projet et de manière à optimiser la succession des tâches du projet afin de réduire les couts est les délais.

II.3.le problème central de l'ordonnancement

Pour le problème central de l'ordonnancement, il s'agit d'ordonnancer en une durée minimale un ensemble $I = \{1, 2, \dots, n\}$ de tâches soumises à des contraintes temporelles de type inégalité de potentiel.

Donc, dans le problème central de l'ordonnancement, on ne tient compte que des contraintes de potentiels qui sont d'antériorité et de durée. Les contraintes disjonctives et cumulatives sont ignorées. Les ressources sont supposées à quantité illimitée.

II.3.1.Modelisation du problème central

Il existe actuellement trois méthodes de modélisation du problème central : le diagramme de Gantt, la méthode des potentiels et la méthode PERT/CPM.

II.3.1.1.Le diagramme de Gantt

Le diagramme de Gantt est la plus ancien des trois techniques de planification. Elle porte le nom de son créateur Henry Laurence GANTT, ingénieur et consultant américain, qui l'a mise au point en 1917. Désigne également sous l'appellation : Graphe a barres, planning de charge ou graphique

d’ordonnement, il a pour but de visualiser l’utilisation des ressources dans le temps afin d’optimiser leur affectation.

Le diagramme de Gantt est un outil permettant de modéliser la planification de tâches nécessaires à la réalisation d’un projet.

Le diagramme de Gantt utilise un repère orthogonal dans un plan. L’axe des abscisses représente le temps et l’axe des ordonnées représente l’ensemble des machines. Pour chaque machine, on représente la séquence de tâches effectuées dans le temps. Les parties en gris représentent les périodes d’oisiveté d’une machine, dues aux éventuelles indisponibilités des tâches.

La figure II.1 représente un ordonnancement à deux machines M1 et M2 et quatre tâches T1, T2, T3, T4 de durées d’exécutions respectives 1, 2, 4,5.

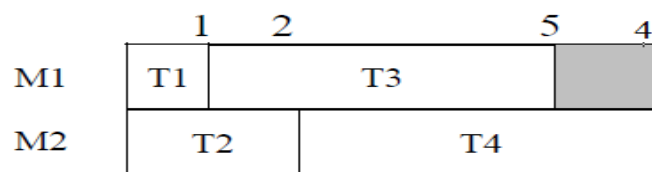


Figure.II.1.Diagramme de Gantt

A) Le principe :

On représente au sein d’un tableau, en ligne les différentes tâches et en colonne les unités de temps (exprimées en mois, semaines, jours, heures...)

La durée d’exécution d’une tâche est matérialisée par un trait au sein du diagramme.

B) La réalisation :

On désigne quatre étapes pour réaliser le diagramme de Gantt :

- Première étape : On détermine les différentes tâches à réaliser et leur durée.
- Deuxième étape : On définit les relations d’antériorité entre les tâches.
- Troisième étape : On représente d’abord les tâches n’ayant aucune antériorité, puis les tâches dont les tâches antérieures ont déjà été représentées, et ainsi de suite...
- Quatrième étape : On représente par un trait parallèle en pointillé à la tâche planifiée la progression réelle du travail.

Exemple

Pour la construction d’une voiture, on doit exécuter les activités {A, B, C et D} rapportées du tableau suivant :

Repère	Désignations	Durée(en semaine)	Activités Antérieures
A	Elaboration de croquis	3	-
B	Construction du châssis	6	A
C	Construction du moteur	4	A
D	Montage moteur châssis	2	B, C
E	Finition	8	D

Table.II.1. Exemple d’ordonnancement (construction d’une voiture)

Le diagramme de Gantt de cet exemple :

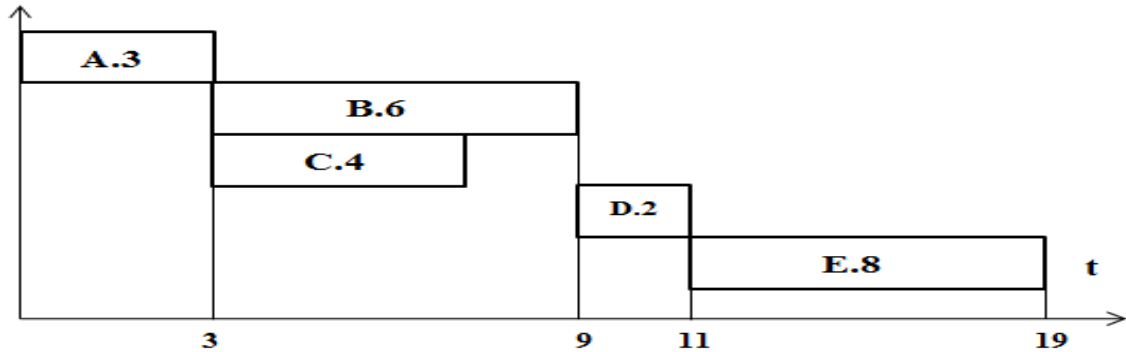


Figure.II.2. Le Diagramme de Gantt

Jusqu’à l’année 1958 on utilisait le diagramme de Gantt pour traiter les problèmes d’ordonnancement. Cette année-là, se sont développées en parallèle deux méthodes fondées sur la théorie des graphes pour ordonnancer les travaux. La méthode des potentiels de Bernard ROY appelée méthode française, et la méthode PERT/CPM appelée méthode américaine.

Algorithmiquement, les deux méthodes sont équivalentes. Elles se distinguent seulement par la manière d’écrire le graphe de réseau. Nous les présenterons dans ce qui suit.

II.3.1.2. Le graphe des potentiels

La méthode des potentiels a été développée vers la fin des années 50 parallèlement à la méthode PERT. Elle est appelée également la méthode MPM (méthode des potentiels Metra) ou encore méthode des potentiels –tâches.

Cette technique d’ordonnancement basée sur la théorie des graphes, visant à optimiser la planification des tâches d’un projet.

Ainsi, le principe de la méthode est le même et donc la méthodologie à adopter sera sensiblement identique à celle du PERT. Seule la représentation diffère.

A) Le principe :

- Les tâches sont représentées par des sommets et les contraintes de succession par des arcs.
- Chaque tâche est renseignée par la date à laquelle elle peut commencer (date au plus tôt) et celle à laquelle, elle doit se terminer (date au plus tard).

- A chaque arc est associée une valeur numérique, qui représente soit une durée d’opération, soit un délai.

B) La construction

Pour construire un graphe MPM, on procède par des niveaux :

- On place d’abord le sommet « α » ;
- On identifie les tâches sans antécédentes (tâche de niveau 1) et les relie au sommet « α » ;
- On identifie ensuite les tâches de niveau 2 (dont les antécédents sont exclusivement les tâches du niveau 1) et on les positionne sur le graphe en les reliant à leurs antécédents ;
- On continue ainsi pour chaque niveau suivant jusqu’à ce que toutes les tâches aient pu être positionnées sur le graphe et reliées entre elles ;
- Les tâches n’ayant pas de descendant sont à la fin reliées au sommet « ω ».

Exemple :

code	Tâche	Durée (jours)	Antériorités
α	Début	0	-
1	Terrassement	5	α
2	Fondation	4	1
3	Colonnes porteuses	2	2
4	Charpente toiture	2	3
5	Couverture	3	4
6	Maçonnerie	5	3
7	Plomberie, électricité	3	2
8	Coulage dalle béton	3	7
9	Chauffage	4	8 et 6
10	Plâtre	10	9 et 5
11	Finitions	5	10
ω	Fin	0	11

Table.II.2. Exemple d’ordonnement (construction d’un nouveau bâtiment).

Le tableau de niveaux :

Niveaux	1	2	3	4	5	6	7	8	9
Tâche	α	1	2	3,7	4, 6, 8	5, 9	10	11	ω

Table.II.3. Table de niveaux

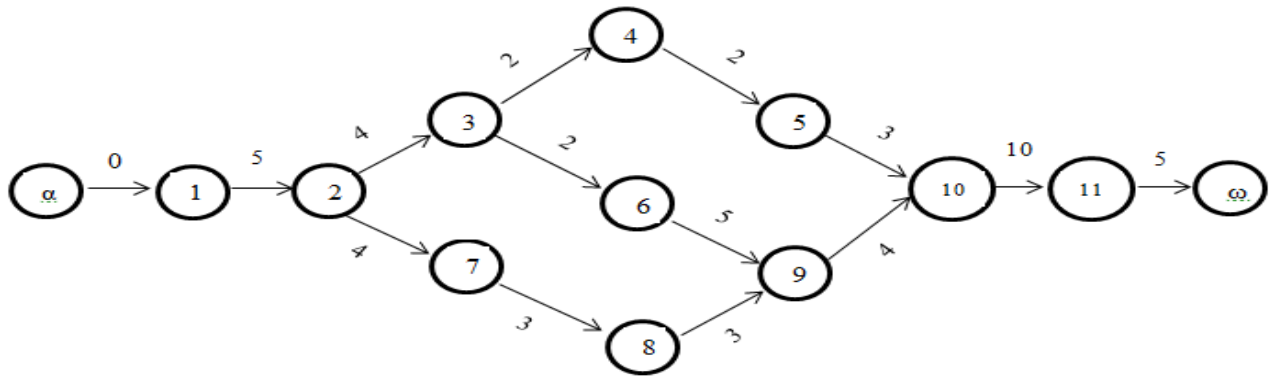


Figure.II.3.Le graphe des potentiels

II.3.1.3 La méthode PERT /CPM

PERT et le CPM ont été indépendamment développés vers la fin des années 50 depuis, ils ont été parmi les plus largement appliqués.

Les versions originales de PERT et du CPM ont eu quelques différences importantes. Cependant, elles également ont eu beaucoup de commun, et les deux techniques ont été graduellement fusionnées au cours des années. En conséquence, les praticiens emploient maintenant généralement les deux noms l'un pour l'autre, ou les combinent dans l'acronyme simple PERT/CPM.

L'application de la technique du PERT a permis de réduire ce délai à quatre ans.

Selon [10] PERT et CPM se diffèrent par la nature de l'information requise, par la capacité de prévoir la date de fin du projet et d'estimer les chances de minimiser le risque de retard dans l'exécution du projet.

Dans CPM, l'estimation des durées respectives se fait de façon déterministe - c'est le cas par exemple des projets de construction qui sont en général tributaires de ressources et s'appuient sur des estimations précises - alors que PERT considère que les durées des activités sont des variables aléatoires et qu'en conséquence leur estimation se fait de façon probabiliste [11], [12], [13]. CPM se prête plus aisément à l'analyse du compromis coût/durée alors que l'un des avantages majeurs du PERT est de prendre en compte l'incertitude et le risque qui entourent les durées respectives du projet (et donc le temps de réalisation du projet) et de calculer la probabilité de compléter le projet à temps [14]. Évidemment PERT est employé dans la gestion des délais et des échéanciers des projets de R&D ou de tout autre projet complexe ou de grande taille où les durées des activités sont sujettes à une forte variabilité et donc à un degré élevé d'incertitude sur leur temps de réalisation.

La méthode PERT est une technique de gestion de projet qui permet de visualiser la dépendance des tâches et de précéder à leur ordonnancement ; c'est un outil de planification. On utilise un graphe de dépendances. Pour chaque tâche, on indique une date de début et de fin plus tôt et au plus

tard. Le diagramme permet de déterminer le chemin critique qui conditionne la durée minimale du projet.

Son but est de trouver la meilleure organisation possible pour qu’un projet soit terminé dans les meilleurs délais, et d’identifier les tâches critiques, c’est-à-dire les tâches qui ne doivent souffrir d’aucun retard sous peine de retarder l’ensemble du projet.

A) Principe

Dans un graphe PERT :

- Chaque tâche est représentée par un arc, auquel on associe un chiffre entre parenthèses qui représente la durée de la tâche.
- Entre les arcs figurent des cercles appelées «sommet » ou « évènement » qui marquent l’aboutissement d’une ou plusieurs tâches. Ces cercles sont numérotés afin de suivre l’ordre de succession des divers évènements.

B) Réalisation

Pour construire un graphe PERT, on utilise la méthode des niveaux.

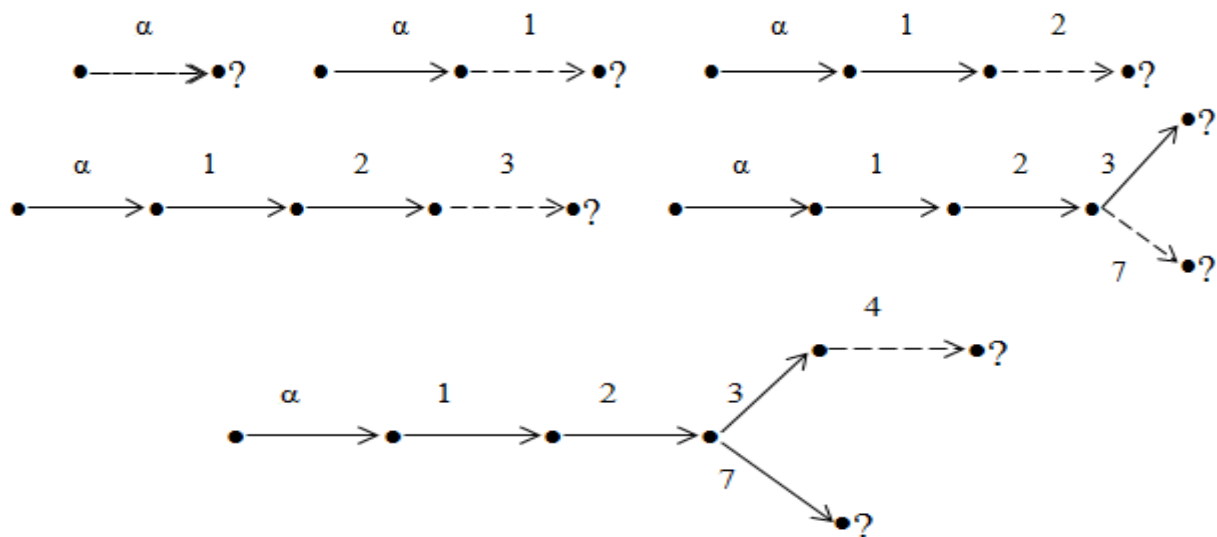
- On détermine les tâches sans antécédents, qui constituent le niveau 1.
- On identifie ensuite les tâches dont les antécédents sont exclusivement du niveau.
- Ces tâches constituent le niveau 2, et ainsi de suite...

C) Dessin du graphe PERT

Généralement, le praticien trouve des difficultés à dessiner le graphe. La figure suivante (Figure II.6) présente une technique nouvelle de construction du graphe PERT. Elle est plutôt pédagogique

Exemple :

Pour illustrer cette technique, considérons l’ordonnement du tableau II.2.



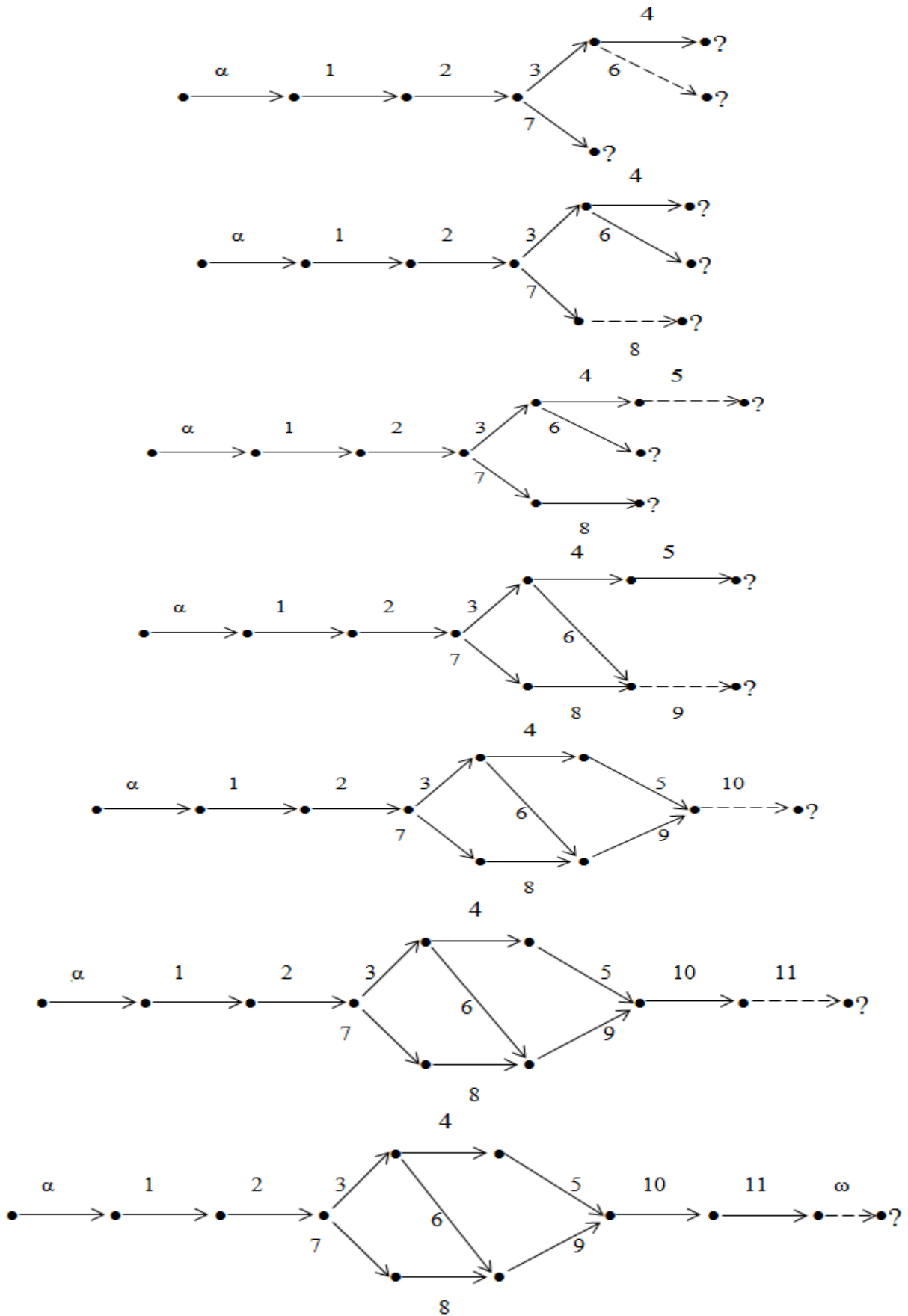


Figure.II.4. Le graphe PERT construit par ajout successif des tâches, les durées n’étant pas inscrites.

D) Notion de tâche fictive

Une tâche fictive représente une contrainte entre tâche non indépendantes. Chaque tâche fictive est représentée par une flèche pointillée, sa durée est nulle, elle ne consomme aucun ressource, elle ne coute donc rien.

Par exemple :

Tâche	Tâche antérieures
A	-
B	A
C	-
D	A, C

Table.II.4.Sous-exemple d’ordonnement

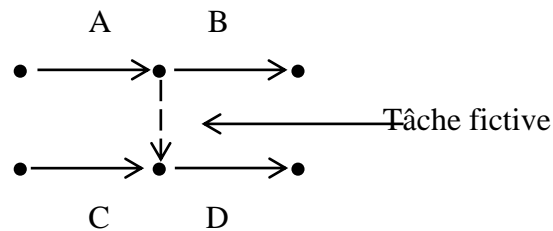


Figure.II.5. sous-exemple pour la tâche fictive

Les tâches A et C sont convergentes et précèdent la tâche D, C ne précède que D mais, en revanche A précède non seulement D mais aussi B. Alors il est nécessaire d’ajouter une tâche fictive.

Exemple :

Tâche	Tâche antérieures
α	-
A	α
B	A
C	A
D	A
E	A, B
F	A, C, D
ω	E, F

Table.II.5.Exemple complet d’ordonnement

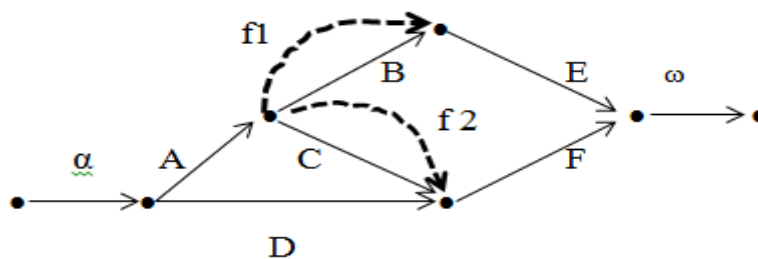


Figure.II.6.Exemple complet pour la tâche fictive

II.4. Comparaison

Nous venons de décrire la modélisation du problème central de l'ordonnancement par les graphes, et plus précisément par la méthode des potentiels et la méthode PERT.

Un inconvénient de cette dernière par rapport à la méthode des potentiels est de doubler initialement les sommets du graphe puisqu'on associe à chaque tâche une étape « début de tâche » et une étape « fin de tâche ».

Une autre difficulté apparaît, comme on l'a vu précédemment, il est souvent nécessaire d'introduire de nombreuses tâches fictives.

De plus, à un même problème correspondent plusieurs graphes PERT plus ou moins simples. N'oublions pas qu'il y a une infinité de graphes PERT à cause des tâches fictives qu'on peut introduire comme on veut et là où on veut puisqu'elles ne nécessitent aucun moyen matériel ou financier qui influe sur la durée ou le coût global du projet.

Parmi les avantages de cette méthode est qu'elle est plus facilement lisible par les praticiens car une tâche est représentée par un arc et un seul. Donc, le nombre de tâches représente le nombre d'arcs réels.

De plus, en disposant ces arcs horizontalement et en leur donnant une longueur proportionnelle à la durée de la tâche, on présente le graphe PERT sous forme de diagramme de Gantt [15].

II.5. La résolution du problème centrale

II.5.1. Dates et marges associées à une tâche

a) Date de début au plus tôt :

La date de début au plus tôt d'une tâche A do_A est la date minimum à laquelle peut commencer. C'est donc la date avant laquelle un événement ne peut se réaliser.

b) Date de début au plus tard

La date de début au plus tard d'une tâche A da_A est la date limite de sa réalisation. Après quoi le projet sera retardé.

c) Date de fin au plus tôt

La date de fin au plus tôt d'une tâche A fo_A est la date à laquelle A se termine en commençant à sa date de début au plus tôt.

d) Date de fin au plus tard

La date de fin au plus tard d'une tâche A fa_A est la date minimum des dates de fin au plus tard des tâches qui lui succèdent. Elle est la date à laquelle A se termine si elle commence à sa date de début au plus tard.

e) Marge totale

La marge totale de A, notée mt_A , est le délai qui peut être accordé à A pour son commencement sans qu’il y ait un retard dans la réalisation du projet.

f) Marge libre

La marge libre de A, notée ml_A , est le délai accordé au commencement de A sans la modification des marges totales des tâches postérieures. Un retard supérieur à la marge libre se répercute sur les tâches suivantes en diminuant leurs marges libres.

II.5.2. Calcul des dates et marges

Une tâche ne peut démarrer que lorsque les tâches de tous les chemins de α à A sont exécutées, en particulier celles du plus long chemin de α à A dont la longueur est la date minimum de début d’exécution de A, c’est-à-dire la date de début au plus tôt do_A .

L’exécution d’un projet ne se termine donc que lorsque toutes ses tâches sont exécutées, en particulier celles de plus long chemin de α à ω .

Les dates de début au plus tôt des tâches d’un problème d’ordonnancement sont donc déterminées à partir de l’arborescence des plus longs chemins du graphe des potentiels préparé ou du graphe PERT.

Les graphes étant sans circuits, nous appliquerons l’algorithme de Belleman pour déterminer les dates de début au plus tôt. Les durées sur les arcs de même extrémité initiale étant ∞ égales dans le graphe des potentiels préparé, on vérifie que les tâches ayant les mêmes antériorités ont les mêmes dates de début au plus tôt.

a) Calcul des dates

Pour calculer la date de début au plus tôt de la tâche A do_A on procède comme suit :

- La date de début au plus tôt de la tâche α est initialisée à 0
- pour les autres tâches :
 - Soit il y a un seul arc entre A et B alors : $do_B = do_A + t_A$
 - Soit il y a k arcs ($k \geq 2$) alors : $do_B = \max (do_{A1} + t_{A1}, \dots, do_{Ak} + t_{Ak})$

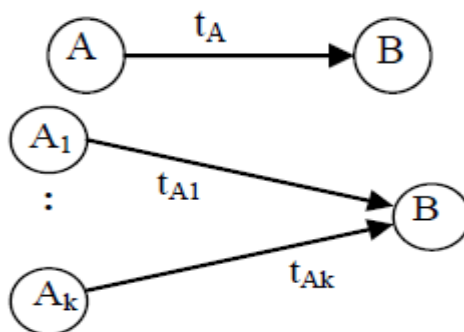


Figure.II.7. Calcul de do_A dans le cas d’un ou plusieurs arcs.

Pour la date de début au plus tard de la tâche A da_A , la technique est la suivante :

- La date de début au plus tard de la tâche ω est initialisée à sa date de début au plus tôt c.à.d.

$$da_\omega = do_\omega$$

- Pour les autres tâches :

- soit il y a un seul arc entre A et B alors : $da_A = da_B - t_A$
- soit il y a plusieurs arcs qui sortent de A alors : $da_A = \min (da_{B1}, \dots, da_{Bk}) - t_A$

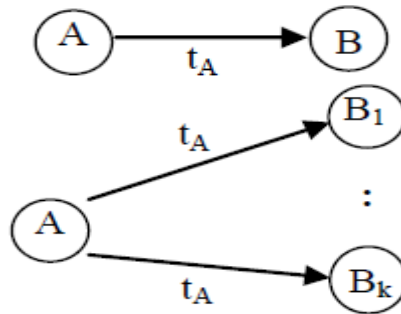


Figure.II.8. Calcul de da_A dans le cas d'un ou plusieurs arcs.

Pour le calcul de la date de fin au plus tôt de la tâche A notée fo_A , elle s'obtient aisément en ajoutant la durée de la tâche A à sa date de début au plus tôt : $fo_A = do_A + d_A$.

La date de fin au plus tard de la tâche A fa_A s'obtient en ajoutant sa durée à sa date de début au plus tard : $fa_A = da_A + t_A$.

b) Calcul des marges

Pour chaque tâche, on obtient l'intervalle de placement en faisant la différence entre la date de début au plus tard et la date de début au plus tôt de la même tâche.

$$ml_A = da_A - do_A$$

La marge libre d'une tâche A est la date de début au plus tôt minimum qui succèdent à A diminuée de sa date de fin au plus tôt.

$$ml_A = \min (do_{A1}, do_{A2}, \dots, do_{Ak}) - fo_A$$

Les tâches A_1, A_2, \dots, A_k sont les tâches qui succèdent à A.

La date de marge libre de A est la date de début au plus tôt de A augmentée de sa marge libre:

$$dml_A = do_A + ml_A$$

II.5.3. Le chemin critique :

Le chemin critique est composé de tâches qui ont une marge totale égale à zéro, voire négative. Par conséquent, le retard des tâches du chemin critique impact la date de fin du projet. Le chemin critique est souvent le chemin le plus long du projet, c'est-à-dire le chemin qui définit la durée globale du projet. Pourtant, plusieurs chemins critiques peuvent exister dans un projet, définis pour des livrables qui contiennent un degré de risque élevé ou qui ont une importance stratégique pour la réussite du projet.

Prenons l’exemple d’ordonnancement (construction d’un nouveau bâtiment) voir le tableau II.2

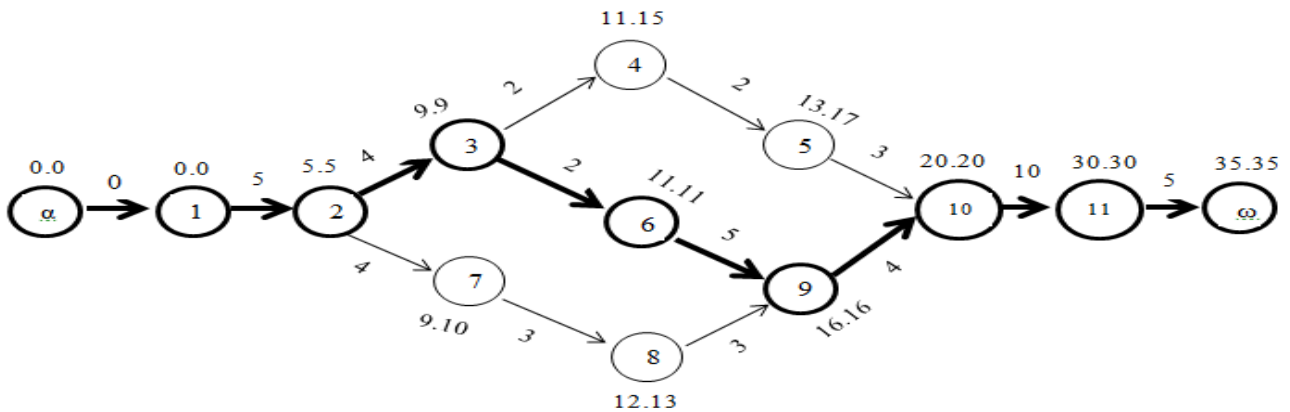


Figure.II.9. Les dates de début au plus tôt et au plus tard dans le graphe des potentiels

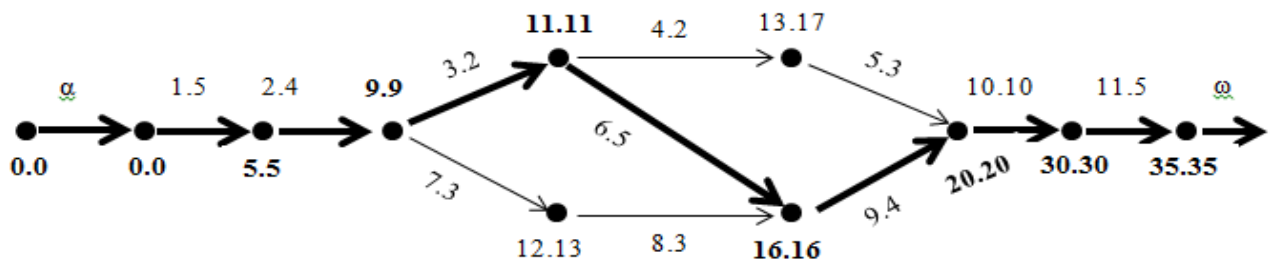


Figure.II.10. Les dates de début au plus tôt et au plus tard dans le graphe PERT

Codes	Durée	Antériorité	Au plus tôt		Au plus tard		Marges	
			Début	Fin	Début	Fin	Libre	Totale
α	0	-	0	0	0	5	0	0
1	5	α	0	5	0	9	0	5
2	4	1	5	9	5	11	0	4
3	2	2	9	11	9	12	0	2
4	2	3	9	11	10	20	1	2
5	3	4	13	16	17	16	4	3
6	5	3	11	16	11	13	0	5
7	3	2	9	12	10	16	1	7
8	3	7	12	15	13	16	1	3
9	4	8,6	16	20	16	20	0	4
10	10	9,5	20	30	20	30	0	10
11	5	10	30	35	30	35	0	5
ω	0	11	35	35	35	35	0	0

Table.II.6. Tableau des données et des résultats.

II.6. Méthode PERT stochastique

Par opposition aux durées déterministes de la méthode du chemin critique, dans la méthode PERT les durées des opérations sont des variables aléatoires supposées indépendantes.

Généralement, les distributions de probabilité des durées pour chacune des tâches sont inconnues et le raisonnement proposé pour parer à cette difficulté est détaillé dans la section suivante.

II.6.1. Le principe du PERT stochastique

Pour modéliser l’incertitude sur la durée des tâches, nous nous basons sur trois estimations pour chaque activité [16] :

- a : la durée minimale de l’opération, qui correspond à l’estimation la plus optimiste,
- m : la durée la plus probable de l’opération,
- b : la durée maximale de l’opération, qui correspond à l’estimation la plus pessimiste.

Ces données peuvent être estimées par des experts dans le cadre d’un projet, et généralement, m est plus proche de a que de b, vu que b est plus difficile à déterminer et est donc choisi “prudemment” assez grand [17]. Cette approche montre qu’en général, la prévision (m) est optimiste, car elle ne tient pas compte des imprévus et retards possibles dans l’exécution des tâches. Ainsi, en pratique, nous avons $b - m \geq m - a$, voire $b - m \gg m - a$.

- La moyenne :
$$E(T) = \frac{(a+b)/2 + m}{3} = \frac{a+b+4m}{6} \quad (1)$$

- La variance :
$$var(T) = \left(\frac{b-a}{6}\right)^2 \quad (2)$$

La méthode du chemin critique peut alors être appliquée en considérant que la durée d’une tâche est donnée par sa valeur moyenne.

Soit μ_i le temps au plus tôt auquel peut avoir lieu l’évènement i dans un projet. Comme les durées des activités précédant i sont des variables aléatoires, μ_i est également une variable aléatoire. En supposant que toutes les activités du projet sont stochastiquement indépendantes, nous pouvons trouver la moyenne et la variance de μ_i par le raisonnement suivant [16].

S’il n’existe qu’un seul chemin allant du nœud de début de projet au nœud i, $E\{\mu_i\}$ est donné par la somme des durées moyennes de toutes les activités le long de ce chemin et $var\{\mu_i\}$ est la somme des variances de ces activités. Si par contre il y a plusieurs chemins entre le nœud 1 et le nœud i, il faudrait théoriquement d’abord développer la distribution statistique pour le chemin le plus long (c’est-à-dire la distribution du maximum de plusieurs variables aléatoires) avant d’enfin trouver la valeur exacte de la moyenne $E\{\mu_i\}$ et de la variance $var\{\mu_i\}$. Cependant, ceci étant généralement difficile, une hypothèse simplificatrice est introduite pour calculer $E\{\mu_i\}$ et $var\{\mu_i\}$: ces valeurs sont supposées égales à celles qui correspondent au chemin arrivant au nœud i et ayant la somme des durées moyennes la plus élevée.

Nous utiliserons cette hypothèse dans l’exemple de la page suivante. Notons que si plusieurs chemins ont la même valeur de $E\{\mu_i\}$, il est recommandé de considérer celui pour lequel la variance $var\{\mu_i\}$ est la plus grande, puisqu’il correspond à une plus grande incertitude et mène de ce fait à des résultats plus “prudents”. Nous avons donc finalement :

$$E\{\mu_i\} = ES(i) \quad (3)$$

$$var\{\mu_i\} = \sum_{k \in K} var(k) \quad (4)$$

Où K représente l'ensemble des activités sur le plus long chemin jusqu'au nœud i .

Comme μ_i est une somme de variables aléatoires indépendantes, nous pouvons utiliser le théorème de centrale-limite ([18], [19]) que nous rappelons ci-dessous :

Soient X_1, X_2, \dots, X_n n variables aléatoires indépendantes, n 'étant un entier suffisamment grand.

Soient $Y = X_1 + X_2 + \dots + X_n$,

$$\mu = \sum_{i=1}^n \mu_{x_i} \text{ Et } \sigma^2 = \sum_{i=1}^n \sigma_{x_i}^2 \quad (5)$$

Où μ_{x_i} et $\sigma_{x_i}^2$ sont la moyenne et la variance de la variable aléatoire X_i , respectivement.

Alors, $\frac{y-\mu}{\sigma}$ aura approximativement une distribution normale $N(0, 1)$.

Donc, pour n assez grand,

$$p\left(\frac{y-\mu}{\sigma} \leq z\right) \cong p(Z \leq z) \quad (6)$$

Où Z est une variable aléatoire normale standard. Notons que généralement, pour des distributions individuelles assez symétriques, cette approximation devient satisfaisante des $n = 10$ [18].

Dans notre cas, nous pouvons donc supposer que μ_i a approximativement une distribution normale de moyenne $E\{\mu_i\}$ et de variance $VAR\{\mu_i\}$. Comme μ_i représente un temps au plus tôt pour l'évènement i , cet évènement va avoir lieu avant un temps t_i avec une probabilité

$$p\{\mu_i \leq t_i\} = \left\{ \frac{\mu_i - E\{\mu_i\}}{\sqrt{var\{\mu_i\}}} \leq \frac{t_i - E\{\mu_i\}}{\sqrt{var\{\mu_i\}}} \right\} = p\{Z \leq K_i\} \quad (7)$$

Où Z est la distribution normale de moyenne nulle et de variance 1, et :

$$K_i = \frac{t_i - E\{\mu_i\}}{\sqrt{var\{\mu_i\}}} \quad (8)$$

Exemple :

Nous allons expliquer les différentes notions vues dans les sections précédentes dans le cadre d’un projet de construction d’une maison. Les tâches intervenant dans la construction sont reprises dans le tableau ci-dessous ainsi que leur relation d’antériorité entre les tâches.

nom	Tâche	Tâche antérieures
A	Réalisation des fonctions	-
B	Construction des murs et plafonds	A
C	Fixation des poutres pour le toit	B
D	Revêtement	C
E	Câblage électrique	B
F	Pose des tuiles	D
G	Protection extérieures	H
H	Pose des fenêtres	B
I	Installation des planches intérieures	E, H
J	Mise en peinture	F, G, I

Table.II.7. Tâches intervenant dans la construction d’une maison

Pour chaque tâche, trois estimations temporelles, reprise dans le tableau ci-dessous.

Activité (i, j)	a_{ij}	m_{ij}	b_{ij}
A	1	3	5
B	1	5	9
C	2	2	3
D	2	3	4
E	2	4	7
F	6	8	11
G	4	5	6
H	1	2	3
I	1	3	6
J	1	2	4

Table.II.8. Estimations temporelles pour chaque activité

Pour chaque tâche, nous pouvons calculer la durée moyenne ainsi que la variance en utilisant les relations (1) et (2). Les résultats de ces calculs sont repris au tableau II.9. Avec ces valeurs, le graphe de la Figure II.11, ou les durées des tâches ont changé.

Activité (i, j)	E	VAR
A	3	0.44
B	5	1.78
C	2.17	0.03
D	3	0.11
E	4.17	1.69
F	8.17	0.69
G	5	0.11
H	2	0.11
I	3.17	0.69
J	2.17	0.25

Table.II.9. Moyenne et variance pour chaque tâche

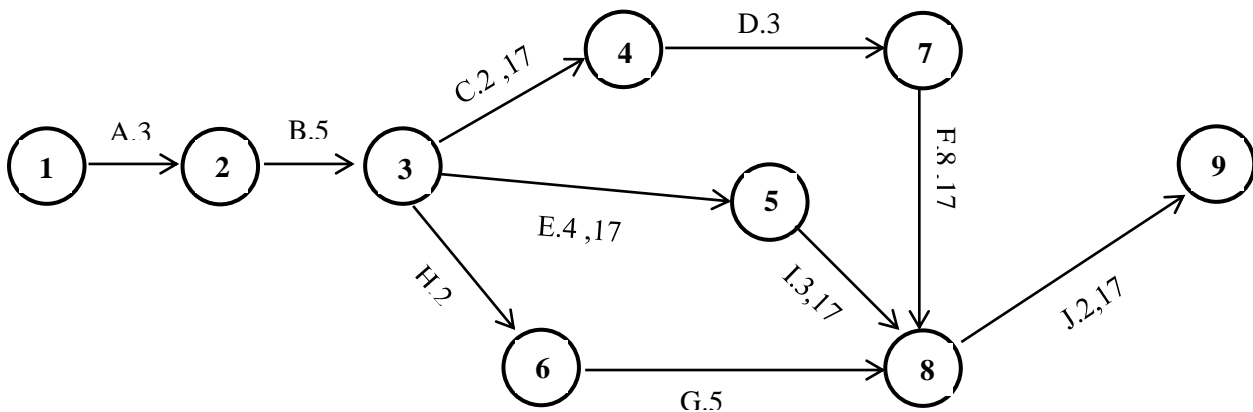


Figure.II.11. Graphe associé au projet, avec les durées moyennes

Intéressons-nous maintenant aux temps au plus tôt pour les évènements.

Nœud i	Chemin	E { μ_i }	VAR { μ_i }
1	1	0	0
2	1 - 2	3	0.44
3	1 - 2 - 3	8	2.22
4	1 - 2 - 3 - 4	10.17	2.25
5	1 - 2 - 3 - 4 - 5	7.17	2.91
6	1 - 2 - 3 - 4 - 6	10	2.33
7	1 - 2 - 3 - 4 - 7	13.17	2.36
8	1 - 2 - 3 - 4 - 7 - 8	21.34	3.05
9	1 - 2 - 3 - 4 - 7 - 8 - 9	23.51	3.30

Table.II.10. Temps au plus tôt attendu et variance pour chaque évènement

Avec :

$$K_9 = \frac{23 - 23.51}{\sqrt{3.30}} = -0.2807 \quad (9)$$

Et donc

$$P\{\mu_9 \leq 23\} \leq 38.95\% \quad (10)$$

Le chemin de longueur maximale, la valeur attendue en moyenne et la variance sont repris au tableau II.10 pour chaque évènement. La moyenne et la variance ont été calculées avec les relations (3) et (4). Nous voyons sur ce tableau que le chemin critique est celui qui parcourt les nœuds 1 - 2 - 3 - 4 - 7 - 8 - 9. Nous pouvons encore calculer la probabilité que la durée totale du projet soit inférieure à 23 semaines en appliquant la formule (7).

Ainsi :

$$P\{\mu_9 \leq 23\} \leq P\{Z \leq K_9\} \quad (11)$$

II.7. Conclusion

Nous avons étudié dans ce chapitre les différentes modélisations des problèmes d'ordonnement de projet dans la littérature ainsi que sa résolution. Cette étude est accompagnée de plusieurs points nouveaux tels que le dessin du graphe PERT.

L'objectif de l'ordonnement central est non seulement d'organiser les activités dans le temps, mais également de permettre aux décideurs d'entreprises d'estimer les dates finales de livrables. Dès lors, l'ordonnement de projet revêt un intérêt capital de par les estimations qu'il propose et qui servent de base à la prise de décision et à la coordination des ressources internes, des approvisionnements, des relations de sous-traitance et du suivi du projet.

CHAPITRE III

LA LOGIQUE FLOUE

CHAPITRE III

LA LOGIQUE FLOUE

III.1. Introduction

La logique floue, ou plus généralement le traitement des incertitudes, est l'une des classes de l'intelligence artificielle. Elle a pour objet l'étude et la représentation des connaissances imprécises et le raisonnement approché.

Cette théorie est très attractive, parce qu'elle est basée sur le raisonnement intuitif et prend en compte la subjectivité et l'imprécision. Mais ce n'est pas une théorie imprécise. C'est une théorie mathématique rigoureuse, adaptée au traitement de tout ce qui est subjectif et/ou incertain.

Ce chapitre commence par une brève historique de la logique floue. On s'intéresse à l'ensemble flou et ses principes de base. Ensuite, on expliquera la structure générale d'un système d'inférence flou.

III.2. Historique de la logique floue

Au départ la théorie de la logique floue, s'affirmait comme une technique opérationnelle. Utilisée à côté d'autres techniques de contrôle avancé, elle fait une entrée discrète mais appréciée dans les automatismes de contrôle industriel. Les bases théoriques de la logique floue ont été formulées au début des années 1965 par le professeur Lotfi A. Zadeh, de l'Université de Berkeley en Californie [20]. Il a introduit la notion de sous-ensemble flou pour fournir un moyen de représentation et de manipulation des connaissances imparfaitement décrites, vagues ou imprécises. A cette époque, la théorie de la logique floue n'a pas été prise au sérieux excepté par quelques experts.

En 1970, c'est la première application de la logique floue dans les systèmes experts d'aide à la décision en médecine, puis en 1975 Mamdani et Assilian publient les premiers résultats permettant une exploitation de cette théorie dans des systèmes de réglage [21]. Les japonais [22], en 1985 étaient les premiers à avoir utilisé la logique floue dans les produits grand public « Fuzzy Logic Inside ».

Freinée par la réticence à appréhender un nouveau concept et par des difficultés de réalisation pratique, la logique floue ne débouche réellement qu'en 1990 avec l'apparition de produits et de systèmes utilisant cette technique. Et en 1995 grâce à Jang, l'application de la logique floue s'élargit aux systèmes à réseaux de neurones et à l'intelligence artificielle.

Aujourd'hui, la logique floue est arrivée à maturité et utilisée dans nombreuses applications industrielles et gestionnaires. Sa mise en œuvre est maintenant facilitée par la disponibilité de microprocesseurs dédiés et d'outils puissants de développement.

III.3. L'idée de la logique floue

Comme la science s'appuie sur la notion de mesure, la question qui se pose est : comment représenter les valeurs non mesurables? Comment représenter ce qui est incertain ou subjectif? Comment représenter les termes du langage humain?. Les descriptions linguistiques d'un système sont souvent vagues. Mais le flou n'est pas imprécis. Si une donnée n'est pas connue précisément, elle peut être exprimée par un intervalle de confiance précis. Cet intervalle est un ensemble de valeurs possible pour la donnée.

III.4. L'utilisation de la logique floue

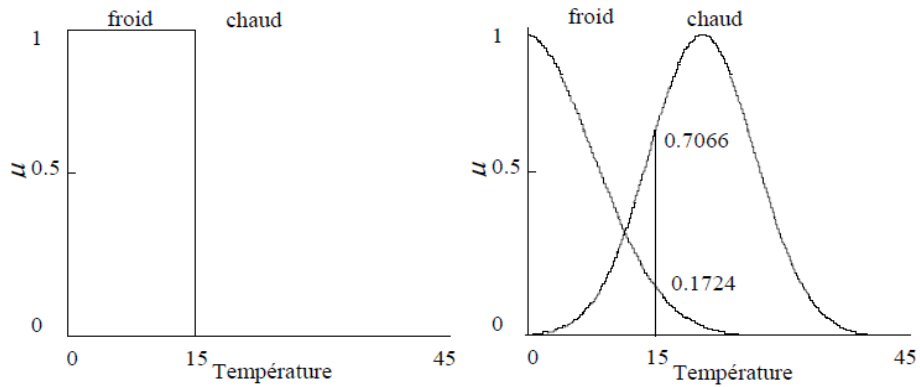
La logique floue est une technique de résolution de problèmes très puissants avec une large applicabilité dans le control et la prise de décision [23]. Elle est très utile lorsque le modèle mathématique du problème à traiter n'existe pas ou existe mais difficile à implémenter, ou il est trop complexe pour être évalué assez rapidement pour des opérations en temps réel [24]. Ou bien lorsque des experts humains sont disponibles pour fournir des descriptions subjectives du comportement du système avec des termes en langage naturel. La logique floue est aussi supposée de travailler dans les situations où il y a de large incertitude et des variations inconnues dans les paramètres et la structure du système.

III.5. Logique classique et logique floue

Dans le cadre de la logique classique, une proposition est soit vraie, soit fausse (1 ou 0).

Par exemple, la logique classique peut facilement partitionner la température d'une pièce en deux sous-ensembles, «moins de 15 degrés» et «15 degrés ou plus». La figure 1.1a montre le résultat de cette partition. Toutes les températures de moins de 15 degrés sont alors considérées comme appartenant à l'ensemble «moins de 15 degrés». On leur affecte une valeur de 1. Toutes les températures atteignant 15 degrés ou plus ne sont pas considérées comme appartenant à l'ensemble «moins de 15 degrés». On leur attribue une valeur de 0.

Cependant, le raisonnement humain s'appuie fréquemment sur des connaissances ou des données inexacts, incertaines ou imprécises. Une personne placée dans une pièce dont la température est soit de 14.95 degrés soit de 15.05 degrés, ne fera certainement pas de distinction entre ces deux valeurs. Cette personne sera pourtant capable de dire si la pièce est «froide» ou «chaude», sans pour cela utiliser de température limite ni de mesure précise.



(a) Deux ensembles selon la logique classique (b) Deux ensembles selon la logique floue

Figure.III.1: classification des températures d’une pièce en deux ensembles.

La logique floue permet de définir des sous-ensembles, comme «froid» ou «chaud», en introduisant la possibilité pour une valeur d’appartenir plus ou moins à chacun de ces sous-ensembles.

III.6. les ensembles flous

La théorie des ensemble flous est une théorie mathématique, elle a été introduire par Lotfi Zadeh en 1965, qui montre que cette théorie est un cas particulier de la théorie des sous ensemble classiques ou les fonctions d’appartenance considérées prennent des valeurs binaires ($\{0,1\}$). Dans la théorie ensembliste classique, l’appartenance d’un élément à un sous-ensemble est définie par une valeur logique standard : 1 si l’élément appartient au sous-ensemble, 0 sinon. Dans la théorie floue, un élément peut appartenir en partie à un sous-ensemble : son degré d’appartenance est décrit par une valeur comprise entre 0 et 1.

III.6.1. Définitions

La notion d’ensemble flou a pour but de permettre l’idée d’une appartenance partielle d’un élément à un ensemble ou à une classe, c’est-à-dire d’autoriser un élément à appartenir plus ou moins fortement à cette classe.

Soit l’ensemble U des valeurs de la variable x , appelé l’univers du discours ; un sous ensemble A de U et une fonction $\mu_A(x)$ compris entre 0 et 1.

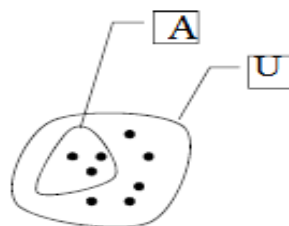


Figure.III.2.Ensemble flou

III.6.1.1. Sous-ensemble flou :

Un sous-ensemble flou A dans un univers du discours X est caractérisée par sa fonction d'appartenance $\mu_A(x)$ qui associe à chaque élément x de X une valeur dans l'intervalle des nombres réels $[0, 1]$.

$$\mu_A : X \rightarrow [0, 1] \quad (12)$$

Ainsi un sous-ensemble flou A dans X peut être représenté par un ensemble de couples ordonnés

$$A = \{(x, \mu_A(x)) | x \in X\} \quad (13)$$

Le sous-ensemble classique n'est en fait qu'un cas particulier de sous-ensemble flou dont la fonction d'appartenance ne prend que les valeurs 0 ou 1. Un sous-ensemble flou A de X est aussi souvent représenté par la notation suivante qui indique pour tout élément x de X son degré $\mu_A(x)$ d'appartenance à A :

$$A = \int_X \mu_A(x)/x \quad \text{si } X \text{ est continu} \quad (14)$$

et

$$A = \sum_{x_i \in X} \mu_A(x_i)/x_i \quad \text{si } X \text{ est discret} \quad (15)$$

Comme les valeurs $\mu_A(x)$ représentent les degrés d'appartenance avec lesquels les x_i appartiennent à A , si $\mu_A(x)$ prend la valeur 1 pour tous les éléments de X , cela signifie que A est identique à X . Au contraire, A est vide si $\mu_A(x)$ prend la valeur 0 sur tout X .

III.6.2. Propriétés des ensembles flous :

A) Support :

Le support d'un sous-ensemble flou A dans un univers du discours X est le sous-ensemble (au sens classique du terme) des éléments de X pour lesquels la fonction d'appartenance prend une valeur strictement positive. C'est l'ensemble des éléments de X qui appartiennent au moins un peu à A :

$$S(A) = \{x | \mu_A(x) > 0\} \quad (16)$$

B) Noyau :

Le noyau d'un sous-ensemble flou A dans un univers du discours X est le sous-ensemble des éléments de X pour lesquels la fonction d'appartenance vaut 1. C'est l'ensemble des points qui appartiennent intégralement à A :

$$N(A) = \{x | \mu_A(x) = 1\} \quad (17)$$

C) Hauteur :

La hauteur d'un sous-ensemble flou A dans un univers du discours X est la valeur maximale prise par la fonction d'appartenance $\mu_A(x)$ sur l'ensemble X. C'est le plus fort degré avec lequel un élément de X appartient à A :

$$H(A) = \sup_{x \in X} \mu_A(x) \quad (18)$$

D) α -coupe :

Pour toute valeur α de l'intervalle $[0, 1]$, on appelle α -coupe d'un sous-ensemble flou A de X, le sous-ensemble noté A_α des éléments de X pour lesquels la fonction d'appartenance est supérieure ou égale à α :

$$A_\alpha = \{x \in X \mid \mu_A(x) \geq \alpha\} \quad (19)$$

Ce sous-ensemble est défini par la fonction caractéristique suivante :

$$X_{A_\alpha} = 1 \quad \text{si et seulement si } (x) \geq \alpha \quad (20)$$

Si nous choisissons $\alpha = 0$, alors A_α est l'univers du discours X. Si nous choisissons $\alpha = 1$, alors A_α est le noyau de A, $N(A)$. Sur la figure III.3 nous illustrons les définitions précédentes par un exemple. Le sous-ensemble flou A est celui des températures tièdes dans l'univers du discours X des températures.

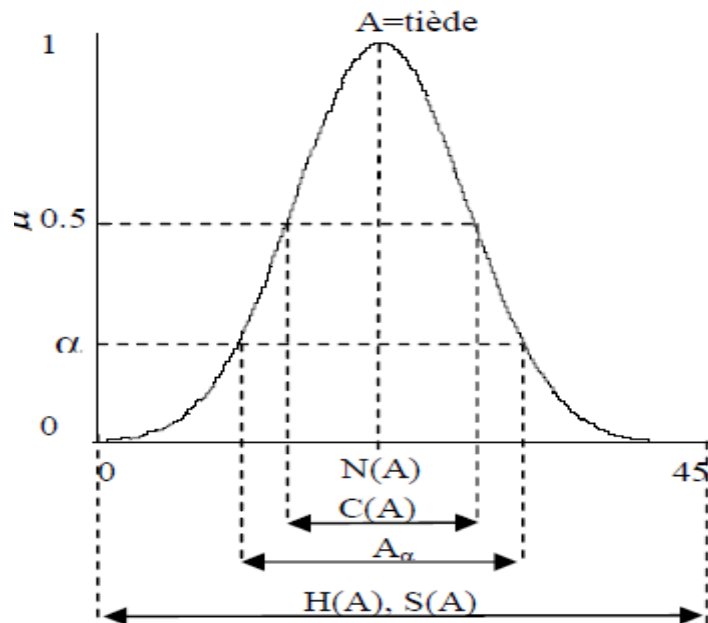


Figure.III.3. Les concepts flous décrivant une température tiède.

III.6.3. Opérations sur les sous-ensembles flous :

Supposons que A et B sont deux sous-ensembles flous définis dans un univers du discours X par les fonctions d'appartenance μ_A et μ_B . On peut définir des opérations ensemblistes telles que l'égalité, l'inclusion, l'intersection, l'union et le complément grâce à des opérations sur les fonctions d'appartenance.

A) Egalité :

A et B sont dits égaux, propriété que l'on note $A = B$, si leurs fonctions d'appartenance prennent la même valeur en tout point de X :

$$\forall x \in X \quad \mu_A(x) = \mu_B(x) \quad (21)$$

B) Inclusion:

A est dit inclus dans B, propriété que l'on note $A \subseteq B$, si tout élément x de X qui appartient à A appartient aussi à B avec un degré au moins aussi grand :

$$\forall x \in X \quad \mu_A(x) \leq \mu_B(x) \quad (22)$$

C) Intersection :

L'intersection de A et B, que l'on note $A \cap B$, est le sous ensemble flou constitué des éléments de X affectés du plus petit des deux degrés d'appartenance μ_A et μ_B :

$$\forall x \in X \quad \mu_{A \cap B} = \min(\mu_A(x), \mu_B(x)) = \mu_A(x) \wedge \mu_B(x) \quad (23)$$

Dans cette définition, min et \wedge désignent l'opérateur de calcul du minimum des deux valeurs.

D) Union :

L'union de A et B, que l'on note $A \cup B$, est le sous-ensemble flou constitué des éléments de X affectés du plus grand des deux degrés d'appartenance μ_A et μ_B :

$$\forall x \in X \quad \mu_{A \cup B} = \max(\mu_A(x), \mu_B(x)) = \mu_A(x) \vee \mu_B(x) \quad (24)$$

Dans cette définition, max et \vee désignent l'opérateur de calcul du maximum de deux valeurs.

E) Complément :

Le complément de A, que l'on note A^c , est le sous ensemble flou de X constitué des éléments x lui appartenant d'autant plus qu'ils appartiennent peu à A :

$$\forall x \in X \quad \mu_{A^c} = 1 - \mu_A(x) \quad (25)$$

F) Produit cartésien :

Le produit cartésien est une méthode de combinaison de sous-ensembles flous définis sur des univers du discours différents. Par exemple, cela permet de définir simplement ce que signifie chaud et humide sur un univers du discours température et hygrométrie.

Soient A_1, A_2, \dots, A_n des sous-ensembles flous définis respectivement dans les univers du discours X_1, X_2, \dots, X_n . Le produit cartésien de A_1, A_2, \dots, A_n , que l'on note $A = A_1 \times A_2 \times \dots \times A_n$, est le sous-ensemble flou défini dans l'univers du discours produit $X = X_1 \times X_2 \times \dots \times X_n$ par la fonction d'appartenance :

$$\forall (x = x_1, x_2, \dots, x_n) \in X \quad \mu_A(x) = \min(\mu_{A_1}(x_1), \mu_{A_2}(x_2), \dots, \mu_{A_n}(x_n)) = \bigwedge_{i=1}^n \mu_{A_i}(x_i) \quad (26)$$

III.6.4. Normes et conormes triangulaires :

Les normes et conormes triangulaires constituent une généralisation des opérations de combinaison de type minimum ou maximum.

t-norme	t-conorme	Négation	Nom
$\text{Min}(x, y)$	$\text{Max}(x, y)$	$1 - x$	Zadeh
$x \cdot y$	$x + y - x \cdot y$	$1 - x$	Probabiliste
$\text{Max}(x + y - 1, 0)$	$\text{Min}(x + y, 1)$	$1 - x$	Lukasiewie
$\frac{xy}{\gamma + (1 - \gamma)(x + y - xy)}$	$\frac{x + y - xy - (1 - \gamma)xy}{1 - (1 - \gamma)xy}$	$1 - x$	Hamacher ($\gamma > 0$)
$\begin{cases} x \text{ si } y = 1 \\ y \text{ si } x = 1 \\ 0 \text{ sinon} \end{cases}$	$\begin{cases} x \text{ si } y = 0 \\ y \text{ si } x = 0 \\ 0 \text{ sinon} \end{cases}$	$1 - x$	Weber

Table.III.1 : Définitions des t-normes et t-conormes les plus utilisées.

III.7. Variables linguistiques

L'ensemble de référence d'un mot du langage naturel s'appelle l'univers du discours.

L'univers du discours d'un mot est un ensemble de termes qui évoquent le même concept mais à degrés différents. Il peut être fini ou non.

Une variable linguistique représente un état dans le système à régler. Chaque variable linguistique est caractérisée par un ensemble tel que : $\{x, T(x), U, G, M\}$ [25][26]

Ou :

x : est le nom de la variable,

$T(x)$: est l'ensemble des valeurs linguistiques que peut prendre x ,

U : est l'univers du discours associé avec la valeur de base,

G : est la règle syntaxique pour générer les valeurs linguistiques de x ,

M : est la règle sémantique pour associer un sens à chaque valeur linguistique.

Exemple

Prenons comme exemple la variable linguistique température ou :

- L'univers du discours U : $U = [0^0 \text{C}, 70^0 \text{C}]$.
- Variable linguistique x : La température
- Valeurs linguistiques $T(x)$: « Faible » « Moyenne » « Elevée »
- Règle syntaxique G : Si $20 < T < 40$ alors température moyenne

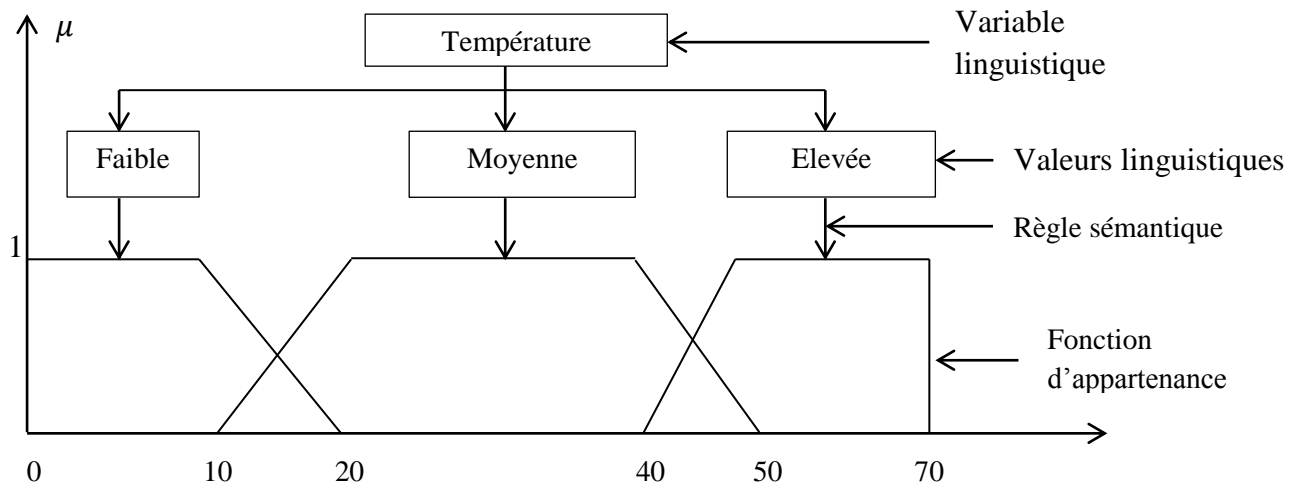


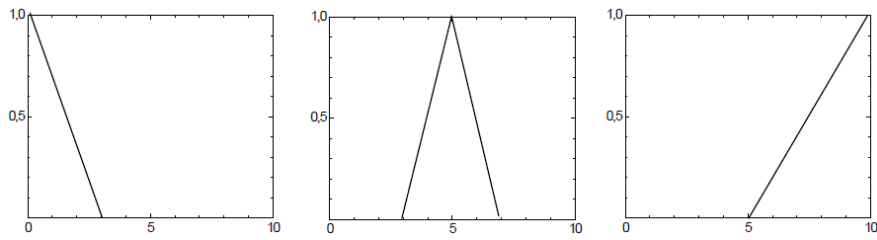
Figure.III.4. Variable linguistique $\{x, T(x), U, G, M\}$ pour décrire la température

III.8. Fonctions d'appartenance

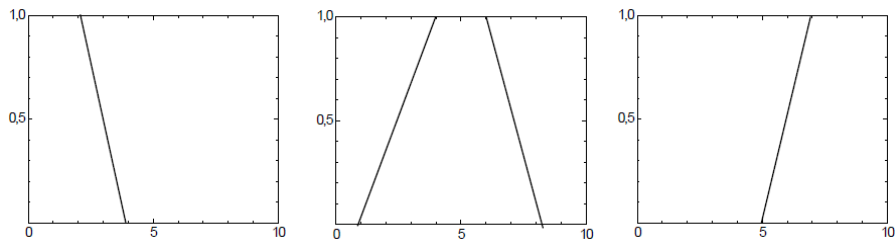
La fonction d'appartenance donne une pondération ou un coefficient de confiance à une variable qualifiée d'incertaine, caractérisant son appartenance à un ensemble de variables linguistiques donne. Elle est désignée par $\mu_E(x)$. L'argument x se rapporte à la variable incertaine, tandis que l'indice E indique l'ensemble de variables linguistiques concerné.

On utilise plusieurs formes pour la description de la fonction d'appartenance. Le choix et la caractérisation de la fonction d'appartenance sont généralement fondés sur l'expérience et le domaine d'application.

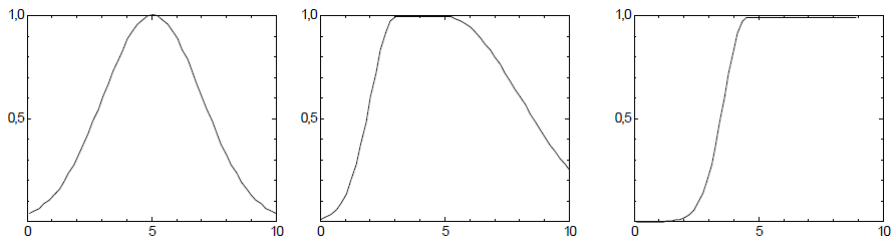
La fonction d'appartenance peut avoir plusieurs formes : triangulaire, trapézoïdale, Gaussienne ou sigmoïde.



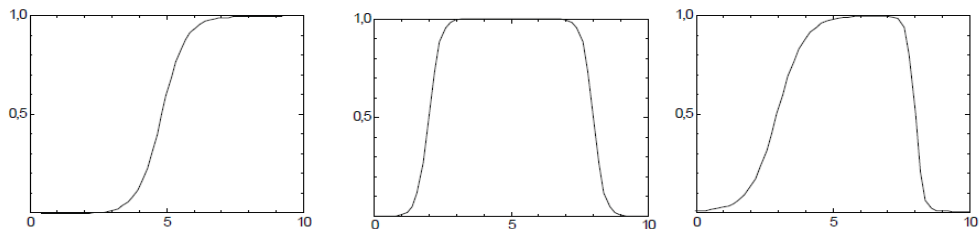
(a) Fonctions d'appartenance triangulaires



(b) Fonctions d'appartenance trapézoïdales



(c) Fonctions d'appartenance Gaussiennes



(d) Fonctions d'appartenance sigmoïdes

Figure.III.5. Différentes formes de fonctions d'appartenance

III.9. Structure générale d'un système flou

Le système à base de la logique floue est composé de quatre blocs principaux (Fig. III.6).

- Fuzzificateur
- Base de connaissances floues,
- Engin d'inférence floue,
- Défuzzificateur.

Chacun de ces blocs fera l'objet d'un développement détaillé.

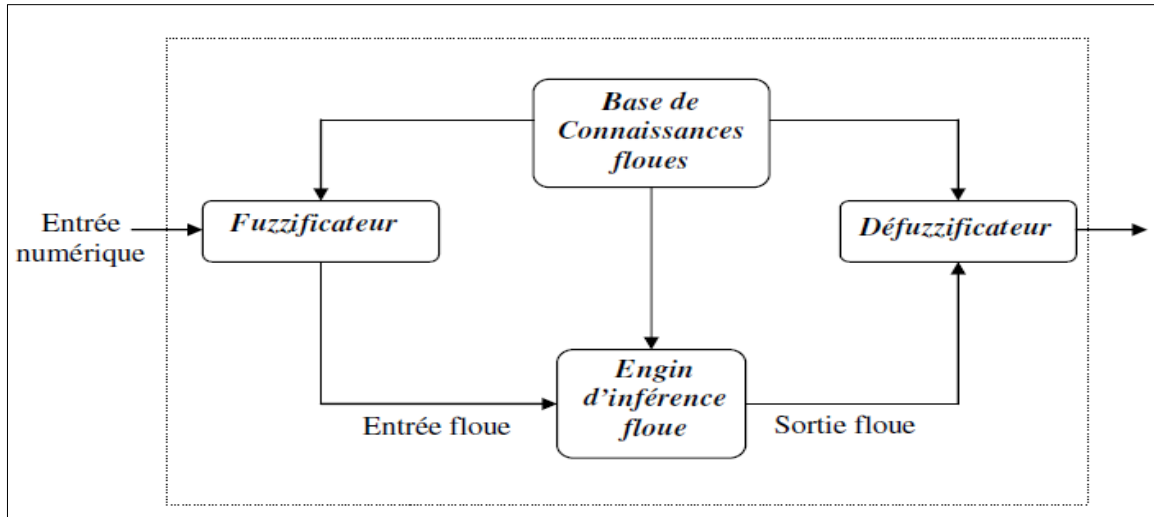


Figure.III.6. Structure générale d'un système basé sur la logique floue

III.9.1. Fuzzificateur

Le fuzzificateur effectue la conversion de l'entrée numérique observée $U_0 = (U_{01}, U_{02}, \dots, U_{0n})^T \in U$ en un ensemble flou $F_x = F_{x0} \times F_{x2} \times \dots \times F_{xn}$ définie dans U .

Cette opération est nécessaire, car dans les applications de la logique floue, les données observées sont souvent numériques, mais dans le système flou la manipulation des données est basée sur la théorie des ensembles flous [24].

Il y'a au minimum deux choix pour cette conversion :

a) Fuzzification singleton

Où l'opérateur de fuzzification converti l'entrée numérique $u_0 \in U$ en un singleton flou F_x dans U tel que :

$$\mu_{F_x}(u) = 1 \quad \text{Si } u = u_0 \quad (27)$$

$$\mu_{F_x}(u) = 0 \quad \text{Si } u \neq u_0 \quad (28)$$

Cette stratégie est largement utilisée dans les applications de contrôle flou, car elle est facile à implémenter.

b) Fuzzification non – singleton

Est une fuzzification pour laquelle $\mu_{F_x}(u)$ est égal à l'unité si $u = u_0$ et décroît quand on s'éloigne de u_0 . Par exemple,

$$\mu_{F_x}(u) = \exp(-(u - u_0)^T \times (u - u_0) / \sigma^2) \quad (28)$$

Où σ est un paramètre caractérisant la forme de μ_{F_x} .

III.9.2. Règles d'Inférence floue

L'idée principale des systèmes basés sur la logique floue, est d'exprimer la connaissance humaine sous la forme de règles linguistiques de forme Si...alors ...Chaque règle a deux parties [23, 27, 28]:

- partie antécédente (prémisse ou condition), exprimée par Si...,
- partie conséquente (conclusion) exprimée par alors.

La partie antécédente est la description de l'état du système. La partie conséquente exprime l'action que l'opérateur qui contrôle le système doit exécuter. Chaque règle floue est basée sur l'implication floue.

Il y a plusieurs formes de Si ...alors ...la forme générale est :

Si (un ensemble de conditions est satisfait) alors (un ensemble de conséquences peut être exécuté).

Il y a plusieurs sortes d'inférences floues parmi lesquels :

- Max-prod méthode
- Min-max méthode
- Méthode de Tsukamoto
- Méthode de Takagi et sugeno
- Méthode somme –prod

III.9.3. Defuzzification

Les méthodes d'inférence fournissent un résultat qui est une fonction d'appartenance. Or, la sortie du contrôleur est en général une grandeur continue, prenant sa valeur dans un intervalle. La défuzzification est le traitement qui permet de définir une correspondance entre le résultat de l'inférence et la grandeur continue fournie en sortie.

a) Défuzzification par centre de gravité :

La défuzzification par centre de gravité consiste à calculer l'abscisse du centre de gravité de la fonction d'appartenance selon :

$$y_{cg} = \frac{\int_y y \cdot \mu_{B_{res}}(y) dy}{\int_y \mu_{B_{res}}(y) dy} \quad (28)$$

En pratique, on estime le centre de gravité en calculant la moyenne d'un certain nombre de points échantillonnés sur la fonction :

$$y_{cg} = \frac{\sum y_i \cdot \mu_{B_{res}}(y_i)}{\sum \mu_{B_{res}}(y_i)} \quad (29)$$

Le temps nécessaire au traitement est directement proportionnel au nombre de points retenus pour le calcul de la moyenne. Selon les contraintes fixées par l'application, il y a un compromis à réaliser entre la précision souhaitée et le temps de calcul disponible.

b) Défuzzification par centre maximum :

Dans cette méthode, la valeur de sortie est estimée par l'abscisse du point correspondant au centre de l'intervalle pour lequel la fonction d'appartenance est maximale. Cette valeur est fournie par l'expression :

$$y_{cm} = \frac{\inf M + \sup M}{2} \quad (30)$$

Dans laquelle M est l'ensemble des points pour lesquels la fonction d'appartenance est maximale :

$$M = \{y \in [-c, c] | \mu_{B_{res}}(y) = H(B_{res})\} \quad (31)$$

Dans le cas discret, on explore en fait la liste de tous les points pour lesquels la fonction d'appartenance est maximale afin de trouver le plus petit et le plus grand.

c) Défuzzification par valeur maximum :

Cette méthode ne s'utilise que dans le cas discret. On choisit comme sortie y_m l'abscisse de la valeur maximale de la fonction d'appartenance résultante $\mu_{B_{res}}(y)$. Lorsque $\mu_{B_{res}}(y)$ est écrêtée, on prend la moyenne des abscisses du maximum :

$$y_m = \frac{\sum_{y_i \in M} y_i}{|M|} \quad (32)$$

Où M est défini dans l'équation (31).

III.10. Avantages et inconvénients de réglage par logique floue

III.10.1. Les Avantages

Les avantages principaux des régulateurs flous sont les suivants :

- ❖ L'incorporation directe des informations floues et linguistiques, provenant d'un expert humain, dans le système flou.
- ❖ Il n'y a pas nécessaire de faire un modèle mathématique du système à régler.
- ❖ Le système flou est une approximation universel, c'est-à-dire, il est suffisamment générale pour générer n'importe quelle action.
- ❖ La logique floue est facile à comprendre par ceux qui ne sont pas des spécialistes, car elle imite la stratégie du raisonnement humain.
- ❖ On peut maîtriser les systèmes non linéaires et difficiles à modéliser.

III.10.2. Les Inconvénients

- ❖ Manque de directives précises pour la conception d'un régulateur,
- ❖ Précision de réglage en général peu élevée.

III.11. Conclusion

Nous avons vu dans ce chapitre l'intérêt de la logique floue dans le domaine du contrôle de processus. Cette approche permet de tenir compte à la fois des connaissances d'un expert humain et de l'incertitude et de l'imprécision des données traitées par le contrôleur.

Les variables linguistiques permettent de traiter ces deux informations initialement très différentes à l'aide d'un formalisme unique.

Dans certains cas, une approche globale de la conception est toutefois possible. Elle est basée sur une méthode d'apprentissage, dans laquelle un système extérieur au contrôleur analyse les performances de ce dernier lorsqu'il utilise un ensemble donné de paramètres. Par essais successifs, le système extérieur peut sélectionner le jeu de paramètres qui assurera le «meilleur» fonctionnement du contrôleur.

CHAPITRE IV

LE PERT FLOU

CHAPITRE IV

LE PERT FLOU

IV.1. Introduction

L'utilisation d'ensembles flous permet donc de représenter une variable par ses valeurs possibles à différents degrés. Cette capacité de représentation des ensembles flous permet d'associer à ces ensembles des symétries différents.

Cette diversité de «sens» associée à des ensembles flous suscitent l'intérêt des chercheurs dans la recherche opérationnelle.

En fait, depuis les années 70, nombre de travaux se sont intéressés aux problèmes de plus court chemin puis de CPM et PERT avec des données floues. Dans le reste de ce chapitre, nous verrons des rappels, notamment sur les notions d'arithmétique floue et sur leur manipulation à l'aide de l'arithmétique, et enfin, nous avons étudié la méthode PERT floue.

IV.2. Notions d'arithmétique floue

IV.2.1. Ensembles flous

Un ensemble flou (Fuzzy) \bar{A} est un sous-ensemble d'un ensemble de référence Ω , dont les limites sont progressives plutôt qu'abruptes. Plus mathématiquement, \bar{A} est défini par sa fonction d'appartenance (membership function) $\bar{A}(x)$ dont les valeurs sont dans l'intervalle $[0, 1]$ pour tout x appartenant à Ω .

Cette fonction d'appartenance attribue à chaque x un degré d'appartenance.

Par exemple : -si $\bar{A}(x) = 1$, x appartient à \bar{A}

- si $\bar{A}(x) = 0$ x n'appartient pas à \bar{A} ;

- si $\bar{A}(x) = 0,4$, nous disons que l'appartenance de x dans \bar{A} est de 0,4.

Notons que dans la littérature, les notations utilisées pour ces différentes entités mathématiques diffèrent d'un auteur à l'autre, la notation μ_A remplaçant souvent $\bar{A}(x)$ pour la fonction d'appartenance, comme dans [29] ou [30]. Nous adoptons ici les notations de [31].

IV.2.2. Nombres flous

Bien qu'un "nombre flou" puisse avoir des formes très variées, dans ce travail, nous ne considérerons que des nombres flous triangulaires ou trapézoïdaux, qui sont les plus rencontrés (et utilisés) dans la littérature.

Un nombre flou triangulaire est défini par trois nombres $a < b < c$, la base du triangle étant l'intervalle $[a, c]$ et le sommet ayant l'abscisse b , comme à la Figure IV.1. Il peut s'écrire

$\bar{N} = (a/b/c)$. Un nombre flou trapézoïdal est défini de la même manière par quatre nombres $a < b < c < d$ comme à la Figure IV.2 et peut s'écrire $\bar{N} = (a/b, c/d)$. Sur ces graphiques, l'axe des abscisses peut représenter une durée et l'axe des ordonnées donne le degré d'appartenance (toujours entre 0 et 1).

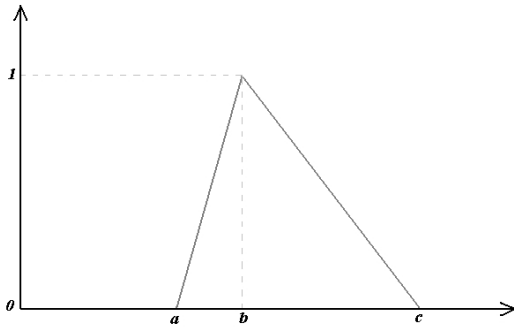


Figure.IV.1. Représentation d'un nombre flou triangulaire

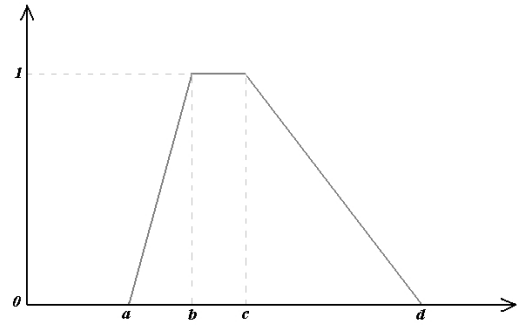


Figure.IV.2. Représentation d'un nombre flou trapézoïdal

Le nombre flou triangulaire peut être vu comme un cas particulier du nombre flou trapézoïdal.

Notons que lors d'opérations arithmétiques sur des nombres flous triangulaires ou trapézoïdaux, le nombre flou résultant ne sera pas nécessairement de la même forme : il pourra être "triangular-like" ou "trapezoidal-like" selon la terminologie anglaise. Nous y reviendrons par la suite.

Au niveau de l'interprétation, nous pouvons remarquer qu'un nombre flou \bar{N} peut être interprété comme un nombre dont la valeur est approximativement N . Cette représentation illustre bien pourquoi la transition entre les valeurs pouvant être N et les valeurs ne le pouvant pas est progressive plutôt qu'abrupte [32].

IV.2.3. Alpha-Cuts

Nous pouvons caractériser un nombre flou trapézoïdal (ou triangulaire) par un intervalle de confiance à un certain niveau α . Il s'agit d'un concept utile pour d'écrire différents groupes de valeurs possibles en faisant une "coupe" dans un sous-ensemble flou. Si \bar{A} est un sous-ensemble flou d'un ensemble, alors un α -cut de \bar{A} , noté $\bar{A}[\alpha]$ est défini comme

$$\bar{A}[\alpha] = \{x \in \Omega \mid \bar{A}(x) \geq \alpha\} \quad (33)$$

Pour tout α tel que $0 < \alpha \leq 1$. Il s'agit donc de tranches horizontales dans un ensemble flou qui produisent un ensemble non flou (un intervalle). Pour $\alpha = 0$, $\bar{A}[0]$ est le support (ou la base) de \bar{A} . Par exemple, à la Figure IV.3, nous avons $\bar{A}[0.25] = [2.8, 8]$ et $\bar{A}[0] = [2, 10]$.

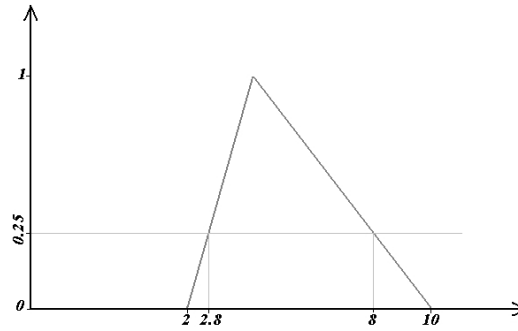


Figure.IV.3.Alpha-cut

IV.2.4.Principe d’extension

Nous allons voir qu’il y a deux méthodes de base permettant d’additionner, soustraire, multiplier ou diviser deux nombres flous \bar{A} et \bar{B} . Commençons par le principe d’extension.

Par définition,

- Si $\bar{C} = \bar{A} + \bar{B}$, alors la fonction d’appartenance pour \bar{C} est

$$\bar{C}(z) = \sup_{x,y} \{ \min(\bar{A}(x), \bar{B}(y)) | x + y = z \} \quad (34)$$

- Si $\bar{C} = \bar{A} - \bar{B}$, alors

$$\bar{C}(z) = \sup_{x,y} \{ \min(\bar{A}(x), \bar{B}(y)) | x - y = z \} \quad (35)$$

- Si $\bar{C} = \bar{A} \times \bar{B}$, alors

$$\bar{C}(z) = \sup_{x,y} \{ \min(\bar{A}(x), \bar{B}(y)) | x \times y = z \} \quad (36)$$

- Si $\bar{C} = \bar{A}/\bar{B}$, alors

$$\bar{C}(z) = \sup_{x,y} \{ \min(\bar{A}(x), \bar{B}(y)) | x / y = z \} \quad (37)$$

En supposant que zéro n’appartienne pas au support de \bar{B} . Dans tous ces cas, C’est aussi un nombre flou. Si \bar{A} et \bar{B} sont triangulaires (trapézoïdaux), alors $\bar{A} + \bar{B}$ et $\bar{A} - \bar{B}$ également, mais $\bar{A} \times \bar{B}$ et \bar{A}/\bar{B} seront “triangular-like” (“trapezoidal-like”), ce qui sera illustré dans la sous-section “Arithmétique floue et alpha-cuts”.

Il est évident que les équations (34) à (37) ne permettent pas de calculer $\bar{A}+\bar{B}$, $\bar{A}-\bar{B}$, etc. de manière simple, ces équations n’étant pas “opérateurs”. Nous pouvons illustrer ce fait par un exemple. Si nous voulons calculer la somme des nombres flous $\bar{A} = (2/3/6)$ et $\bar{B} = (3/4/6)$, nous devons pour chaque z pouvant être obtenu comme une somme possible de x et y, calculer C(z). Par exemple, pour z = 9, il y a une infinité d’alternatives : z = 3 + 6, z = 4 + 5, z = 4, 93 + 4, 07, etc. Pour chaque alternative, il faut prendre le minimum entre $\bar{A}(x)$ et $\bar{B}(y)$. $\bar{C}(z)$ est alors la valeur maximale trouvée parmi toutes les alternatives. Dans notre exemple, nous obtenons $\bar{C}(9) = 0,6$ pour x = 4, 2 et y = 4, 8.

Nous allons présenter une autre procédure basée sur la notion d'alpha-cuts et sur l'arithmétique d'intervalle.

IV.2.5. Théorie des possibilités

La théorie des possibilités a été introduite en 1978 par Lotfi Zadeh [33] afin de prendre en compte la représentation de la notion d'incertitude sur la véracité d'une affirmation, que les données manipulées dans cette affirmation soient précises ou non. Si la théorie des sous-ensembles flous permet de traiter les termes « a peu près » ou « environ », la théorie des possibilités intégrée, quant à elle, la notion de variable linguistiques qui gèrent des notions comme « très », « beaucoup », « peu », « plutôt ».

D.Dubois et H.Prade [32] soulignent le fait que la théorie des probabilités ne permet pas de représenter la différence entre deux états de connaissance se distinguant uniquement par une différence de confiance dans les informations disponibles. Ainsi, D.Dubois et H.Prade introduisent l'outil des mesures de possibilité comme outil de représentation de l'incertain plus général permettant de représenter l'ignorance et de prendre en compte la pertinence d'une information incertaine.

Soit \bar{X} une variable floue dont les valeurs sont restreintes par une distribution de possibilité \bar{A} , où \bar{A} est un sous-ensemble flou de \mathbb{R} . Les distributions de possibilités doivent être normalisées, donc $\bar{A}(x) = 1$ pour certains x de \mathbb{R} .

Si E est un sous-ensemble non flou de \mathbb{R} , la possibilité que X ait ses valeurs dans E se calcule de la manière suivante [31] :

$$Poss[\bar{X} \in E] = \sup\{\bar{A}(x) | x \in E\} \quad (38)$$

Ceci est analogue à la théorie des probabilités, en utilisant le supremum dans la théorie des ensembles flous à la place de la sommation pour les probabilités (et le minimum à la place des multiplications). En effet, si X est une variable aléatoire de densité de probabilité $f(x)$, la probabilité que \bar{X} ait ses valeurs dans E est

$$Prob[\bar{X} \in E] = \int_E f(x)dx \quad (39)$$

Considérons maintenant des variables floues $\bar{X}_1, \bar{X}_2, \dots, \bar{X}_n$ de distributions de possibilité associées $\bar{A}_1, \bar{A}_2, \dots, \bar{A}_n$ respectivement. Soit $\bar{X} = (\bar{X}_1, \dots, \bar{X}_n)$ et $x = (x_1, \dots, x_n) \in \mathbb{R}^n$. En supposant que les variables floues sont non interactives (ce qui est analogue à l'indépendance dans la théorie des probabilités) nous pouvons former leur distribution de possibilité jointe :

$$\pi(x_1, \dots, x_n) = \min\{\bar{A}_1(x_1), \dots, \bar{A}_n(x_n)\} \quad (40)$$

Avec $x \in \mathbb{R}^n$. Pour tous ensemble non flou $E \subset \mathbb{R}^n$, nous avons

$$Poss[\bar{X} \in E] = \sup\{\bar{A}(x) | x \in E\} \quad (41)$$

IV.3. Le PERT flou

Maintenant que nous avons passé en revue les notions de base d'arithmétique floue, nous allons pouvoir étudier le problème du PERT flou.

Comme nous l'avons vu pour le PERT stochastique, en général, les durées des activités d'un projet sont incertaines, celles-ci étant considérées comme des variables aléatoires. Pour le PERT flou, ces mêmes durées sont considérées comme des variables floues. Soit T_{ij} la durée de la tâche (i, j) et T_i le moment auquel toutes les tâches aboutissant au nœud i se terminent. Si le nœud de fin d'un projet est noté n, dans le PERT stochastique, T_n sera une variable aléatoire et nous voudrions déterminer sa fonction densité de probabilité, pour calculer $\text{Prob}[T_n \leq \lambda]$ pour différentes échéances temporelles λ . Or dans le PERT probabiliste, il est très difficile de trouver cette densité de probabilité, comme on l'a souligné dans la section consacrée à cette méthode. Dans le PERT flou, T_n est une variable floue et nous voulons trouver sa distribution de possibilité, pour avoir $\text{Poss}[T_n \leq \lambda]$ pour différentes valeurs de λ . Pour cela, nous allons supposer que les durées des tâches sont non interactives (ce sera notre seule hypothèse) et représenter les durées par des nombres flous trapézoïdaux.

Soit $M_{ij} = (a_{ij} / m_{ij}^1, m_{ij}^2 / b_{ij})$ la distribution de possibilité pour T_{ij} . La possibilité que $T_{ij} = x$ est donc égale à $M_{ij}(x)$, ce que nous notons $\text{Poss}[M_{ij} = x] = M_{ij}(x)$. Dans le nombre flou trapézoïdal M_{ij} , a_{ij} est la durée la plus optimiste, b_{ij} la durée la plus pessimiste, m_{ij}^1 et m_{ij}^2 les bornes inférieure et supérieure de la durée la plus plausible. Nous avons donc $0 \leq a_{ij} < m_{ij}^1 \leq m_{ij}^2 < b_{ij}$, ce qui permet de traiter des nombres flous triangulaires (si $m_{ij}^1 = m_{ij}^2$).

Pour déterminer la distribution de possibilité de la durée d'un projet, il faut tout d'abord former la distribution de possibilité jointe pour toutes les tâches. Soit $\langle T_{ij} \rangle$ un vecteur de durées possibles pour toutes les tâches du projet. La distribution jointe $\mu(\langle T_{ij} \rangle)$ est définie, pour des variables floues non interactives T_{ij} par :

$$\mu(\langle T_{ij} \rangle) = \min\{M_{ij}(T_{ij}) \mid \text{pour toutes les activités}\} \quad (42)$$

Si $a_{ij} < T_{ij} < b_{ij}$ pour toutes les activités.

Si un T_{ij} n'est pas dans cet intervalle, alors $\mu(\langle T_{ij} \rangle) = 0$ [18].

$$\Gamma_x = \{\langle T_{ij} \rangle \mid T_n = x\} \quad (43)$$

Soit l'ensemble de tous les vecteurs de durées d'activités qui produisent une durée totale de projet égale à x . Alors,

$$\text{Poss}[T_n = x] = \sup\{\mu(\langle T_{ij} \rangle) \mid \langle T_{ij} \rangle \in \Gamma_x\} \quad (44)$$

car $\mu(\langle T_{ij} \rangle)$ est une distribution de possibilité. La distribution de possibilité pour T_n est un ensemble flou M_n tel que $\text{Poss}[T_n = x] = M_n(x)$, où M_n est construit à partir de l'équation (44).

Nous allons maintenant voir comment calculer M_n en pratique. Soient

$$L_{ij}(y) = (m_{ij}^1 - a_{ij})y + a_{ij} \quad (45)$$

$$R_{ij}(y) = (m_{ij}^2 - b_{ij})y + b_{ij} \quad (46)$$

Pour $0 \leq y \leq 1$. Il suffit dès lors, pour les parties gauche (L) et droite (R) des intervalles flous, de procéder à de simples additions et applications de l'opérateur max à chaque nœud du graphe et pour chacune des valeurs choisies de y .

IV.3.1.Exemple

Nous allons appliquer les notions vues ci-dessus au graphe de la Figure IV.4. Les estimations des durées des activités sont reprises au Tableau IV.1.

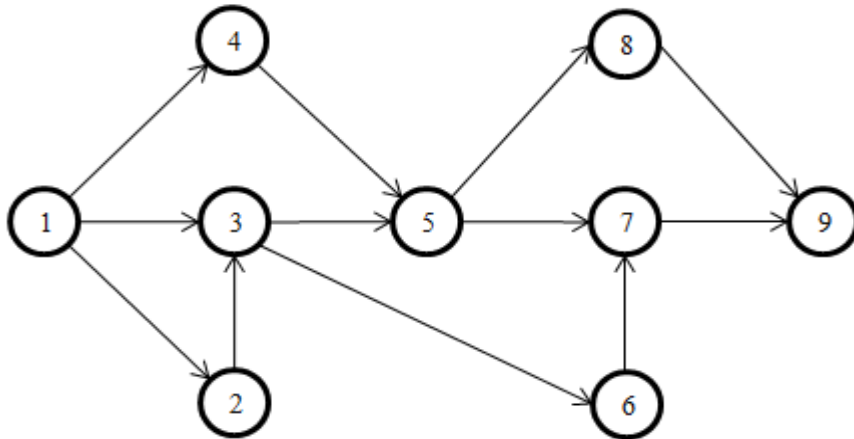


Figure.IV.4.Projet à traiter par le PERT flou

Activité	a_{ij}	m_{ij}^1	m_{ij}^2	b_{ij}
(1,2)	5	6	6	7
(1,3)	8	9	9	11
(1,4)	2	3	3	5
(2,3)	2	4	4	7
(3,5)	3	4	5	6
(3,6)	3	5	5	6
(4,5)	2	3	4	6
(5,7)	4	5	5	6
(5,8)	2	4	6	7
(6,7)	3	4	4	8
(7,9)	4	5	5	8
(8,9)	3	4	5	6

Table.IV.1. Estimations des durées associées au projet du graphe

Il consiste à calculer pour différentes valeurs de $y \in [0, 1]$ les fonctions $L_{ij}(y)$ et $R_{ij}(y)$. Pour chaque valeur de y , il suffit dès lors de calculer le temps auquel peut avoir lieu chaque évènement, jusqu'à l'évènement de fin, et ce, pour la partie gauche (L) et la partie droite (R) des nombres flous considérés pour les durées.

Nous avons dessiné le nombre flou résultant pour T9 (date de fin du projet) à la Figure IV.4. Sur cette figure, nous pouvons constater, aussi bien à gauche qu'à droite, un changement de pente rendant le nombre flou dessiné non trapézoïdal. Ces changements de pente correspondent à des changements de chemin critique : dans la partie gauche du nombre flou, le chemin de longueur maximale est $1 - 3 - 5 - 7 - 9$ pour un alpha de moins de 0,5 et $1 - 2 - 3 - 5 - 7 - 9$ pour alpha supérieur à 0,5. Dans la partie droite, pour alpha inférieur à 0,6, le chemin de durée maximale est $1 - 2 - 3 - 6 - 7 - 9$ tandis qu'au-dessus de cette valeur d'alpha, c'est le chemin $1 - 2 - 3 - 5 - 8 - 9$.

Nous pouvons également remarquer sur cette figure que l'étendue du nombre flou représentant la durée totale du projet est énorme si nous la comparons à celle des durées individuelles des tâches. C'est l'un des défauts du PERT flou : plus nous sommes des durées d'activités floues, plus l'incertitude sur la durée finale est grande. Néanmoins, nous pouvons choisir de ne considérer qu'une partie de l'étendue de ce nombre flou : par exemple, nous pouvons choisir de considérer l'alpha-cut du nombre en question, pour un alpha de 0,75, en nous basant sur l'idée qu'il faut un degré de plausibilité suffisant pour être crédible. Ainsi, nous pourrions comparer les dispersions pour un alpha allant de 0,5 à 1 et retenir la moyenne, soit 0,75 pour alpha. Cela reviendrait, dans notre exemple, à considérer un intervalle de temps de 22,5 à 27,75 environ plutôt que de 19 à 36 unités de temps, tout en gardant à l'esprit que le maximum de possibilité se situe entre 24 et 26 unités de temps.

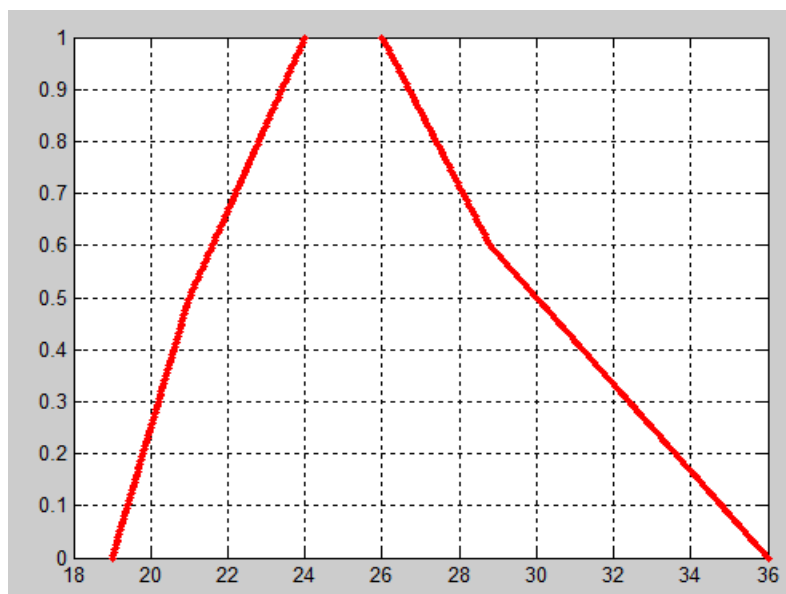


Figure.IV.5. Nombre flou obtenu pour la durée totale du projet

IV.3.2. Difficultés inhérentes au PERT flou

Comme nous avons pu nous en rendre compte au travers de l'exemple ci-dessus, il est simple de calculer les temps flous de début au plus tôt des activités d'un graphe en utilisant la méthode du chemin critique. Cependant, déterminer des dates de début au plus tard et des marges de temps pose beaucoup plus de problèmes. Jusqu'en 2003, ces grandeurs ne pouvaient d'ailleurs pas être calculées de manière satisfaisante, ce qui en fait un problème récent.

Le lecteur intéressé peut se référer à [34], cet article étant à la base du traitement rigoureux de ce problème.

La difficulté essentielle provient du fait que la soustraction en arithmétique floue n'est pas l'inverse de l'addition [35]. De ce fait, nous pouvons utiliser des relations de récurrence directe, mais pas de la méthode du chemin critique.

De plus, pour calculer la marge de temps disponible sur une activité, nous avons besoin de connaître les dates de début ou de fin au plus tard des activités (qui se calculent difficilement). Une activité est considérée comme critique dans la méthode CPM lorsque sa date de fin au plus tard moins sa date de fin au plus tôt est nulle. Cette notion n'a que peu de sens lorsque les durées des tâches sont imprécises, la notion de tâche critique devenant également incertaine.

Nous ne traiterons pas ce problème (qui à lui seul pourrait faire l'objet d'un mémoire de fin d'études) dans le cadre de ce travail, où nous nous concentrons essentiellement sur la durée totale la plus probable (ou possible) d'un projet.

IV.4. Conclusion

La logique floue n'est pas vraiment une méthode de résolution, mais représente un outil et concept performant pour prendre en considération l'ensemble des données et des paramètres incertains du problème abordé.

La logique floue présente, grâce au concept de fonction d'appartenance, un bon compromis entre la flexibilité du travail et la puissance de la représentation. Il permet donc de modéliser l'incertitude et l'imprécision.

Les partisans de l'ordonnancement flou justifient l'utilisation du nombre flou par le fait que les distributions de probabilités des durées des tâches nécessitent un certain historique. Comme son nom l'indique, cette méthode utilise la théorie de sous-ensemble floue utilisée dans l'intelligence artificielle.

CHAPITRE V

LA COMPARAISON ENTRE LES TROIS METHODES

CHAPITRE V

LA COMPARAISON ENTRE LES TROIS METHODES

V.1. Introduction

Maintenant que les trois méthodes auxquelles nous nous intéressons ont été exposées théoriquement, nous pouvons en entamer une comparaison. Notons avant tout qu'une comparaison peut porter sur des points très variés, comme par exemple :

- ❖ Les trois méthodes mènent-elles au même chemin de durée maximale ?
- ❖ Quelle est la meilleure méthode ?
- ❖ ...

Dans ce qui suit, nous allons tout d'abord nous focaliser sur le premier point, qui demande l'implémentation de ces trois méthodes. D'après les résultats obtenus nous réponds au deuxième point.

V.2. Les outils de développement

V.2.1. Système d'exploitation

Pour que notre travail atteigne l'objectif qu'on visite, on a pris l'initiative d'exploiter et d'implémenter nos algorithmes sur la version : Windows 7 sur un PC de type i3, 1.7GHz, 2.00 Go de RAM.

Ce choix se traduit par l'efficacité de ce l'environnement en ce qui concerne la structure d'interaction événementielle qu'elle dispose pour communiquer avec de applications actives, ainsi que les ressources de la machine qu'il offre aux différentes applications, enfin, son système d'allocation de mémoire qui est un des meilleurs présents dans ce domaine.

V.2.2. Delphi 2007

Pour l'implémentation de notre application, nous avons opté pour le langage Delphi 7.

Delphi est à la fois un langage de programmation orienté objet et un environnement de développement intégré (EDI).

L'EDI Delphi est un EDI propriétaire fonctionnant sous Windows créé en 1995 par Borland. À l'époque, créer des programmes graphiques sous Windows se faisait en grande majorité en utilisant soit la chaîne de compilation Visual C++, soit le RAD Visual Basic. Le premier outil étant excessivement complexe et le second assez peu structuré, Delphi apparut alors comme une alternative viable pour beaucoup de développeurs qui souhaitaient créer des programmes pour Windows.

L'environnement de développement s'appuie sur un éditeur d'interface graphique associé à un éditeur de code source. Il doit son succès à sa facilité d'utilisation pour développer des applications graphiques et/ou liées aux bases de données. L'interface de développement permet l'ajout de composants tiers (graphiques ou non) via un système de composants. La figure.V.1 présente l'interface de Delphi7. [36]

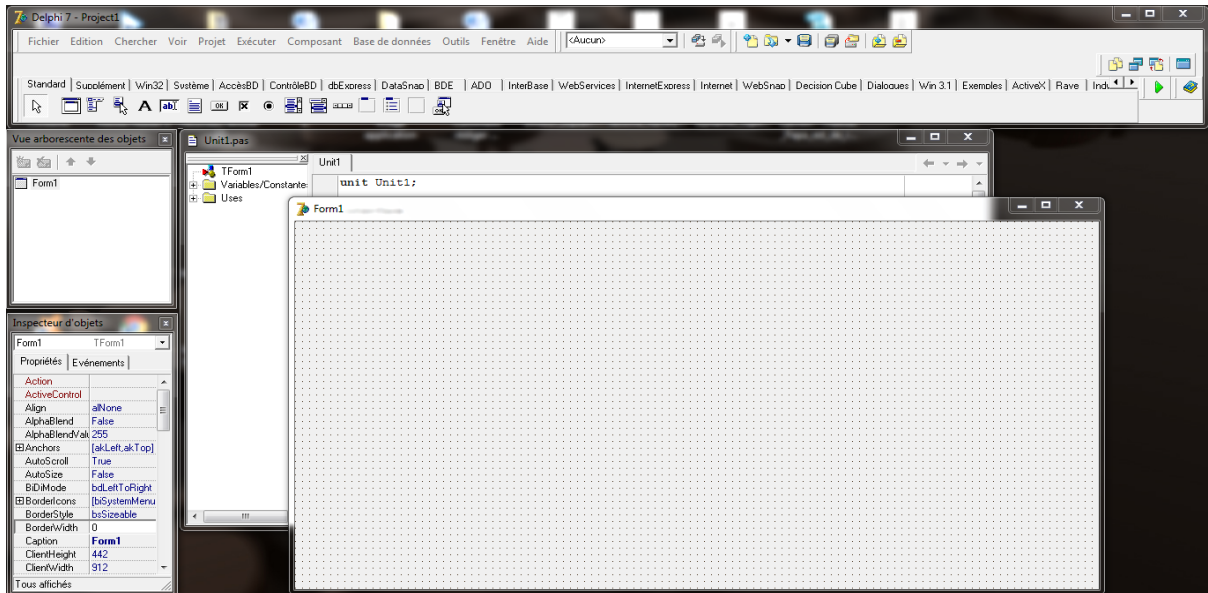


Figure.V.1. l'interface de Delphi 7

V.2. Le projet

Nous allons donc traiter un exemple de la construction d'un nouveau bâtiment (table.II.2).

V.2.1. L'architecture de projet

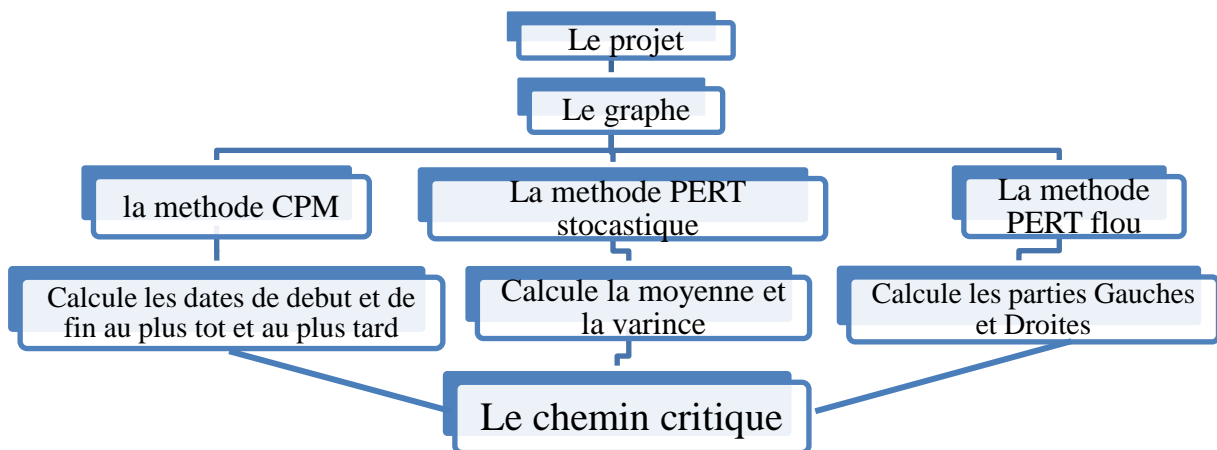


Figure.V.2.Architecture de projet

V.3. Résultats et comparaison

On va appliquer les trois méthodes sur cet exemple pour obtenir le chemin critique

V.3.1. Le chemin critique

➤ Par la méthode CPM

	Debut	1	2	3	4	5	6	7	8	9	10	11	Fin
Debut	1	2	3	6	9	10	11	12					
1													
2													
3													
4													
5													
6													
7													
8													
9													
10													
11													
Fin													

Figure.V.2.Chemin Critique par CPM

➤ Par la méthode PERT stochastique

	Debut	1	2	3	4	5	6	7	8	9	10	11	Fin
Debut	1	2	7	8	9	10	11	12					
1													
2													
3													
4													
5													
6													
7													
8													
9													
10													
11													
Fin													

Figure.V.3.Chemin Critique par PERT stochastique

Avec :

$$K_{12} = \frac{36-36.68}{\sqrt{4.33}} = - 0.3268 \quad (47)$$

Et donc

$$P\{\mu_{12} \leq 36\} \leq 32.68 \% \quad (48)$$

➤ **Par la méthode PERT flou**

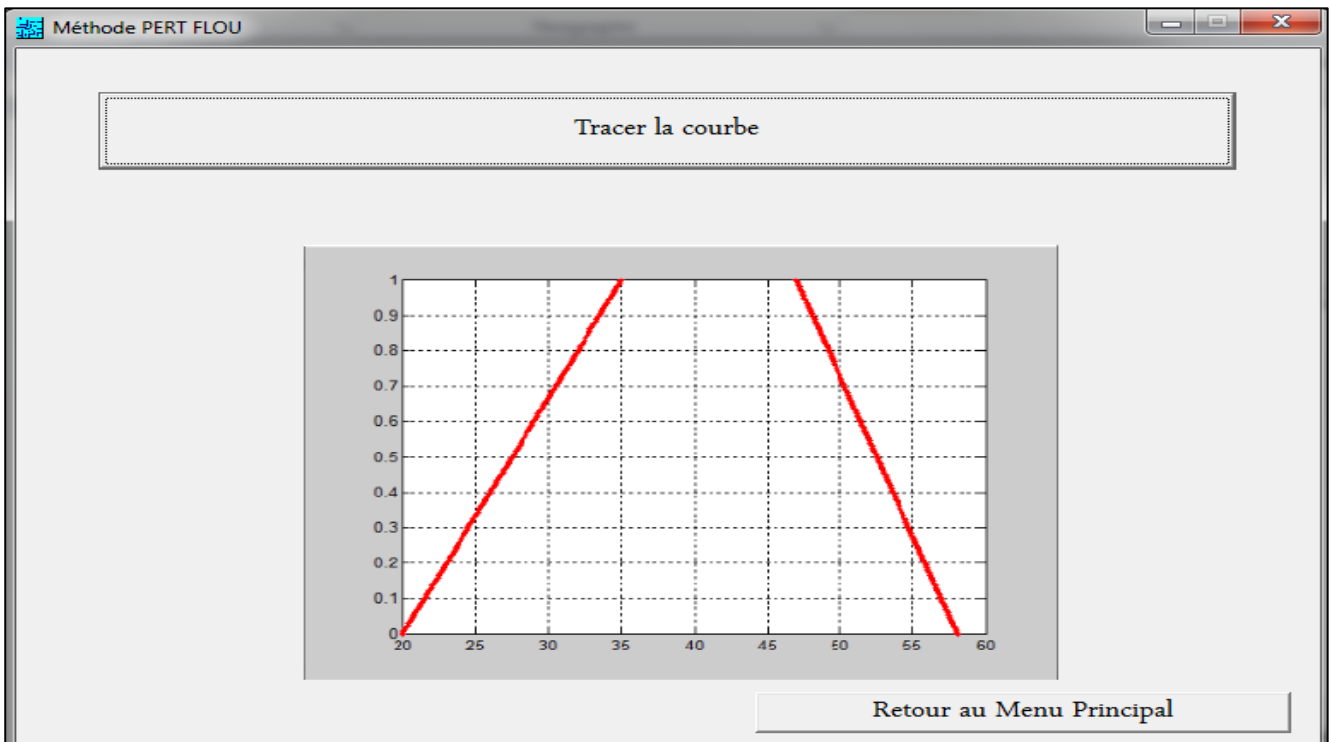


Figure.V.4.Chemin Critique par PERT flou

Nous avons dessiné le nombre flou résultant pour T12 (date de fin du projet) à la Figure V.4. Sur cette figure, nous pouvons constater, aussi bien à gauche qu'à droite, un changement de pente rendant le nombre flou dessiné non trapézoïdal. Ces changements de pente correspondent à des changements de chemin critique : dans la partie gauche du nombre flou, le chemin de longueur maximale est 1 – 2 – 3 – 6 – 9 – 10 – 11 – 12 pour un alpha égale à 0 jusqu'à 0.9 et pour un alpha égale à 1 en remarque deux chemin de longueur maximale : 1 – 2 – 3 – 6 – 9 – 10 – 11 – 12 et 1 – 2 – 7 – 8 – 9 – 10 – 11 – 12. Dans la partie droite, pour égale à 0.1 jusqu'à 0.9, le chemin de durée maximale est 1 – 2 – 7 – 8 – 9 – 10 – 11 – 12 et pour un alpha égale à 0 en remarque aussi deux chemin de longueur maximale : 1 – 2 – 3 – 6 – 9 – 10 – 11 – 12 et 1 – 2 – 7 – 8 – 9 – 10 – 11 – 12.

V.3.2. La comparaison

La méthode CPM donne des durées statiques mais la méthode PERT donne des durées plus ou moins proches de la réalité. Alors la méthode PERT est meilleure que la méthode CPM parce que l'estimation des durées des tâches est plus juste.

Par beaucoup d'aspects, comme la fidélité à la réalité sachant qu'une estimation de durées est vague et non probabiliste, la commodité d'utilisation et la qualité des résultats, le modèle du PERT flou est plus proche de la réalité et plus efficace et plus simple à utiliser qu'un modèle stochastique parce qu'il ne nécessitant aucun calcul de moyenne ou variance.

V.4. Quelques interfaces de l'application

Nous allons définir les différentes interfaces existant dans l'application.

V.4.1. Fenêtre principale

Contient 2 boutons :

- Le bouton Entrer : pour avoir la fenêtre de travail.
- Le bouton Sortir : pour sortir de l'application.

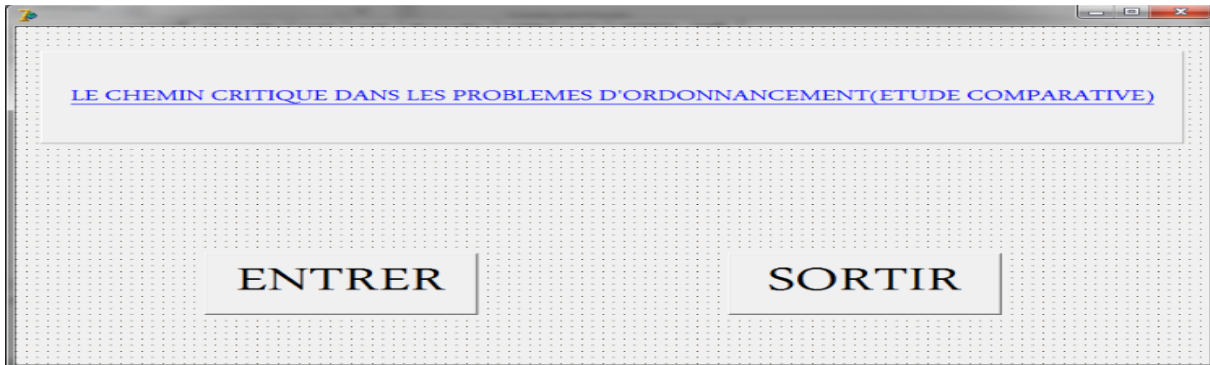


Figure.V.5. Fenêtre principale

V.4.2. Fenêtre de travail

Elle contient le menu suivant :

- Les méthodes** : contient les trois méthodes CPM, PERT stochastique, PERT flou.

En prends comme exemple l'implémentation de la méthode CPM

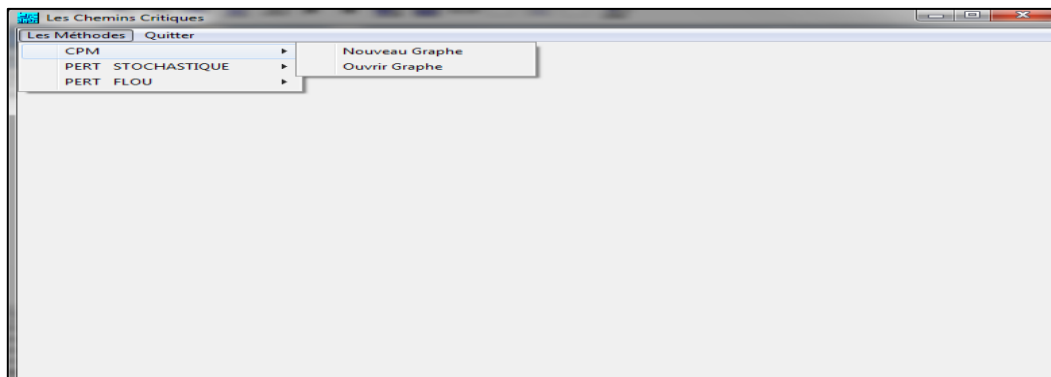


Figure.V.6. Menu des méthodes

- **Nouveau graphe** :

Pour saisir un nouveau graphe on va donner le nom du graphe et le nombre des nœuds.

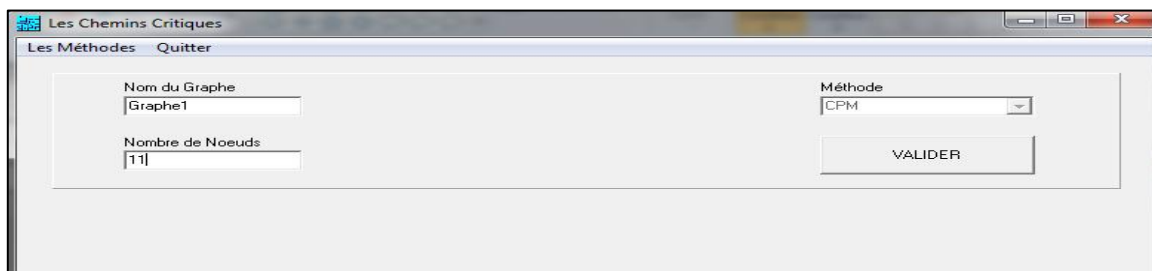


Figure.V.7. Saisir un nouveau graphe

Et quand on fait la validation on obtient la Figure.V.8 et on va remplir la matrice et on clique sur enregistrer pour enregistrer le graphe ou abandonner pour annuler l'enregistrement voir la Figure.V.8.

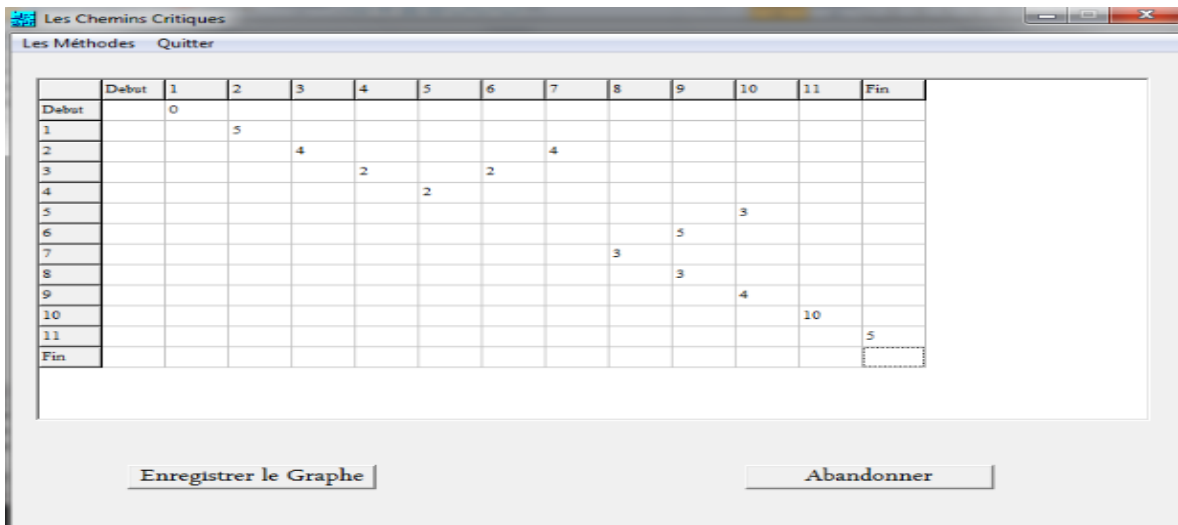


Figure.V.8. Enregistrer ou abandonner le nouveau graphe

- **Ouvrir graphe :**

Il affiche une liste des choix pour les graphes enregistrer et lorsqu'on choisit un graphe on choisir aussi est ce que calcule par étape ou direct.

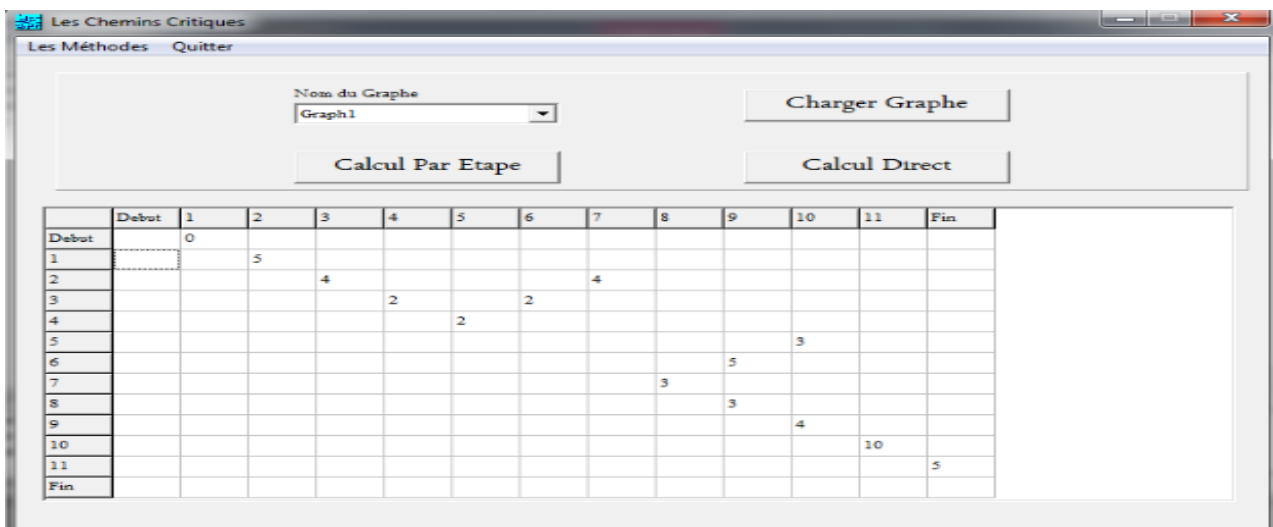


Figure.V.9. Chargement du graphe enregistré

- ❖ **Calcul par étape :**

La figure.V.10 contient les étapes pour calculer le chemin critique.

- Calcul des dates aux plus tôt :

Ce bouton permet de calculer la date au plus tôt de chaque tâche.

- Calcul des dates aux plus tard :

Ce bouton permet de calculer la date au plus tard de chaque tâche.

- Calcul des marges

Ce bouton permet de calculer la marge libre de chaque tâche.

- Afficher tous les chemins

L'affichage de tous les chemins existé dans le graphe.

- Afficher les dates aux plus tôt et plus tard de chaque tâche

Ce bouton permet d'afficher les dates aux plus tôt et plus tard pour chaque tâche.

- Afficher le chemin critique

Ce bouton permet d'afficher les chemins critiques.

- Retour au menu principal :

Ce bouton pour aller à la fenêtre de travail.

The screenshot shows a window titled 'Méthode CPM' with several control buttons at the top: 'Calcul des Dates au plus tôt', 'Calcul des Dates au plus tard', 'Calcul des marges', 'Afficher tous les chemins', 'Afficher les dates plutot et plustard de chaque tache', and 'Afficher le chemin critique'. Below these is a table with 14 columns (Debut, 1-11, Fin) and 14 rows (Debut, 1-11, Fin). The table contains numerical values representing task durations and dependencies. At the bottom right, there is a button labeled 'Retour au Menu Principal'.

	Debut	1	2	3	4	5	6	7	8	9	10	11	Fin
Debut		0											
1			5										
2				9				9					
3					11		11						
4						13							
5											16		
6										16			
7									12				
8										15			
9											20		
10												30	
11													35
Fin													

Figure.V.10. Le calcul par étape

- ❖ **Calcul direct :** contient le résultat final voire la figure.V.2.

- b) **Quitter :** pour sortir de la fenêtre de travail

V.5. Conclusion

En guise de conclusion, nous pouvons remarquer que quelle que soit la forme sous laquelle elle se présente, la méthode PERT peut être d'une grande utilité pour toutes les personnes impliquées dans un projet, car elle pousse systématiquement à analyser les activités et la structure du projet, ce qui ne peut qu'être bénéfique pour terminer dans les temps.

Aucune des méthodes exposées dans ce rapport ne peut être décrite comme la méthode PERT absolue, chacune ayant ses avantages et ses inconvénients.

CONCLUSION GENERALE

CONCLUSION GENERALE

Comme souvent le cas dans un mémoire de fin d'études tout au long de ce travail, il a fallu des choix : le temps étant limité, il n'était pas possible de traiter l'entièreté du sujet, il est très vaste.

Les problèmes de l'ordonnancement sont très variés, ils diffèrent d'un cas à un autre selon la nature des opérations, et les caractéristiques des ressources, les contraintes portant sur les opérations, et les critères à optimiser.

Depuis leur début d'utilisation vers la fin des années 50, PERT/CPM et MPM ont été employés intensivement pour aider les chefs de projet dans la planification, la programmation, et le commandement de leurs projets. Avec le temps, ces deux techniques graduellement ont bénéficié des avantages des uns et des autres.

L'application de PERT/CPM commence en cassant le projet vers le bas en ses différentes activités, en identifiant les prédécesseurs immédiats de chaque activité, et en estimant la durée de chaque activité.

Le PERT flou très peu connue des utilisateurs contrairement à son analogue stochastique pour lequel il existe de nombreux programmes commerciaux, est un axe de recherche très attirant.

Nous avons l'idée que le PERT flou fait appel à moins d'hypothèses de calcul que le PERT stochastique, qu'il est également plus simple à utiliser, ne nécessitant aucun calcul de moyenne ou variance, et que contrairement au PERT stochastique, il permet de tenir compte des changements de chemin de durée maximale en fonction de l'incertitude sur la durée totale. C'est ce dernier point qui le démarque le plus du PERT stochastique.

Le modèle du PERT flou est plus proche de la réalité et plus efficace qu'un modèle stochastique.

Le PERT flou répond aux conditions de planning "tendu", ce qui n'est pas le cas du PERT stochastique qui allonge la durée des tâches. Toutefois, cette méthode du PERT flou ne permet pas (ou pas encore) de déterminer des marges de temps dans tous les cas, ce problème étant particulièrement difficile à résoudre et toujours en cours de recherche.

Perspectives

Il serait dommage de s'arrêter là. De nombreuses pistes, pouvant apporter une véritable aide, restent à explorer. Voici ci-dessous celles qui nous semblent les plus prometteuses.

La méthode PERT-coût permet au chef de projet d'étudier l'effet sur le coût total de changer la durée estimée du projet en diverses valeurs alternatives. Les données importantes pour chaque activité sont le temps et le coût lorsqu'elles sont faites de la manière normale et puis quand elles sont entièrement accélérées. Et le problème de réduction des tâches fictives dans le graphe PERT qui appelé dans la littérature « le graphe PERT minimal ».

BIBLIOGRAPHIE

BIBLIOGRAPHIE

- [1] J.CARLIER, P.CHRETIENNE, Problèmes d'ordonnancement (modélisation / complexité/ algorithmes), Masson, 1988.
- [2] P. Esquirol, P. Lopez, *L'ordonnancement*, Economica, 1999.
- [3] G. FINK, Recherche opérationnelle et réseaux, Lavoisier, Paris, 2002.
- [4] A. HAIT, C. ARTIGUES, P. BAPTISTE, M. TREPANIER, Ordonnancement sous contraintes d'énergie et de ressources humaines. In: 11e congrès de la Société Française de Génie des Procédés, 9-11 Oct 2007, Saint-Etienne, France.
- [5] MOUHOUB NASSER EDDINE, Algorithmes de construction de graphes dans les problèmes d'ordonnancement de projet, DOCTORAT, SETIF, 2011.
- [6] B.L. MACCATHY & J. LIU. Addressing the gap in scheduling research: a review of optimization and heuristic methods in production scheduling. *International Journal of Production Research*, vol. 31, no. 1, pages 59-79, 1993.
- [7] C. ESSWEIN, Un apport de flexibilité séquentielle pour l'ordonnancement robuste, Mémoire de thèse, Université de Tours, décembre 2003.
- [8] R. KUMAR, Evolutionary multiobjective combinatorial optimization. In *GECCO '07: Proceedings of the 2007 GECCO conference companion on Genetic and evolutionary computation*, pages 3366.3390, New York, NY, USA, 2007. ACM.
- [9] L. H. TRUNG, Utilisation d'ordre partiel pour la caractérisation des solutions robustes en ordonnancement, Thèse de Doctorat, Institut national des sciences appliquées de Toulouse, 2005.
- [10] L. A. IKA, L'analyse de la valeur acquise en contexte d'interdépendance des chemins : une analyse à explorer, Université du Québec en Outaouais, ASAC, 2004.
- [11] F.T.S. CHAN, Resource management in project scheduling through simulation, *International Journal of Computer Applications in Technology*, vol. 10, no1 / 2, 81 – 89, 1997.
- [12] K.R MACLEOD & P.F. PETERSEN, Estimating the tradeoff between resource allocation and probability of on – time completion in project management, *Project Management Journal*, vol. 27, n0 1, 26– 33, 1996 .
- [13] P.B ROBINSON, The Performance Measurement Baseline- A statistical view, *Project Management Journal*, vol. 28, no 2, 47 –52, 1997.
- [14] J. IZUCHUKWU, Shortening the critical path, *Mechanical Engineering*, vol. 112, n° 2, 59-60, 1990.
- [15] D. DUVIVIER, Etude de l'hybridation des méta-heuristiques, application à un problème d'ordonnancement de type job shop, thèse de Doctorat, Université de Littoral Côte d'Opale, 2000.
- [16] H. A. Taha. *Operations Research. An Introduction*, 2nd edition. Macmillan Publishing Co. Inc., New York, 1976.
- [17] D.Golenko-Ginzburg. On the Distribution of Activity Time in PERT. *The Journal of the Operational Research Society*, Vol. 39, No. 8, Aug. 1988.

- [18] H. G. Daellenbach, J. A. George, and D. C. McNickle. Introduction to Operations Research Techniques, 2nd edition. MacMillan, New York, 1983.
- [19] Ph. Vincke. Cours de STAT012 : Probabilités et Statistiques. Université Libre de Bruxelles, 2004.
- [20] Zadeh, L. A. Soft computing and fuzzy logic. IEEE Software, 11(6), pp 48-56, 1994
- [21] Mamdani, E.H., Assilian, S. An experiment in linguistic synthesis with a fuzzy logic controller. *Int. J. Man Mach. Studies*, 7(1) : 1–13, 1975.
- [22] J.B.Kiszka, M.M. Gupta, et P.N. Nikiforuk. “ Energetic stability of fuzzy dynamic systems”. IEEE Trans. on Syst.Man Cybern. SMC-15(5): 783-792,1985.
- [23] Chuen Chien, “Fuzzy Logic in Control Systems : Fuzzy Logic Controller, part I” IEEE Transactions on System Mann. And cyri Rsetrs, Vol. 20 No. 2, March/April 1990 pp : 405-417.
- [24]Hassina Megherbi, “Control intelligent d’un bras Manipulateur” mémoire de Magister, université de Sétif 1998.
- [25] Witold Pedrycz, Fernando Gomide. “An introduction to fuzzy sets: Analysis and Design”. Massachusetts, 1998.
- [26] R.BABUŠKA. “Fuzzy and neural control”. DISC Course Lecture Notes Control, Engineering Laboratory of Information Technology and Systems, Delft University of Technology, Delft, the Netherlands, 2001.
- [27] Jelena Godjevac, “Idées nettes sur la logique floue” Press Poly techniques et Universitaires Romandes 1999.
- [28] Arnauld Kaufmann, “Introduction à la logique floue” Technique de l’ingénieur, Traité informatique industrielle pp.1-9.
- [29] D.Dubois, H.Fargier, and P.Fortemps. Fuzzy scheduling: Modelling flexible constraints vs. coping with incomplete knowledge. European Journal of Operational Research 147, 2003.
- [30] S.Chanas and P.Zielinski. Critical path analysis in the network with fuzzy activity times. Fuzzy Sets ans Systems 122, 2001.
- [31] J.J.Buckley, E.Esmali, and T.Feuring. Fuzzy Mathematics in Economics and Engineering. Springer-Verlag, Heidelberg, Germany, 2002.
- [32] D.Dubois and H.Prade. Fuzzy Real Algebra : Some Results. Fuzzy Sets ans Systems 2, North-Holland Publishing Company, 1979.
- [33] L.A.Zadeh, “ Fuzzy sets as a basis for a theory of possibility”, Fuzzy sets and Systems, vol., n^o1, 1978, pp.328.
- [34] D.Dubois, H.Fargier, and V Galvagnon. On latest starting times and floats in activity networks with ill-known durations. European Journal of Operational Research 147, 2003.
- [35] M. Gagnon. COA Modeling with Fuzzy Information. Defence Research and Development Canada, 2002.
- [36] [https://fr.m.wikipedia.org/wiki/Delphi_\(langage\).html](https://fr.m.wikipedia.org/wiki/Delphi_(langage).html) , consulté le : 11/05/2017.

ملخص

نظرية الجدولة هي فرع من البحث العمليتي التي تركز على حساب مواعيد التنفيذ الأمثل للمهام، مع العديد من أساليب الجدولة: مخطط غانت، طريقة PERT/CPM و MPM إدارة المشاريع، و بغض النظر عن طبيعة و حجم المشروع هو أحد القواعد المهمة في مجتمع البحث العمليتي. هذا العمل هو لمحة عامة على جميع طرق الجدولة وتحديدًا طريقة PERT و نحن نهتم باستخدام المنطق الضبابي في سياق المشاكل التي قد تكون ذات صلة بمشروع مشاكل الجدولة. يعرض هذا العمل حساب المسار الحرج بواسطة ثلاث طرق PERT/CPM و PERT العشوائية و PERT الغامضة، والمقارنة بين هذه النتائج.

كلمات مفتاحية: مخطط غانت، طريقة MPM، طريقة PERT/CPM، PERT العشوائية، PERT الضبابي.

Abstract

The Scheduling theory is a branch of operational research which focuses on the calculation of optimal execution dates of tasks, with several scheduling methods: GANTT chart, the MPM method, and PERT / CPM method.

Project management, regardless of the nature and size of a project is a very important operational research community.

This work gives an overview of all the scheduling methods and more precisely the PERT method.

We are interested in the use of fuzzy logic in the context of problems that may be related to project scheduling problems.

This work presents the calculation of the critical path by three methods PERT / CPM, PERT stochastic and PERT fuzzy, and the comparison between these results.

Key words: GANTT chart, the MPM method, PERT / CPM method, PERT stochastic, PERT fuzzy.

Résumé

La théorie de l'ordonnancement est une branche de la recherche opérationnelle qui s'intéresse au calcul de dates d'exécution optimales de tâches, avec plusieurs méthodes d'ordonnancement : diagramme de GANTT, la méthode MPM et la méthode PERT/CPM.

La gestion de projet, quel que soit la nature et taille d'un projet, est une composante très importante de la communauté de recherche opérationnelle.

Ce travail fait un tour d'horizon sur l'ensemble des méthodes d'ordonnancement et plus précisément la méthode PERT.

On s'intéresse à l'utilisation de la logique floue dans le cadre de problèmes pouvant s'apparenter aux problèmes d'ordonnancement de projet.

Ce travail présente le calcul du chemin critique par trois méthodes PERT / CPM, PERT stochastique et PERT floue, et la comparaison entre ces méthodes.

Mots clés : diagramme de GANTT, la méthode MPM, la méthode MPM, PERT stochastique, PERT floue.