Optimisation et complexité

Maher REBAI: Enseignant en informatique à l'école supérieure d'ingénieurs Léonard de Vinci Paris la défense

EFREI, Paris

Année Universitaire : 2019-2020

Chapitres

- Introduction à la RO
- Modélisation par les graphes
- La programmation linéaire
- Problème du flot maximal
- Problème d'affectation
- Problème de transport
- Phénomènes aléatoires en RO

Introduction

Problèmes

- **Problème 1:**

On doit organiser un pont aérien pour transporter 1600 personnes et 90 tonnes de bagages. Les avions disponibles sont de deux types: 12 du type A et 9 du type B. Le type A peut transporter, à pleine charge, 200 personnes et 6 tonnes de bagages. Le type B, 100 personnes et 6 tonnes de bagages. La location d'un avion du type A coûte 800.000 euro; la location d'un avion du type B coûte 200.000 euro.

Déterminer le coût du pont aérien minimal.

- **Problème 2:**

Dans un atelier de production, nous avons N tâches à exécuter sur une machine. Chaque tâche est caractérisée par une durée opératoire et une date due (date de livraison). Déterminer l'ordre de passage des tâches sur la machine qui minimise la somme des retards.

Définitions et règles

Définitions:

- La recherche opérationnelle est l’ensemble **des méthodes et techniques** d’analyse et de synthèse des phénomènes d’organisation permettant l’élaboration de la meilleure solution envisageable.
- La recherche opérationnelle est un outil d’**aide à la décision** qui permet d'optimiser une fonction économique en présence de contraintes multiples.
- La RO prend des formes très diverses en univers certain ou univers aléatoire.

Règles:

- Les choix des objectifs et du critère d’optimisation sont à la charge de l'entrepreneur
- la RO consiste à trouver la meilleure solution envisageable (solution optimale ou proche de l’optimale).

Domaines d'application

- **Domaine de transport et de distribution**
  - Recherche d'une tournée entre les clients qui minimise le coût de transport sachant que le départ soit d'un entrepôt
- **Domaine de planification et d'ordonnancement.**
  - Recherche du meilleur séquencement des tâches qui minimise la date de fin d'exécution de la dernière tâche
- **Domaine économique et financier**
  - Recherche de la meilleure affectation d'un capital sur des banques proposant des offres d'épargnes (maximisation des revenus)
- **Domaine militaire**
  - Placement du minimum nombre de radars tout en assurant une couverture totale de l'espace à surveiller
- **Domaine informatique:** Recherche du meilleur ordonnancement des tâches sur les processeurs
- **Etc…**

Types de problèmes traités

- **Problème combinatoire**
- **Problème aléatoire**

Types de problèmes traités

Problème combinatoire

- Un problème est dit **combinatoire** lorsqu'il comprend un grand nombre de solutions admissibles parmi lesquelles on cherche une solution optimale ou proche de l'optimum.
- Le but d'utiliser les méthodes de la recherche opérationnelle dans la résolution de ce genre de problème est pour éviter l'énumération de toute les solutions possible afin de choisir la meilleure (affecter 25 secrétaires à 25 postes $25! \approx 1{,}5\ 10^{25}$ solutions. Si 1 solution$/\mu s$ alors il faut 5.10^9siècles pour énumérer toutes les solutions)
- Deux type de problèmes: discret (variables entières) et continu (variables non entières)
- Exemples de problèmes:
  - ***Problème d'ordonnancement :*** pour la réalisation d'un projet consistant à accomplir des tâches soumises à des contraintes (d'antériorité, de dates…), trouver l'ordre des tâches, la solution la moins longue (et/ou la moins chère), les marges de réalisation des tâches…

Types de problèmes traités

- **Problème combinatoire**
  - Exemples de problèmes:
    - ***Chemin optimal*** dans un graphe: trouver le chemin le moins (plus) cher pour aller de A à B.
    - ***Problème de flot maximal*** dans un réseau de transport (graphe avec E/S) : acheminer le plus possible de marchandises de E vers S compte tenu des capacités de transport de chaque arc.
    - ***Problème d'affectation*** : pour 25 personnes à affecter dans 25 postes : chaque personne donne ses préférences (de 1 à 25). On cherche une solution satisfaisant le maximum personnes.
    - ***Problème de transport*** : trouver les quantités x_{ij} à transporter depuis m usines vers les distributeurs pour minimiser la somme des coûts de transport.
    - ***Problème du voyageur de commerce*** : trouver le parcours minimisant le coût total (déplacement + hôtel) du voyageur qui doit visiter tous ses clients et revenir chez lui.

Types de problèmes traités

Problème aléatoire

- Un problème est dit aléatoire s'il consiste à trouver une solution optimale à un problème qui se pose en termes incertains (probabilité, espérance, processus stochastique…).
- Exemples de problèmes:
  - ***Files d'attente*** (grandeurs aléatoires de base : instants d'arrivée des clients, durée du service) : optimiser le compromis entre le nombre de serveurs et la durée d'attente des clients.
  - ***"Fiabilité"*** *:usure et renouvellement des équipements* (grandeurs aléatoires : instants des pannes, durée des réparations) : optimiser le compromis entre le coût des arrêts du matériel et le taux de renouvellement du matériel.
  - ***Gestion des stocks*** (grandeurs aléatoires : instants des demandes de pièces, quantités demandées) : optimiser le compromis entre le coût de stockage et le coût des ruptures de stock.

Conclusion

- **Pour résoudre un problème d'optimisation avec la RO**
  - Modéliser le problème (graphe, modèle mathématique, simulation, etc)
  - Appliquer une méthode de résolution (en générale, une méthode à base de code informatique)
- **La recherche opérationnelle est une discipline carrefour entre**
  - ***<u>Mathématiques</u>*** algèbre de Boole, algèbre linéaire, théorie des graphes, probabilités, processus stochastiques.
  - ***<u>Informatique</u>*** l'informatique est un outil pour la RO (algorithmes rapides). La RO est un outil pour l'informatique (problème de files d'attentes).
  - ***<u>Économie</u>*** Recherche des solutions optimales et proches de l'optimales.

Modélisation par les graphes et résolution

Vocabulaire de la théorie des graphes

- **Graphe:**

Un graphe G(X,U) est la donnée

- d'un ensemble fini X appelé ensemble des sommets
- D'une partie U de X×X appelée ensemble des arcs (arêtes)

On dit que le graphe G(X,U) contient un arc joignant l'extrémité initiale x et l'extrémité terminale y.

- **Arête:**

Connexion entre deux sommets A et B de G(X,U) sans spécification de l'extrémité initiale et de l'extrémité terminale.

- **Arc**

Connexion entre deux sommets A et B de G(X,U) avec spécification de l'extrémité initiale et de l'extrémité terminale. On note l'arc d'extrémité initiale A et d'extrémité terminale B par (A,B)

Vocabulaire de la théorie des graphes

- **Graphe orienté:**

Un graphe G(X,U) est dit orienté s'il est composé par

- d'un ensemble fini X appelé ensemble des sommets
- D'une partie U de X×X appelée ensemble des arcs

- **Graphe non orienté:**

Un graphe G(X,U) est dit orienté s'il est composé par

- d'un ensemble fini X appelé ensemble des sommets
- D'une partie U de X×X appelée ensemble des arêtes

- **Sous graphe:**

Soit G(X,U) un graphe et G'(X',U') un autre graphe. Si X'⊂ X et U'⊂ U alors G' est un sous graphe de G.

- **Chemin**

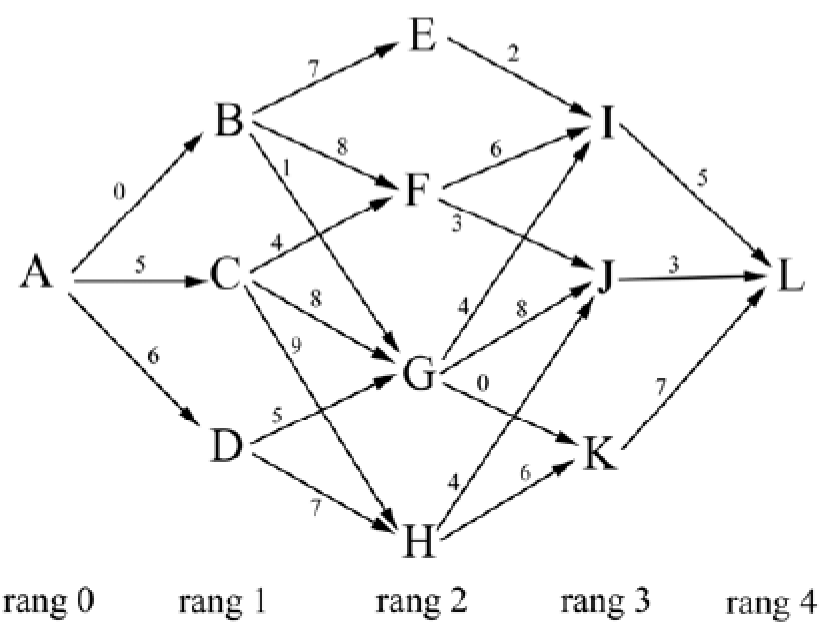
Un chemin est une suite d'arcs U_1, U_2, …, U_n, tel que l'extrémité terminale de l'arc U_i égale à l'extrémité initiale de l'arc U_{i+1}.

Modélisation par des graphes et résolution

Les graphes sont beaucoup utilisés dans la modélisation de nombreux problèmes industriels, informatique, économique, militaire, etc.

- **Problèmes décomposables en sous problèmes**
  - **Problème de recherche de chemin de coût minimal**

énoncé : Une voie de chemin de fer doit être construite entre les villes A et L. Trouver les villes intermédiaires pour que le coût de construction soit minimal (coût 0 = voie déjà construite).



- pas de retour en arrière,
- les arcs vont tous d'un rang *i* à un rang *i*+1.

Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
  - **Problème de recherche de chemin de coût minimal**
  - **Problème de sac à dos (Knapsac)**

énoncé: un randonneur veut remplir son sac à dos en maximisant la valeur nutritive totale du sac sans dépasser le poids de 15 kg. On dispose de 3 aliments avec les caractéristiques suivantes :

aliment	1	2	3
Poids p_i (kg)	6	3	9
Valeur nutritive c_i	15	10	35

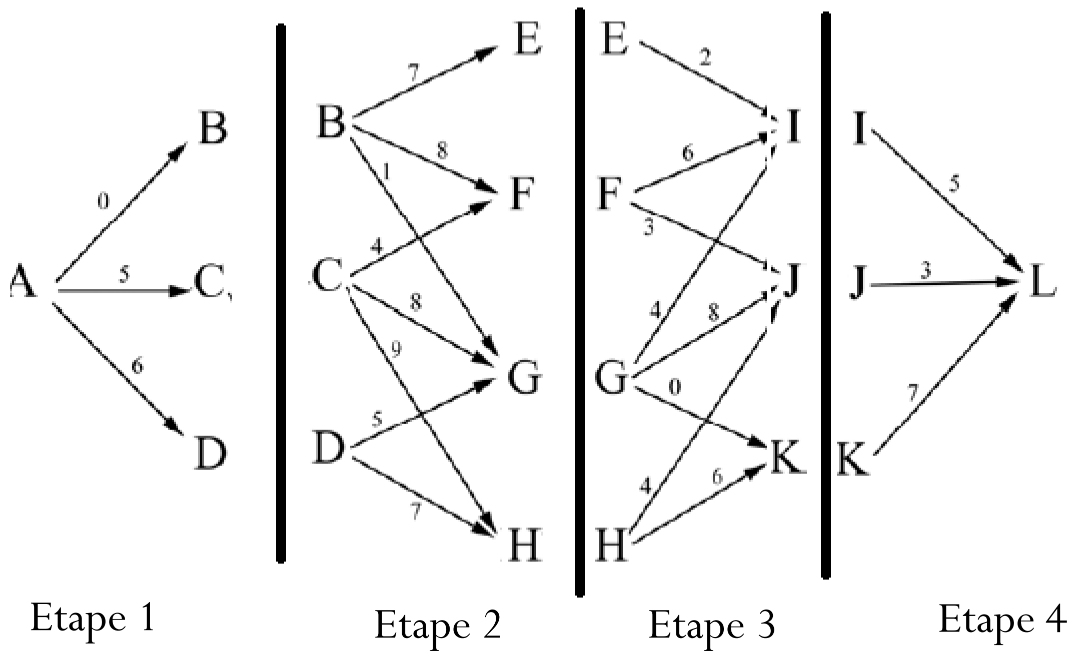
Soit x_i le nombre de boites retenues pour l'aliment i. On cherche les x_i qui maximisent $c_1 x_1 + c_2 x_2 + c_3 x_3$ sous la contrainte de poids $p_1 x_1 + p_2 x_2 + p_3 x_3 \leq 15$.

Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes: résolution par la programmation dynamique**
  - Un algorithme de programmation dynamique n’est pas toujours possible.
  - Il faut que le problème à résoudre vérifie une propriété de sous structure optimale.
  - Il faut qu’une solution optimale d’un problème contienne aussi des solutions optimales pour ses sous problèmes.
  - ***<u>Méthode de résolution :</u>***
    - Définir les étapes du problème
    - Définir les alternatives dans chaque étape
    - Définir l’état (l’évaluation) de chaque alternative d’une étape
    - Déterminer la solution en fonction du critère à optimiser

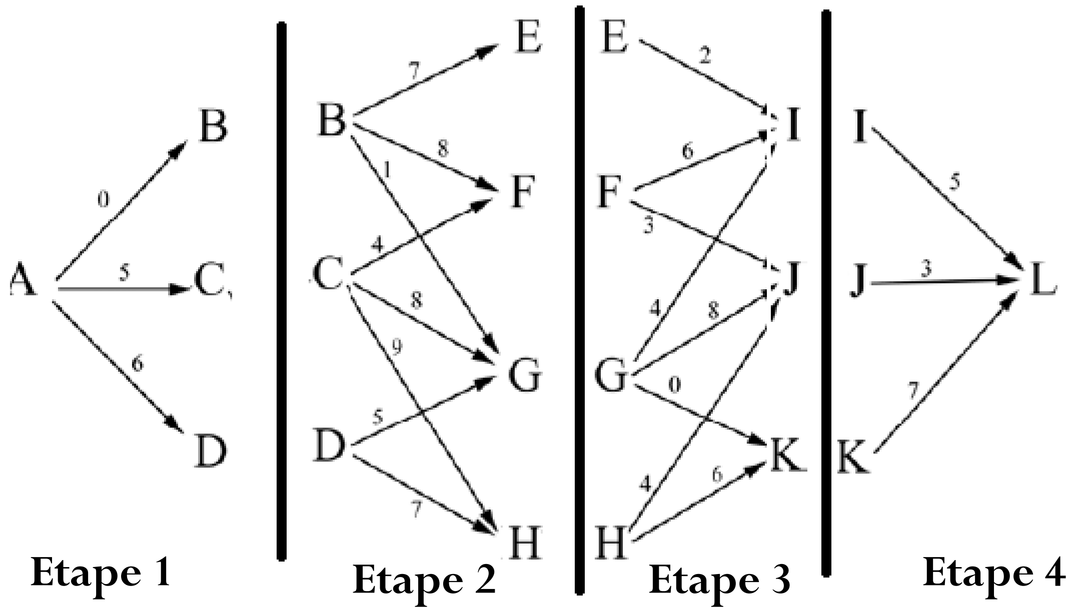
Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
  - **Problème de recherche de chemin de coût minimal**
  - ***Principe de résolution :*** "tout sous-chemin d'un chemin optimal est forcément optimal"



Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
  - **Problème de recherche de chemin de coût minimal**
  - ***<u>Méthode de résolution :</u>*** **La programmation dynamique**

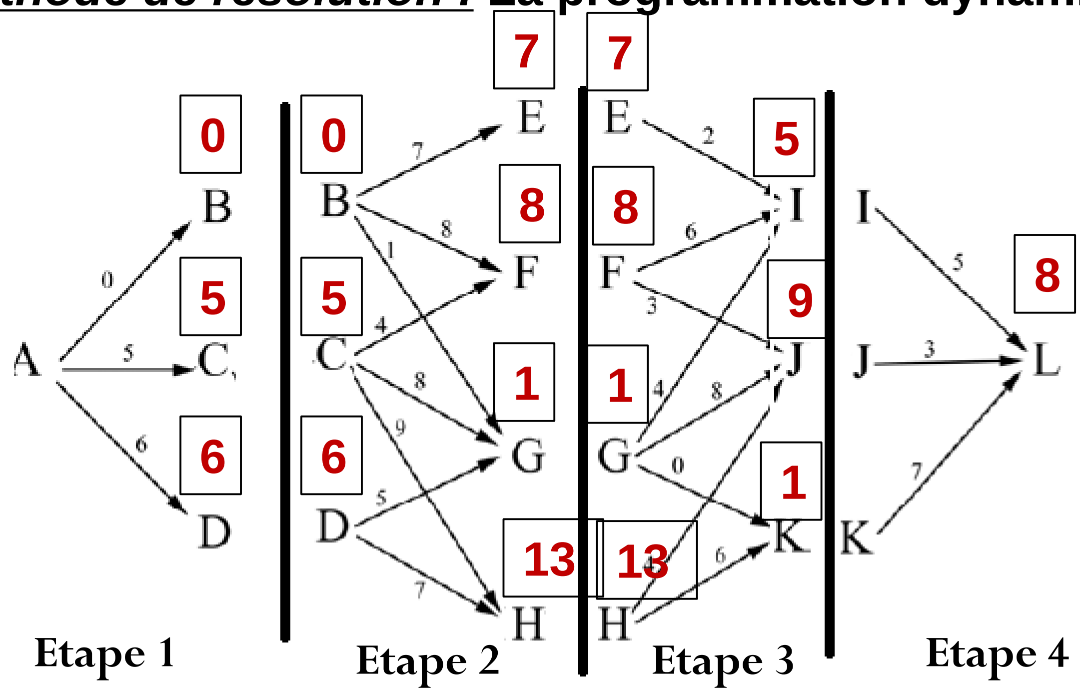


Alternatives

De A à B	De B à E, F, G	De E à I	De I à L
De A à C	De C à F, G, H	De F à I,J	De J à L
De A à D	De D à G, H	De G à I,J,K	De k à L
		De H à J,K	

Modélisation par des graphes et résolution

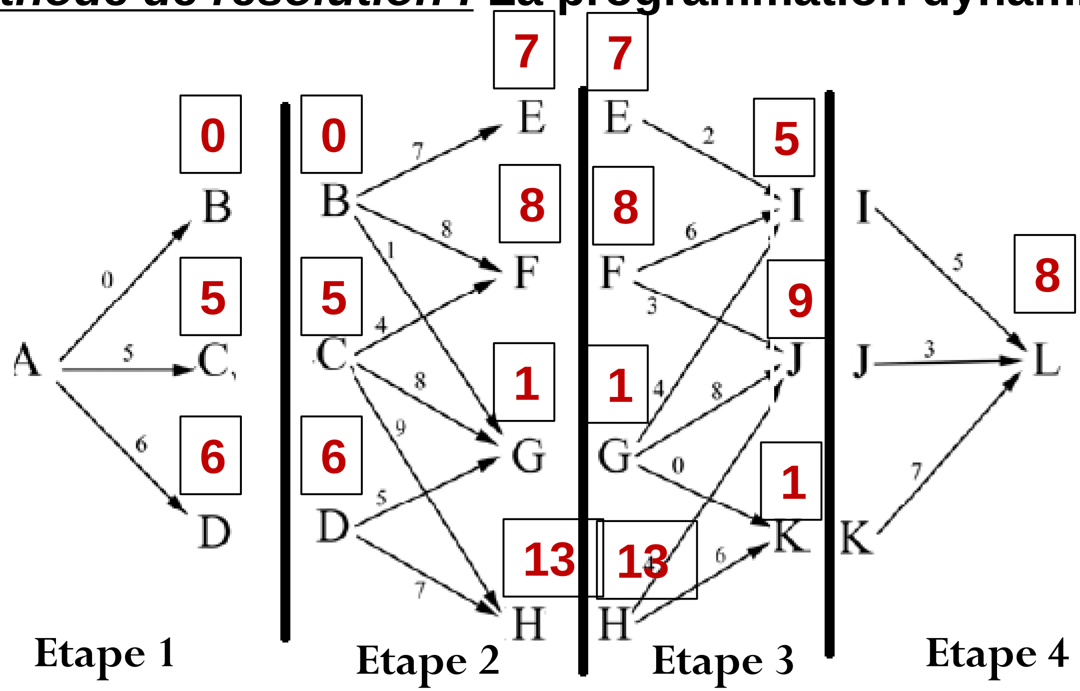
- **Problèmes décomposables en sous problèmes**
  - **Problème de recherche de chemin de coût minimal**
  - ***<u>Méthode de résolution :</u>*** **La programmation dynamique**



les états de l'étape 1	les états de l'étape 2	les états de l'étape 3	Étape 4
Min d(B)= 0	Min d(E)= 7	Min d(I)= 5	Min d(L)= 8
Min d(C)= 5	Min d(F)= 8	Min d(J)= 9	
Min d(D)= 6	Min d(G)= 1	Min d(K)= 1	
	Min d(H)=13		

Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
  - **Problème de recherche de chemin de coût minimal**
  - ***<u>Méthode de résolution :</u>*** **La programmation dynamique**



Fonction récursive de calcul

Soit $F_i(x_i)$ la distance minimale jusqu'au nœud i de l'étape i

Soit $C(x_{i-1},x_i)$ le coût entre le nœud x_{i-1} et x_i

$$F_i(x_i)=\min\{C(x_{i-1},x_i) + F_{i-1}(x_{i-1})\}$$

Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **But:** il s'agit d'ordonner dans le temps un ensemble d'opérations (tâches) contribuant à la réalisation d'un projet, ces opérations étant soumises à des contraintes.
  - **Contraintes:** 3 types de contraintes
    - ✓ potentielles : contrainte de précédence (la tâche *a* doit se dérouler avant la tâche *b*) et contrainte de date (la tâche ne peut commencer avant le 15/3, la tâche doit se terminer avant le 3/4).
    - ✓ disjonctives : imposent la non réalisation simultanée de 2 tâches (ex : si on a une seule machine).
    - ✓ cumulatives : prennent en compte les limites des ressources (homme, machines, moyens financiers, etc)

Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **But:**
  - **Contraintes:**
  - **Processus d'ordonnancement (5 actions)**
    - ✓ Analyse du projet (découpage en tâches avec les durées et les contraintes).
    - ✓ Ordonner les tâches en respectant les contraintes → graphe.
    - ✓ Trouver le chemin critique (il donne la durée minimale du projet).
    - ✓ Calculer les dates de démarrage au plus tôt et au plus tard des tâches → marges de réalisation.
    - ✓ Réactualisation permanente du graphe, des dates, des marges… pour prendre en compte les aléas extérieurs

Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **But:**
  - **Contraintes:**
  - **Méthodes d'ordonnancement (5 actions)**
  - **Mise en place des actions (action 1)**
    - ✓ Découper le projet en sous ensembles élémentaires à fonction simple (tâches) et définir pour chacune une durée, un état d'entré et un résultat final qui peut être l'état d'entré d'une autre tâche.

Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **But:**
  - **Contraintes:**
  - **Méthodes d'ordonnancement (5 actions)**
  - **Mise en place des actions (action 2)**
    - ✓ Quand l'ensemble des tâches à réaliser sont répertoriées, il faut passer à l'ordonnancement de ces tâches. Pour cela, il faut tenir compte des contraintes de précédence et construire une matrice d'antériorité.

Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **But:**
  - **Contraintes:**
  - **Méthodes d'ordonnancement (5 actions)**
  - **Mise en place des actions (action 2)**.

	opération	durée (mois)	tâches pré requises	recherche des rangs
a	construction des voies d'accès	4	-	-
b	travaux de terrassement	6	*a*	*a*
c	construction des bâtiments administratifs	4	-	-
d	commande de matériel électrique	12	-	-
e	construction de la centrale	10	*b, c, d*	*b, c, d*
f	construction du barrage	24	*b, c*	*b, c*
g	installation des galeries et conduites forcées	7	*a*	*a*
h	montage des machines	10	*e, g*	*e, g*
i	essais de fonctionnement	3	*f, h*	*f, h*

Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **Mise en place des actions (action 2: matrice des précédences)**.

	a	b	c	d	e	f	g	h	i	
A										0
B	1									1
C										0
D										0
E		1	1	1						3
F		1	1							2
G	1									1
H					1		1			2
I						1		1		2

En ligne et colonne, chacune des tâches est représentée et $a_{ij} = 1$ si la tâche en colonne j précède directement celle en ligne i. Les tâches de niveau 1 sont les tâches i pour lesquelles la somme sur les colonnes vaut zéro. Ensuite, on supprime ces tâches de la matrice et on déduit les tâches de niveau 2 de la même manière. Ce procédé est répété jusqu'à ce que toutes les tâches soient classées.

Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **Mise en place des actions (action 2: matrice des précédences).**

	a	b	c	d	e	f	g	h	i	
A										0
B	1									1
C										0
D										0
E		1	1	1						3
F		1	1							2
G	1									1
H					1		1			2
I						1		1		2

- Niveau 1 = {A,C,D}

Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **Mise en place des actions (action 2: matrice des précédences).**

	a	b	c	d	e	f	g	h	i	
A										0
B	1									0
C										0
D										0
E		1	1	1						1
F		1	1							1
G	1									0
H					1		1			2
I						1		1		2

- Niveau 1 = {A,C,D}
- Niveau 2={B,G}

Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **Mise en place des actions (action 2: matrice des précédences).**

	a	b	c	d	e	f	g	h	i	
A										0
B	1									0
C										0
D										0
E		1	1	1						0
F		1	1							0
G	1									0
H					1		1			1
I						1		1		2

- Niveau 1 = {A,C,D}
- Niveau 2={B,G}
- Niveau 3={E,F}

Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **Mise en place des actions (action 2: matrice des précédences).**

	a	b	c	d	e	f	g	h	i	
A										0
B	1									0
C										0
D										0
E		1	1	1						0
F		1	1							0
G	1									0
H					1		1			0
I						1		1		1

- Niveau 1 = {A,C,D}
- Niveau 2={B,G}
- Niveau 3={E,F}
- Niveau 4={H}

Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **Mise en place des actions (action 2: matrice des précédences).**

	a	b	c	d	e	f	g	h	i	
A										0
B	1									0
C										0
D										0
E		1	1	1						0
F		1	1							0
G	1									0
H					1		1			0
I						1		1		0

- Niveau 1 = {A,C,D}
- Niveau 2={B,G}
- Niveau 3={E,F}
- Niveau 4={H}
- Niveau 5={I}

Modélisation par des graphes et résolution

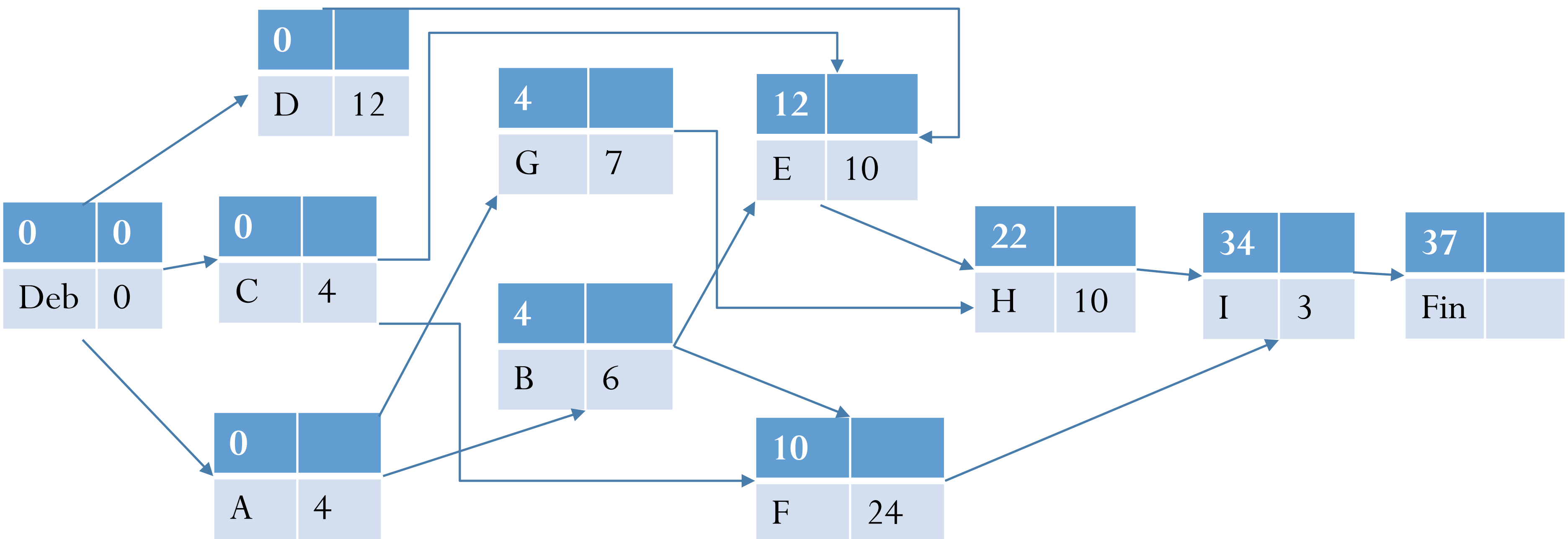
- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **Mise en place des actions (action 2: Graphe potentiels-tâches)**.
    - ✓ Les tâches, caractérisées par un numéro et une durée, sont représentées à l'aide de rectangles et possèdent un début et une fin.
    - ✓ Les relations de précédence sont matérialisées par des flèches

date + tôt	date + tard
Tâche n	Durée

Représentation d'une tâche dans la methode MPM

Modélisation par des graphes et résolution

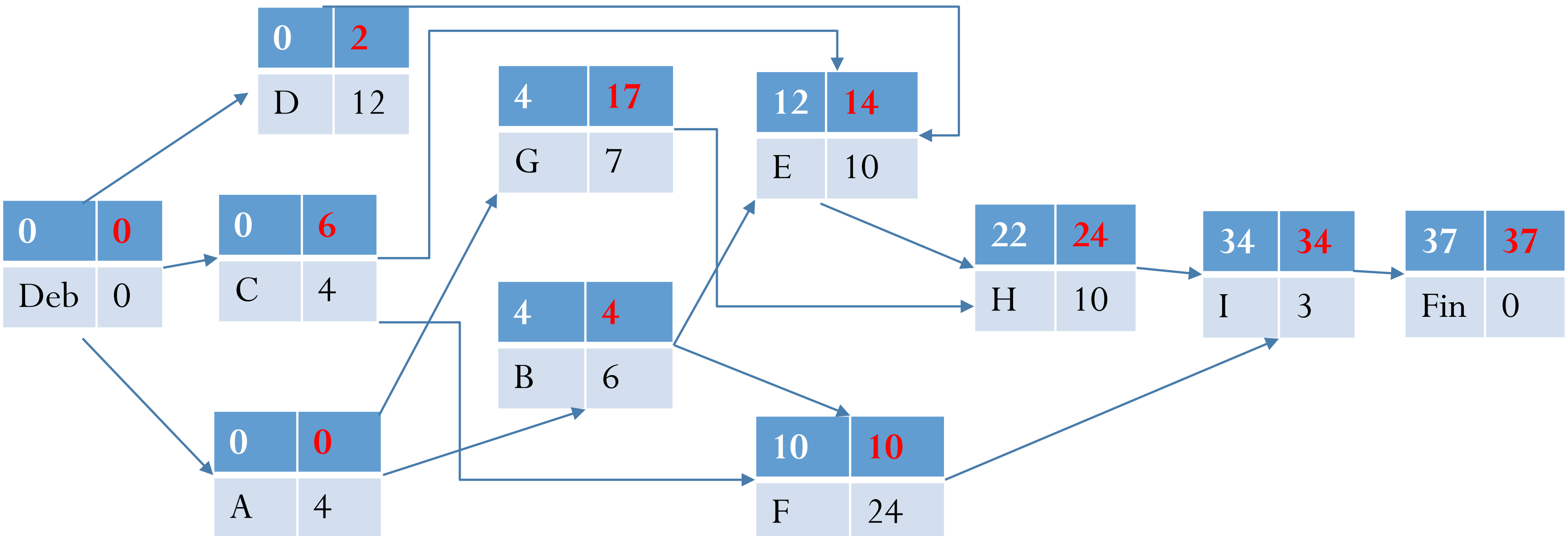
- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **Mise en place des actions (action 2: Graphe potentiels-tâches)**.



Date au plus tôt: on regarde parmi l'ensemble des tâches en relation de précédence directe avec i laquelle se terminera en dernier. Quand toutes les tâches ont été traitées, on regarde celle se terminant en dernier.

Modélisation par des graphes et résolution

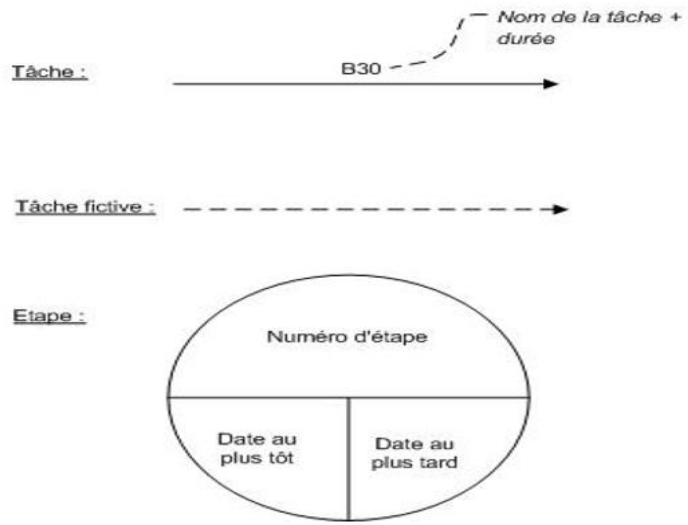
- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **Mise en place des actions (action 2: Graphe potentiels-tâches).**



<u>Date au plus tard:</u> On part de la dernière tâche pour laquelle on fixe la date au plus tard égale à la date au plus tôt, et on remonte le réseaux en suivant les précédences. Pour une tâche i donnée, on regarde parmi ses successeurs directs quelle tâche j commencera en premier par rapport à sa programmation au plus tard. La date au tard de la tâche i est obtenue en retranchant de la date au plus tard de la tâche j la durée de la tâche i.

Modélisation par des graphes et résolution

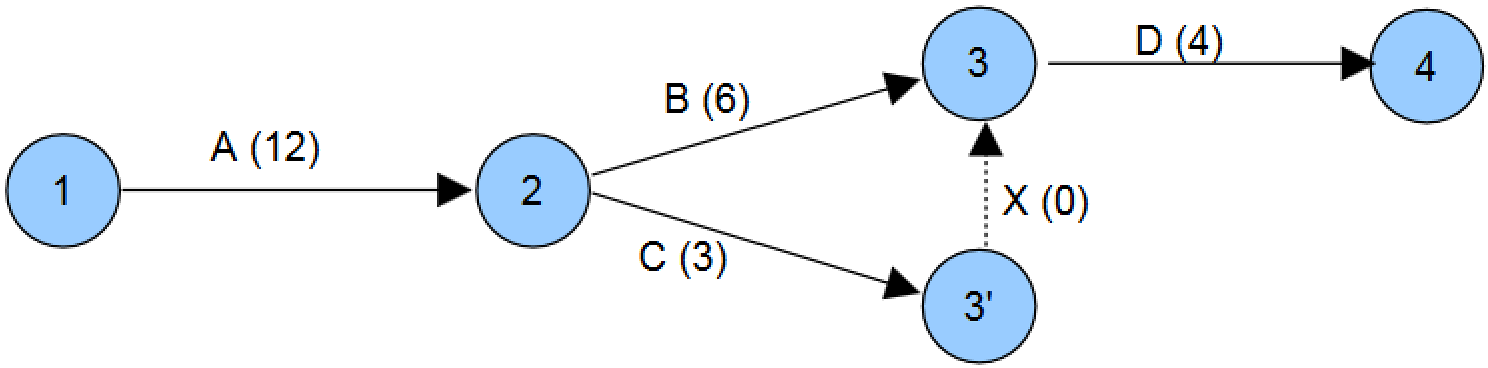
- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **Mise en place des actions (action 2: Graphe PERT).**
    - ✓ Les tâches, caractérisées par un numéro et une durée, sont représentées à l'aide de flèches.
    - ✓ Chaque tâche est précédée et suivie par une étape.



Règles et notations de représentation

Modélisation par des graphes et résolution

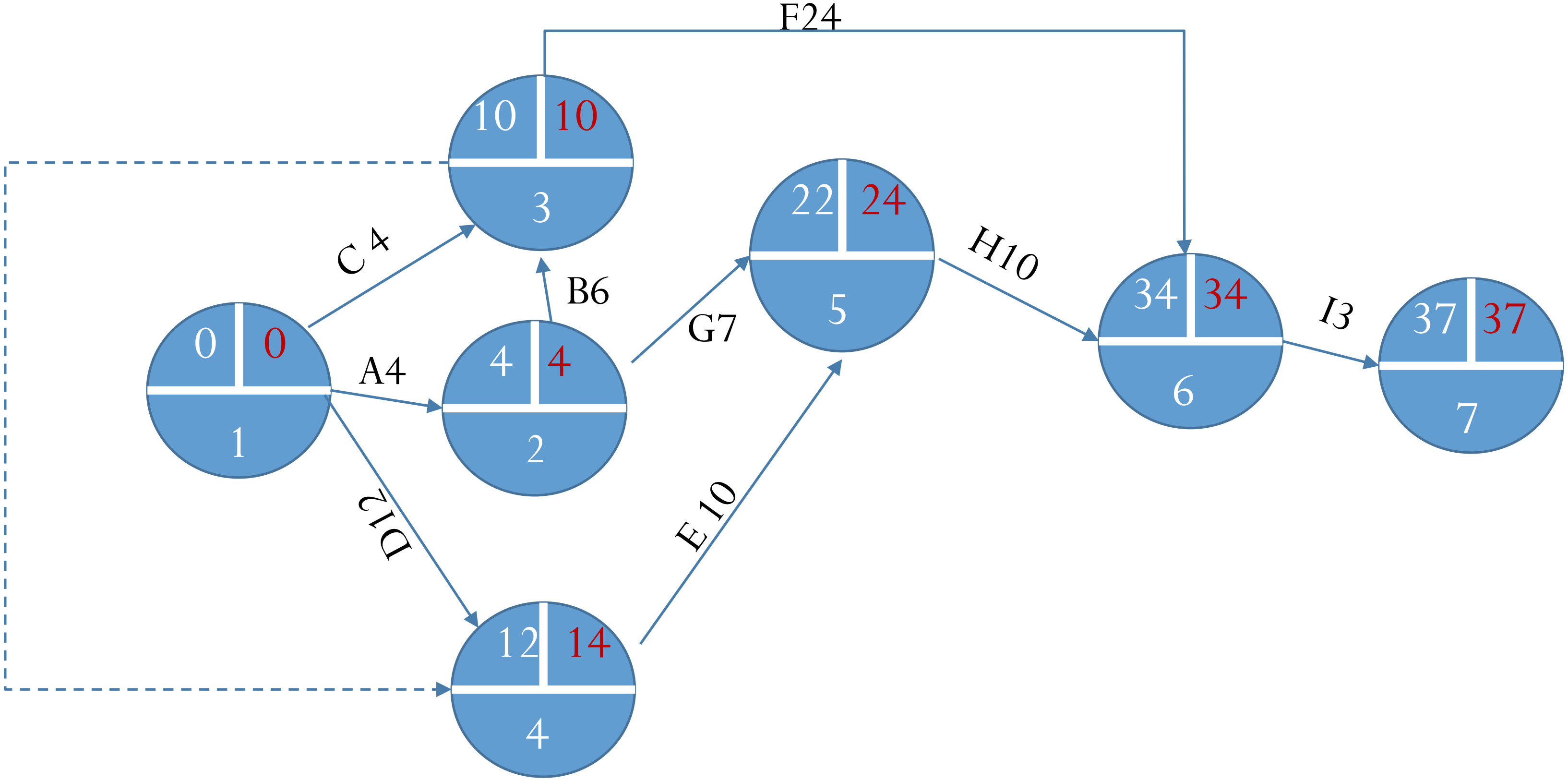
- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **Mise en place des actions (action 2: Graphe PERT).**
    - ✓ Si l'on souhaite que D ne commence que si B et C sont terminées:



    - ✓ Du fait de la règle de construction qui interdit de faire se dérouler les deux tâches B et C simultanément, nous utilisons une tâche x (0) dite «tâche fictive» qui sert à représenter ce type de contraintes de liaison (contraintes d'antériorité)

Modélisation par des graphes et résolution

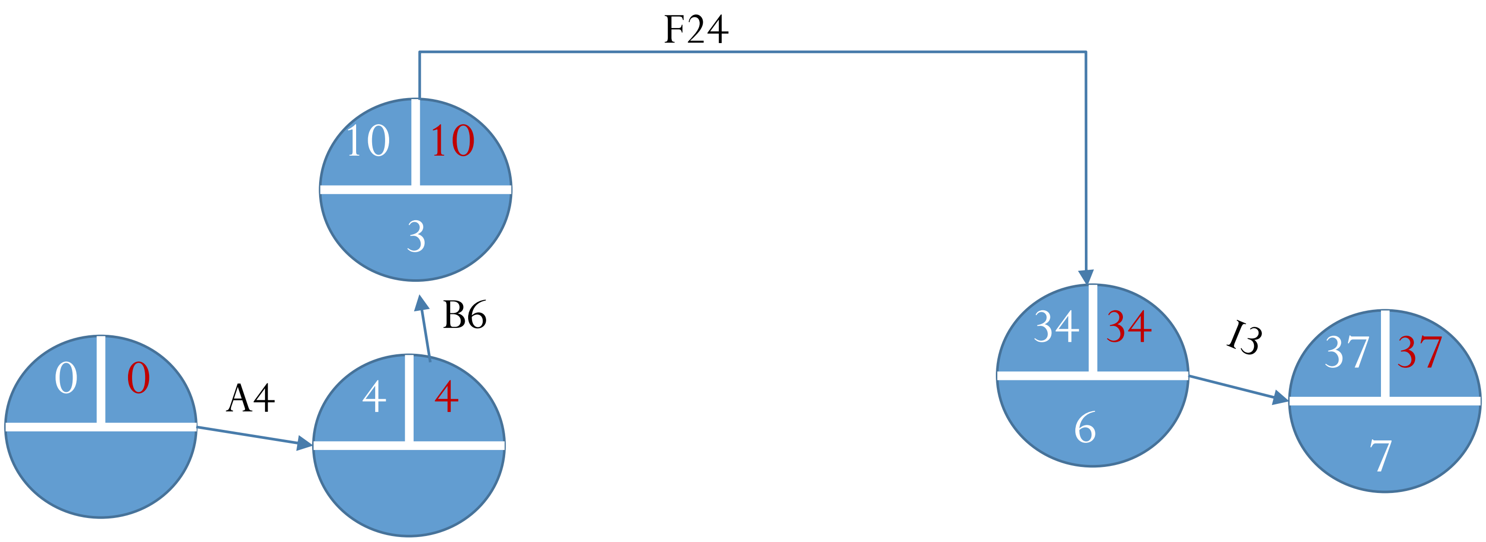
- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **Mise en place des actions (action 2: Graphe PERT).**



Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **Mise en place des actions (action 3)**.

Le chemin critique est le chemin qui passe par les sommets pour lesquels la date au plutôt est égale à la date au plus tard et par les tâches dont les durées ont contribuées au calcul des dates au plutôt.



Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **Mise en place des actions (action 4)**.

Marge totale d'une tâche : représente le retard maximum sur le début de la tache i si l'on accepte de repousser les opérations suivantes sans toutes fois allonger la durée totale du projet.

$MT_i = DateAuPlusTard\ i - DateAuPlusTot\ i$

Marge libre d'une tâche i : représente le retard maximum du démarrage de la tache i sans que la date de démarrage des autres opérations soit modifiée.

$ML_i = Min_{j \in J}(DateAuPlusTot\ j - (DateAuPlusTot\ i + Duree\ i))$ *(avec J l'ensemble de ses successeurs directs)*

Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **Mise en place des actions (action 4).**

Tâche	a	b	c	d	e	f	g	h	i
Deb Tôt	0	4	0	0	12	10	4	22	34
Deb Tard	0	4	6	2	14	10	17	24	34
MT	0	0	6	2	2	0	13	2	0
ML	0	0	6	0	0	0	11	2	0

Modélisation par des graphes et résolution

- **Problèmes décomposables en sous problèmes**
- **Problème d'ordonnancement**
  - **Mise en place des actions (action 4)**.
  - **Diagramme de GANTT**

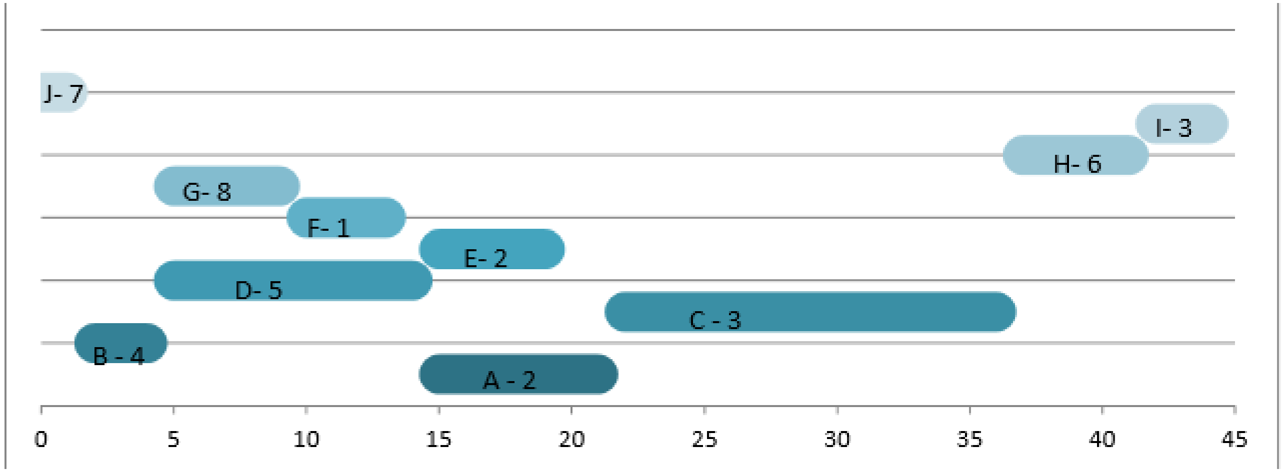
Quand on connaît les dates de début et fin prévues pour chacune des tâches du projet, on peut représenter graphiquement la planification sur un diagramme de GANTT.

Dans ce diagramme, les lignes représentent chacune des tâches et les colonnes correspondent aux périodes (jours, semaines ou mois du calendrier selon le projet). Le temps estimé pour une tâche se modélise par une barre horizontale dont l'extrémité gauche est positionnée à la date de démarrage prévue et l'extrémité droite à la date prévue pour la fin de réalisation.

Il est généralement possible (et utile) de faire apparaître des ressources, humaines ou matérielles, afin de permettre une estimation des besoins et donner une idée du coût global.

Modélisation par des graphes et résolution

- **Diagramme de GANTT**



- **Diagramme de des charges**