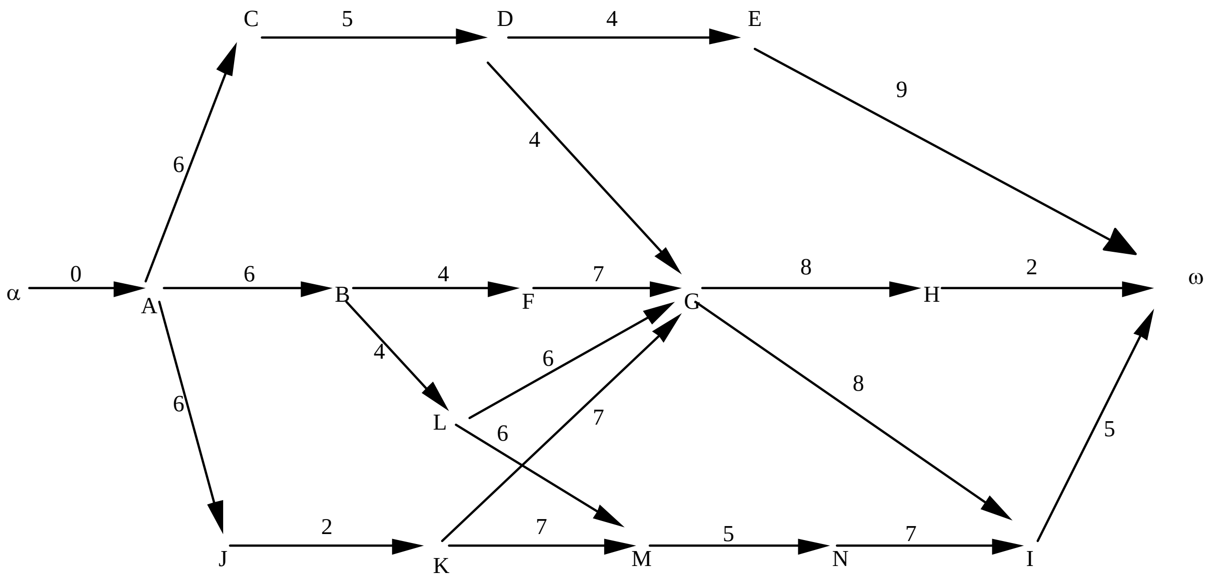
Le problème central de l'ordonnancement - Exercices - corrigé

I a) Représentation du graphe potentiel-tâches



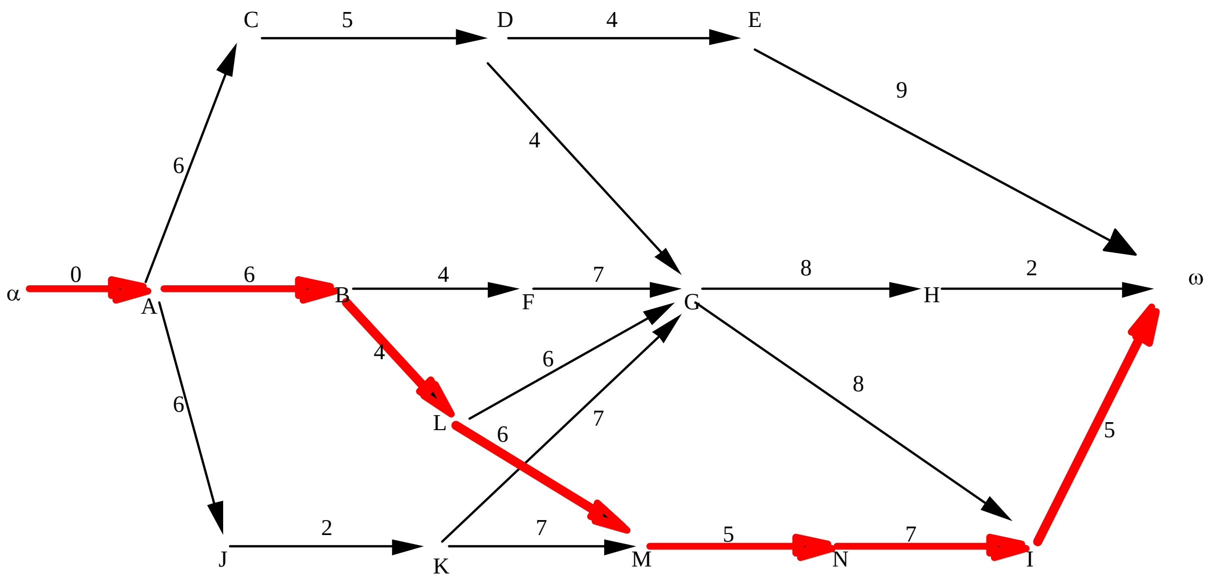
NB : L'introduction de la tâche α n'est pas indispensable dans la mesure où A est la seule tâche sans prédécesseur.

b) Calcul des dates au plus tôt :
Le graphe est sans circuit. On est dans le cas standard. La date au plus tôt de chaque tâche se calcule à partir de celles de ses prédécesseurs.
Ordre d'examen possible :
A , B, C ou J , D F ou L ou K , E G , M , H N I , ω

Tâches	Durée	t_i^*	T_i^*	Marge Totale	Marge libre
α	0	0	0	0	
A	6	0	0	0	
B	4	6	6	0	
C	5	6	11	5	0
D	4	11	16	5	0
E	9	15	24	9	9
F	7	10	13	3	0
G	8	17	20	3	0
H	2	25	31	6	6
I	5	28	28	0	
J	2	6	7	1	0
K	7	8	9	1	1
L	6	10	10	0	
M	5	16	16	0	
N	7	21	21	0	
ω	0	33	33	0	

(Vous avez intérêt à calculer en mettant ces résultats sur le graphe à coté de chaque sommet)
Durée minimale des travaux : 33
Tâches critiques : A, B, I, L , M, N
Chemin critique : α, A, B, L, M, N, I, ω

Les seules tâches de marge libre non nulle sont E, H et K : par exemple K a une date au plus tôt de 8 et ses deux successeurs G et M une date au plus tôt respectivement de 17 et 16 : si on commence K à 9, sa date au plus tard, M pourra encore commencer à la date 16 et G à 17.



Chemin critique

c) Diagramme de Gantt :

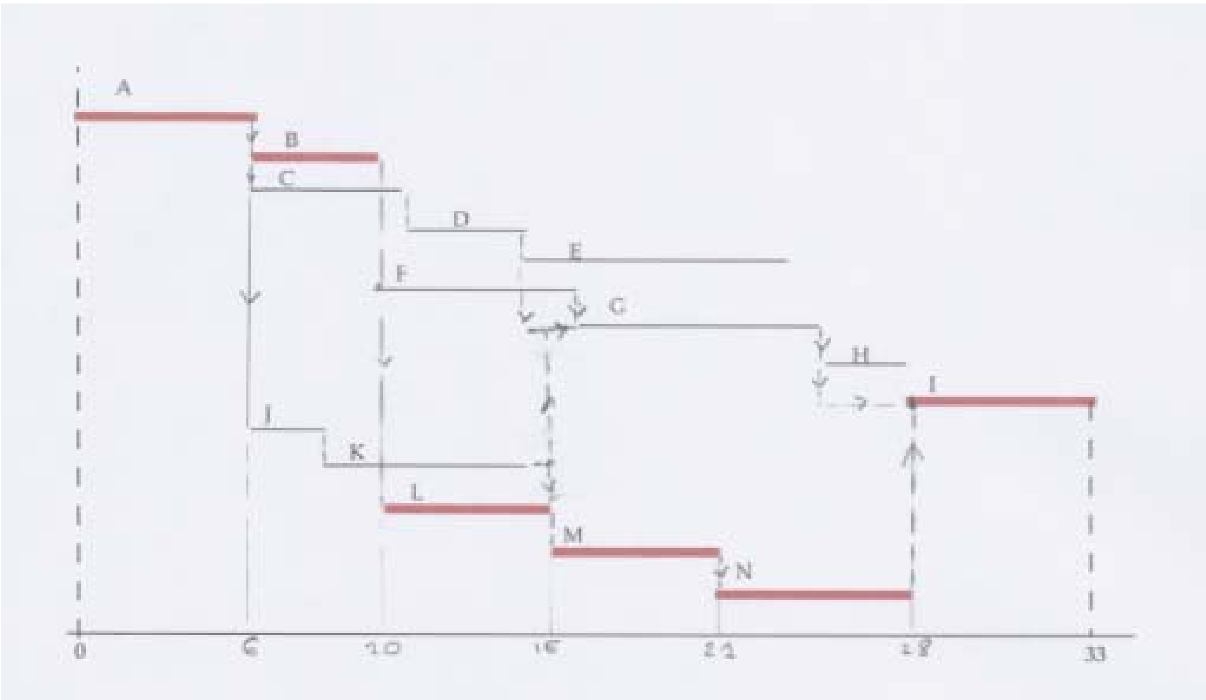


Diagramme de Gantt au plus tôt

Pour le diagramme au plus tard, il faut retarder toutes les tâches non critiques d'une durée égale à leur marge totale.

d) La tâche F doit commencer dès que B est terminée, se traduit par : $t_F = t_B + 4$ puisque B dure 4.

On vérifie que la représentation graphique et le calcul des dates au plus tard fournissent bien le résultat attendu : la date de début au plus tard de F doit devenir égale à 10.
$t_F \geq t_B + 4$ et $t_F \leq t_B + 6$ soit $t_B \geq t_F - 4$

On représente cette contrainte par 2 arcs (B,F) de longueur 4 et (F , B) de longueur –4.
(reportez ces arcs sur le graphe)
Ceci crée un circuit de longueur 0 dans le graphe.
Il existe un nouveau chemin de F à ω : le chemin F, B , L, M, N , I, ω de longueur –4 + 4 + 6 + 5 + 7 +5 = 23 . La date au plus tard de F devient égale à 10 (33 – 23) qui correspond bien à la fin de B et est égale à la date au plus tôt de F qui n'a pas changé (=10) . F devient critique.

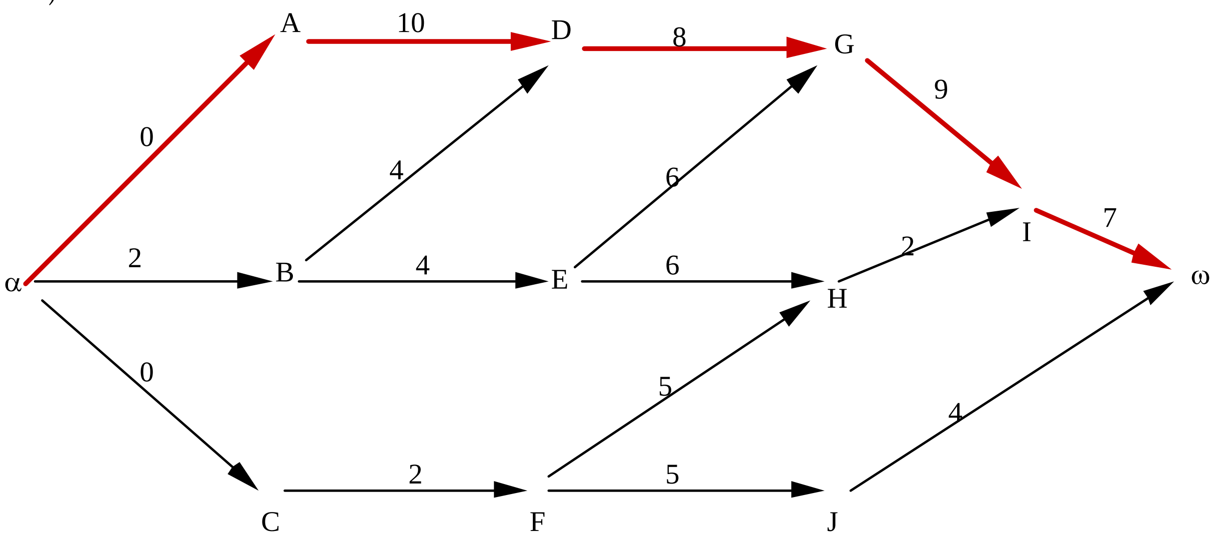
- La tâche E doit être terminée au plus tard 25 unités de temps après le début des travaux.

Cette contrainte ne change pas la durée totale.

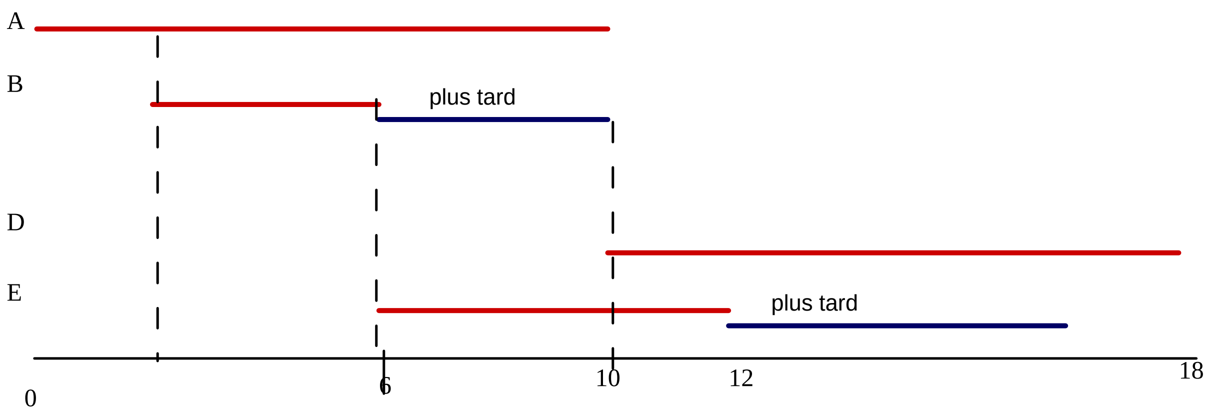
$t_E + 9 \leq 25$ soit $t_E \leq 16$ ou $t_\alpha \geq t_E - 16$.
On représente cette contrainte par un arc (E, α) de longueur – 16. *(à reporter sur le graphe)*
On crée ainsi un **nouveau chemin de E à ω** constitué de l'arc (E, α) suivi d'un chemin de α à ω, dont le plus long est évidemment le chemin critique de longueur 33. La longueur du plus long chemin de E à ω est maintenant de – 16+ 33 = 17
On a alors que la date de début au plus tard de E est égale à date de fin souhaitée des travaux (33) moins la longueur du plus long chemin de E à ω (17) soit 33 – 17 = 16. On aura donc bien E terminée avant la date 25.

- Les tâches B et D doivent commencer simultanément se traduit par : $t_B = t_D$
Cette contrainte est équivalente à : $t_D \geq t_B$ et $t_D \geq t_B$.
Elle est représentée par 2 arcs de longueur nulle (B,D) et (D,B) . *(à reporter sur le graphe)*
Il existe maintenant un chemin de longueur 11 de α à B : α A C D B . La date de début au plus tôt de B devient égale à 11. Comme cette tâche était critique la durée minimale des travaux est augmentée.
Le nouveau chemin critique est α A C D B L M N I ω. La durée totale étant de 38.

II a)



Si on considére les tâches A et B par exemple :
ces 2 tâches utilisent chacune 3 unités de ressource 1 pour laquelle on ne dispose que de 5 unités. Elles ne peuvent avoir lieu simultanément.
A est critique et la date de début au plus tard de B fait qu'on ne peut la reculer après la fin de A.
On a le même type de problème avec les 2 tâches D et E pour la ressource 2.



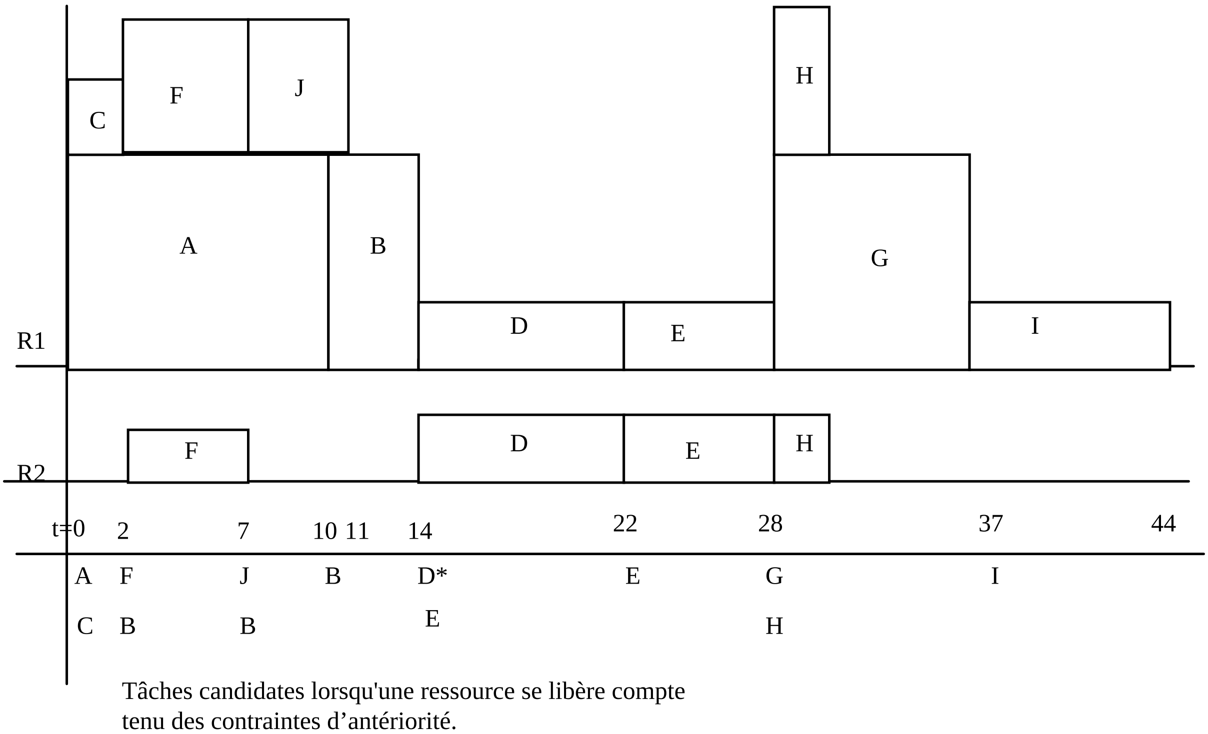
b) Les tâches sont classées dans l'ordre suivant (par leur date de début au plus tard) :

A	B	D	E	C	G	F	H	I	J
0	6	10	12	18	18	20	25	27	30

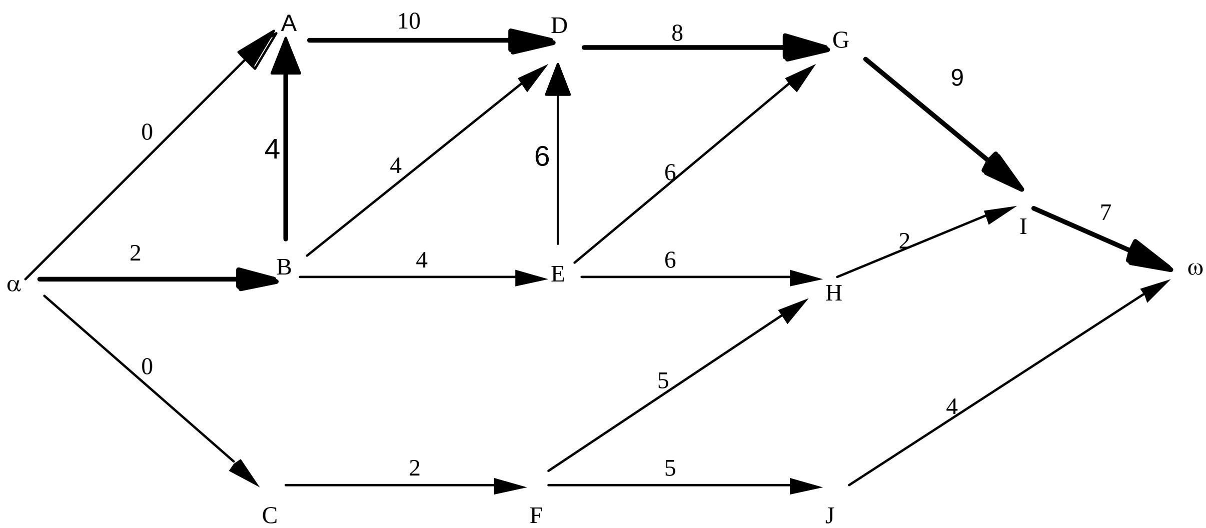
On fait une affectation aux dates:
t=0 A et C, t=2 F , t=7 J , t=10 B , t =14 D , t=22 E , t=28 G et H, t=37 I

Ce qui conduit à une durée totale de 44 alors qu'en négligeant les ressources la durée était de 34.
L'ordre de priorité retenu a imposé que D soit placé avant E.

L'utilisation des ressources correspondantes est la suivante :



c) Sur le graphe initial, on ajoute 2 arcs (B,A) et (E,D).
On résout à nouveau le problème sans tenir compte des ressources.

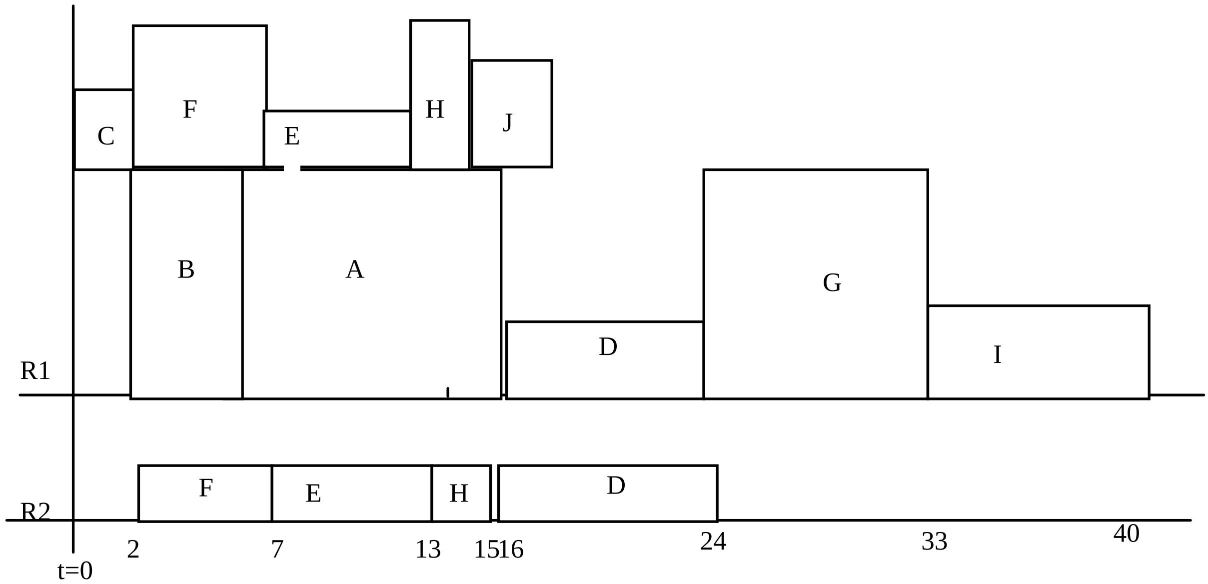


Les dates de début au plus tôt et au plus tard sont données par :

	A	B	C	D	E	F	G	H	I	J	ω
Début au plus tôt	**6**	**2**	0	**16**	6	2	**24**	12	**33**	7	**40**
Début au plus tard	**6**	**2**	24	**16**	10	26	**24**	31	**33**	36	**40**

La durée minimale est maintenant de 40, mais on n'a pas tenu compte des ressources.

Une fois les tâches critiques placées, on peut placer les autres en les reculant par rapport à leur date au plus tôt, tout en respectant leur date au plus tard, construisant ainsi un ordonnancement compatible avec les contraintes de ressources.



E ne commence qu'à la date 7 , H en 13, J en 15 .

Nous avons vu que, quoiqu'il arrive nous ne pouvions avoir A et B simultanément, de même D et E.
La première solution A avant B et D avant E conduisait à une durée totale de 34 mais non réalisable.

Pour savoir si la solution précédente est optimale, il resterait à tester le choix de mettre A avant B et E avant D et celui de mettre B avant A et D avant E.
Dans le premier cas on peut vérifier qu'on obtient une durée minimum de 44 , même si elle est réalisable du point de vue des ressources elle est moins bonne que celle que nous venons d'obtenir.
Il en est de même dans le deuxième cas qui conduit à une durée minimale de 46.

On peut donc conclure que la solution qui consiste à gérer le conflit entre les tâches A et B et les tâches D et E en mettant B avant A et E avant D est optimale.

Nous venons de traiter un problème d'ordonnancement avec contraintes de ressources. Si le nombre de tâches en conflit était important, nous ne pourrions pas obtenir une solution de manière efficace, l'examen de toutes les possibilités étant trop coûteux en temps de calcul.
Nous verrons ultérieurement comment on peut traiter ce genre de problèmes.