Problème de gestion de projet à contraintes de ressource
Correction

Question 1 Les ordonnancements suivants sont réalisables :

- Du temps 0 à 4, la ressource 1 est réalisée, puis de 4 à 7, la ressource 2, puis de 7 à 12 les ressources 3 et 4, et de 12 à 15, la ressource 4.
- Du temps 0 à 3, la ressource 2 est réalisée, puis de 3 à 8, les ressources 3 et 4, puis de 8 à 11 les ressources 4 et 1, et de 11 à 12, la ressource 1.

Question 2 Le 1er problème peut être ramené à un problème RCPSP avec une seule ressource de capacité 1 et les demandes en ressources des différentes tâches r_{j1} sont de 1.

Le 2e problème peut aussi être ramené à un problème RCPSP avec une seule ressource de capacité m et les demandes en ressources des différentes tâches r_{j1} sont de 1.

Le 3e problème correspond à un problème RCPSP avec m ressources de capacité 1 chacune et les demandes en ressources des différentes tâches sont telles que $r_{jk} = 1$ si $M_k \in \mu_j$ et $r_{jk} = 0$ sinon.

Question 3 La solution S est donnée par les instants de démarrage (dont on peut en déduire facilement les instants de fin) de chaque activité. La taille de S est donc bien bornée par la taille de l'entrée du problème RCPSP.

Pour vérifier que la solution S est réalisable, il faut :

- vérifier que les contraintes de capacité sont vérifiées à chaque instant : à chaque ajout ou arrêt d'une activité (prise dans l'ordre chronologique du démarrage des activités), il faut vérifier que les capacités des ressources ne sont pas dépassées (additions + comparaisons).
- vérifier que les contraintes de précédences sont vérifiées : il faut vérifier que $S_i + p_i \leq S_j$ si $i \to j$. (addition + comparaison)

Enfin, il faut vérifier que le temps de réalisation de S est inférieur à y. Toutes ces opérations peuvent se faire en $O(n.r + |A|)$.

Question 4 Cherchons à montrer que la version de décision du problème RCPSP avec $y = 2$ est déjà NP-complet.

Considérons une instance au problème Partition et transformons la en un problème RCPSP. On considère un problème RCPSP avec une seule ressource telle que $R_1 = \frac{1}{2}.\sum_{i=1}^{n} a_i$. Ce problème a n activités avec un temps de traitement $p_i = 1$ pour chacune et une utilisation de la ressource telle que $r_{i1} = a_i$.

Une telle transformation se fait en temps polynomial.

Montrons maintenant qu'une solution au problème Partition existe si et seulement si un ordonnancement réalisable au problème RCPSP précédent avec un temps de réalisation inférieur ou égal à 2 existe.

Si le problème Partition a une solution, alors il existe un ensemble $I \subset \{1, \cdots, n\}$ tel que $\sum_{i\in I} a_i = \sum_{i\in I} r_{i1} = R_1$. Alors on peut ordonnancer toutes les activités de I dans l'intervalle de temps $[0, 1]$, puis toutes les activités non dans I dans l'intervalle $[1, 2]$. On a bien un ordonnancement réalisable dont le temps de réalisation est égal à 2.

Inversement, si on a un ordonnancement réalisable dont le temps de réalisation est inférieur ou égal à 2, étant donnés les paramètres du problème, toute la ressource est utilisée avec toutes les activités sur l'intervalle de temps $[0, 2]$. Ceci implique que l'ensemble des activités ordonnancées dans l'intervalle de temps $[0, 1]$, et constituant un ensemble I, sont telles que $\sum_{i\in I} a_i = \sum_{i\in I} r_{i1} = R_1$. De même pour l'ensemble des activités ordonnancées dans l'intervalle de temps $[1, 2]$. I est donc une solution au problème Partition.

Question 5 r_i correspond au plus long chemin entre 0 et i sur le graphe G d'activité sur les nœuds moins p_i où la longueur d'un chemin correspond à la somme des poids associés aux nœuds composant ce chemin. De même q_i correspond au plus long chemin entre i et $n+1$ moins p_i.

Question 6

i	1	2	3	4
r_i	0	0	3	0
q_i	0	5	0	0
d_i	12	7	12	12

Question 7 On a $S_i + p_i > S_j$ (activité i termine après le début de j) et $S_j + p_j > S_i$ (activité j termine après le début de i).

Question 8 Pour C_0, il suffit de prendre les relations de précédences du problème. Pour D_0, il suffit de considérer les activités 2 à 2 et de regarder si $r_{ik} + r_{jk} > R_k$ pour une ressource k. Si c'est cas, alors $i - j \in D_0$. Pour N_0, il suffit de considérer les activités 2 à 2 et de regarder si $max\{d_i, d_j\} - min\{r_i, r_j\} < p_i + p_j$. Si c'est le cas, alors $i || j \in N_0$.

Question 9 Pour n'importe quelle paire d'activités i, j, on a $S_i + p_i \leq S_{n+1} \leq B \leq B + S_j$. D'où $S_j - S_i \geq p_i - B$.

Question 10 La relation est évidente à montrer si $i = j$. Si $i \to j$, on a $S_i + p_i \leq S_j$ et donc $S_j - S_i \geq d_{ij}$. Si $i || j$, on a $S_j + p_j > S_i$. Donc $S_j + p_j - 1 > S_i$, d'où $S_j - S_i \geq d_{ij}$. Si i et j sont 2 activités quelconque, on a aussi $S_j - S_i \geq d_{ij}$ d'après la question 9.