

10. Ordonnancement de tâches dépendantes

10.1. Contraintes de précédence

- sur l'ordre d'exécution des tâches les unes par rapport aux autres
- généralement décrites par un graphe orienté G
 - ✓ $T_a < T_b$ indique que la tâche T_a est un prédécesseur de T_b
 - ✓ $T_a \rightarrow T_b$ indique que la tâche T_a est un prédécesseur immédiat de T_b
- on prend seulement la précédence simple
 - si T_i est périodique de période P_i , alors T_j l'est aussi et $P_j = P_i$
- principe de l'établissement de l'ordonnancement :
 - transformer l'ensemble des tâches dépendantes en un ensemble de tâches « indépendantes » que l'on ordonnancera par un algorithme classique
 - par des modifications des paramètres des tâches
 - ✓ si $T_i \rightarrow T_j$ alors la règle de transformation doit respecter
 - ✗ $r_j > r_i$
 - ✗ $P_{i0} > P_{j0}$
 - validation de l'ordonnançabilité selon des critères utilisés pour des tâches indépendantes

10.2. contraintes de précédence et Rate Monotonic

- la transformation s'opère hors ligne sur la date de réveil et sur les délais critiques
 - ✓ $r^*_i = \text{Max}\{r_i, r^*_j\}$ pour tous les j tels que $T_j \rightarrow T_i$
 - ✓ si $T_i \rightarrow T_j$ alors $P_{i0} > P_{j0}$
- dans le respect de la règle d'affectation des priorités par RMA

10.3. contraintes de précédence et Deadline Monotonic

- la transformation s'opère hors ligne sur la date de réveil et sur les délais critiques
 - ✓ $r^*_i = \text{Max}\{r_i, r^*_j\}$ pour tous les j tels que $T_j \rightarrow T_i$
 - ✓ $D^*_i = \text{Max}\{D_i, D^*_j\}$ pour tous les j tels que $T_j \rightarrow T_i$
 - ✓ si $T_i \rightarrow T_j$ alors $P_{i0} > P_{j0}$
- dans le respect de la règle d'affectation des priorités par DMA

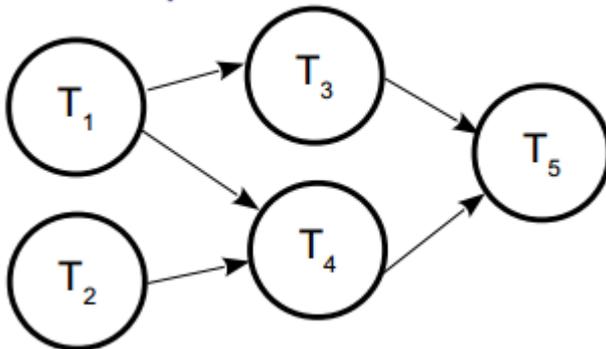
10.4. contraintes de précédence et EDF

- modification des échéances de façon à ce qu'une tâche ait toujours un d_i inférieur à celui de ses successeurs (algorithme de Chetto et al.)
- une tâche ne doit être activable que si tous ses prédécesseurs ont terminé leur exécution
- modification de la date de réveil et de l'échéance
 - ✓ $r^*_i = \text{Max}\{r_i, \text{Max}\{r^*_j + C_j\}\}$ pour tous les j tels que $T_j \rightarrow T_i$
 - ✓ $d^*_i = \text{Min}\{d_i, \text{Min}\{d^*_j - C_j\}\}$ pour tous les j tels que $T_i \rightarrow T_j$

- ✓ on itère sur les prédécesseurs et successeurs immédiats
- ✓ on commence les calculs par les tâches qui n'ont pas de prédécesseurs pour le calcul des r et par les tâches qui n'ont pas de successeur pour le calcul des d

Exemple

Soient 5 tâches dépendantes comme on peut le voir sur la figure et ayant les paramètres mentionnés dans le tableau



tache	ri	Ci	di
T1	0	1	5
T2	5	2	7
T3	0	2	5
T4	0	1	10
T5	0	3	12

Par application de l'algorithme EDF, on transforme les tâches dépendantes en tâches indépendantes, on obtient les nouvelles valeurs de ri et di

tache	Ri*	Ci	Di*
T1	0	1	3
T2	5	2	7
T3	1	2	5
T4	7	1	9
T5	8	3	12

Il nous reste qu'à appliquer maintenant EDF