

Résumé des symboles clés

B_i	Blocage de la tâche i , dû à des verrous, des entrées/ sorties, de l'inversion de priorités, <i>etc.</i>
C_i	WCET de la tâche i , typiquement obtenu de manière analytique (pensez à computing ou à calcul)
C_i^f	Coût associé au code de tolérance aux pannes (pensez à fault) pour la tâche i , ce qui tient compte entre autres de la gestion des cas d'exceptions
$C(k)$	Pire temps d'exécution dans la section critique k (ici, section critique est toute zone susceptible de provoquer un blocage en attente d'une ressource)
D_i	Échéance (pensez à deadline) pour démarrer la tâche sporadique i .
$hp(i)$	Ensemble des tâches $j: P_i < P_j$, donc de plus haute priorité que celle de la tâche i . En pratique, dans un SETR, seules ces tâches peuvent interférer avec la tâche i .
$hep(i)$	Ensemble des tâches $j: P_i \leq P_j$, donc de priorité supérieure ou égale (higher-than or equal) à celle de la tâche i . En pratique, dans un SETR, seules ces tâches peuvent interférer avec la tâche i .
I_i	Interférence maximale de la tâche i . Notez que la tâche la plus prioritaire n'en aura pas, alors que les autres en auront, par définition. Par I_i , on entend les cas où une tâche $j: P_i < P_j$ empêcherait, de par son exécution, l'ordonnancement de la tâche i
J_i	Ce qu'on nomme le Release Jitter de la tâche i , et qui intervient quand la tâche i est lancée par une tâche $j: j \neq i$ et qui représente le pire cas de latence entre la demande de lancement et le lancement effectif
N	Nombre de tâches à considérer (cardinalité de l'ensemble des tâches)
R_i	Temps maximal d'exécution (Release time) de la tâche i lorsque l'interférence I_i est considérée dans le calcul. On considère aussi parfois le blocage B_i
P_i	Priorité de la tâche i
T_i	Période (une bonne idée pour une mnémonique?) de la tâche i
T_f	Intervalle minimal entre deux pannes, lorsque du code de tolérance aux pannes doit être considéré
$usage(k, i)$	Fonction booléenne évaluée à 1 seulement si la ressource k est utilisée par au moins une tâche de priorité $j: P_i > P_j$