

CR-08 : Ordonnancement

Loris Marchal, CR CNRS au LIP

vague 1, 2?

Ordonnancement :

- ▶ une partie de la Recherche Opérationnelle
- ▶ allocation de **ressources** à des **tâches** sous certaines **contraintes** dans le but d'optimiser une **métrique** de performance

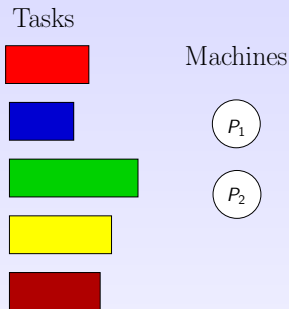
Contexte :

- ▶ systèmes informatiques
- ▶ plates-formes de calcul distribuées

Pré-requis (recommandé) :

- ▶ Algorithmes parallèles (cours de M1)

Exemple 1 :

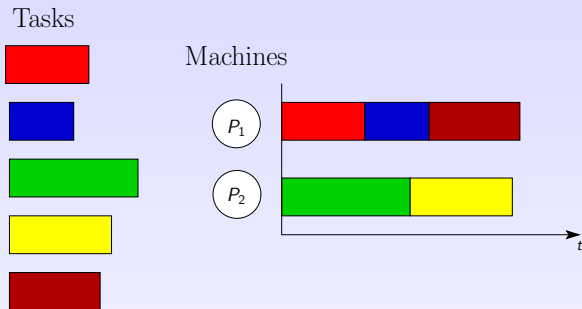


Objectifs :

- ▶ Minimiser le temps total d'exécution (C_{max})
- ▶ Minimiser le temps moyen d'exécution $\sum_i C_i$
- ▶ Idem avec pondérations $\sum_i w_i C_i$
- ▶ Avec des *deadlines*, minimiser le nombre de tâches en retard

Types de résultats étudiés : NP-complétude, algorithmes, algorithmes d'approximation, bornes d'(in-)approximation

Exemple 1 :

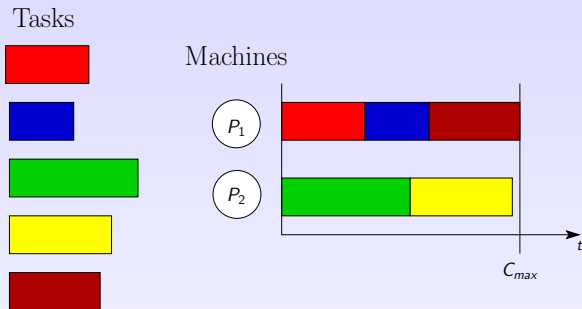


Objectifs :

- ▶ Minimiser le temps total d'exécution (C_{max})
- ▶ Minimiser le temps moyen d'exécution $\sum_i C_i$
- ▶ Idem avec pondérations $\sum_i w_i C_i$
- ▶ Avec des *deadlines*, minimiser le nombre de tâches en retard

Types de résultats étudiés : NP-complétude, algorithmes, algorithmes d'approximation, bornes d'(in-)approximation

Exemple 1 :

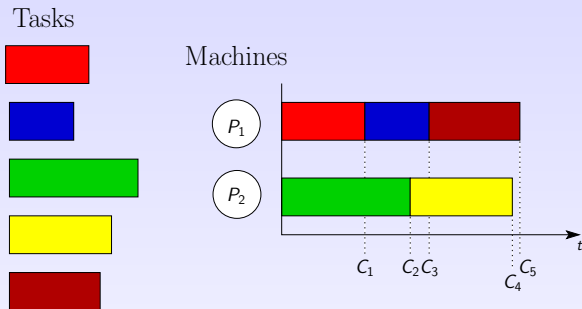


Objectifs :

- ▶ Minimiser le temps total d'exécution (C_{max})
- ▶ Minimiser le temps moyen d'exécution $\sum_i C_i$
- ▶ Idem avec pondérations $\sum_i w_i C_i$
- ▶ Avec des *deadlines*, minimiser le nombre de tâches en retard

Types de résultats étudiés : NP-complétude, algorithmes, algorithmes d'approximation, bornes d'(in-)approximation

Exemple 1 :

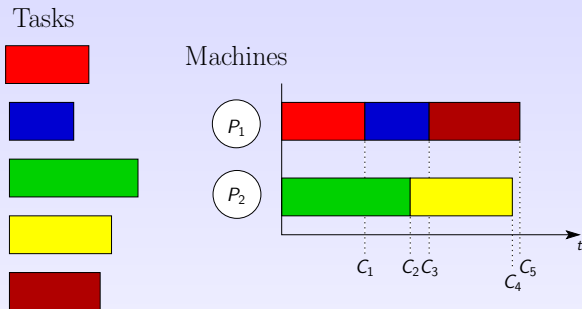


Objectifs :

- ▶ Minimiser le temps total d'exécution (C_{max})
- ▶ Minimiser le temps moyen d'exécution $\sum_i C_i$
- ▶ Idem avec pondérations $\sum_i w_i C_i$
- ▶ Avec des *deadlines*, minimiser le nombre de tâches en retard

Types de résultats étudiés : NP-complétude, algorithmes, algorithmes d'approximation, bornes d'(in-)approximation

Exemple 1 :

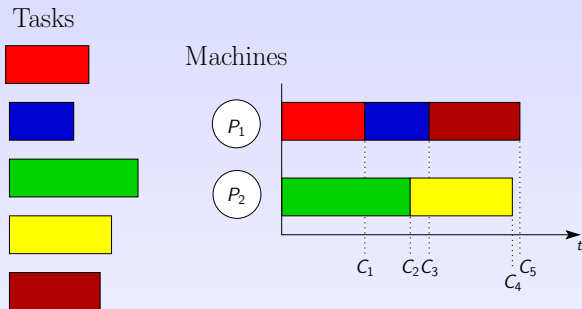


Objectifs :

- ▶ Minimiser le temps total d'exécution (C_{max})
- ▶ Minimiser le temps moyen d'exécution $\sum_i C_i$
- ▶ Idem avec pondérations $\sum_i w_i C_i$
- ▶ Avec des *deadlines*, minimiser le nombre de tâches en retard

Types de résultats étudiés : NP-complétude, algorithmes, algorithmes d'approximation, bornes d'(in-)approximation

Exemple 1 :

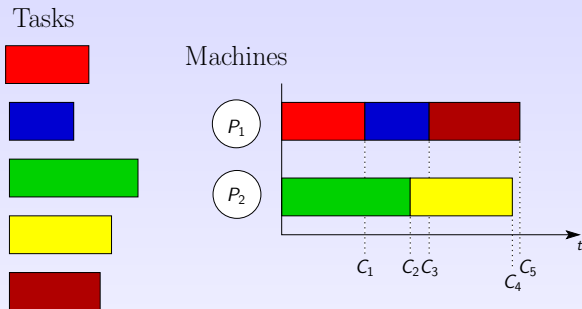


Objectifs :

- ▶ Minimiser le temps total d'exécution (C_{max})
- ▶ Minimiser le temps moyen d'exécution $\sum_i C_i$
- ▶ Idem avec pondérations $\sum_i w_i C_i$
- ▶ Avec des *deadlines*, minimiser le nombre de tâches en retard

Types de résultats étudiés : NP-complétude, algorithmes, algorithmes d'approximation, bornes d'(in-)approximation

Exemple 1 :

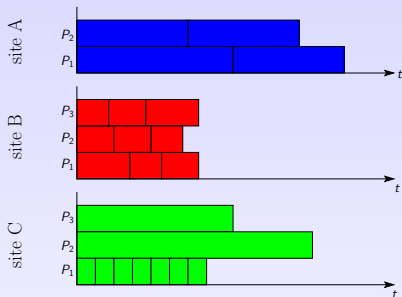


Objectifs :

- ▶ Minimiser le temps total d'exécution (C_{max})
- ▶ Minimiser le temps moyen d'exécution $\sum_i C_i$
- ▶ Idem avec pondérations $\sum_i w_i C_i$
- ▶ Avec des *deadlines*, minimiser le nombre de tâches en retard

Types de résultats étudiés : NP-complétude, algorithmes, algorithmes d'approximation, bornes d'(in-)approximation

Exemple 2 :



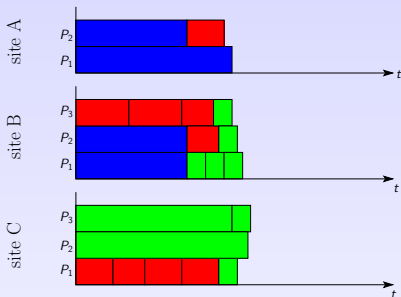
Contexte :

- ▶ Plusieurs organisations possèdent leur propre *cluster* de calcul
- ▶ Partager des ressources pour augmenter la performance globale
- ▶ Égoïsme : une organisation peut se retirer si la performance pour ses tâches décroît.

Types de résultats :

- ▶ Borne sur le rapport de performance avec/sans égoïsme
- ▶ Algorithmes d'approximation
- ▶ Utilisation de la théorie des jeux pour modéliser les organisations. . .

Exemple 2 :



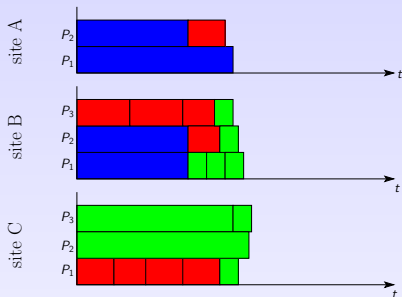
Contexte :

- ▶ Plusieurs organisations possèdent leur propre *cluster* de calcul
- ▶ Partager des ressources pour augmenter la performance globale
- ▶ Égoïsme : une organisation peut se retirer si la performance pour ses tâches décroît.

Types de résultats :

- ▶ Borne sur le rapport de performance avec/sans égoïsme
- ▶ Algorithmes d'approximation
- ▶ Utilisation de la théorie des jeux pour modéliser les organisations. . .

Exemple 2 :



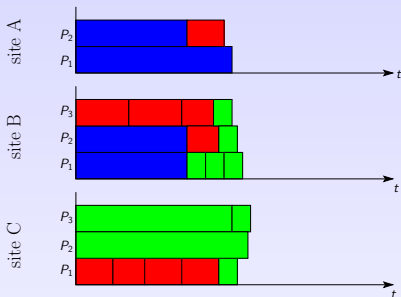
Contexte :

- ▶ Plusieurs organisations possèdent leur propre *cluster* de calcul
- ▶ Partager des ressources pour augmenter la performance globale
- ▶ Égoïsme : une organisation peut se retirer si la performance pour ses tâches décroît.

Types de résultats :

- ▶ Borne sur le rapport de performance avec/sans égoïsme
- ▶ Algorithmes d'approximation
- ▶ Utilisation de la théorie des jeux pour modéliser les organisations. . .

Exemple 2 :



Contexte :

- ▶ Plusieurs organisations possèdent leur propre *cluster* de calcul
- ▶ Partager des ressources pour augmenter la performance globale
- ▶ Égoïsme : une organisation peut se retirer si la performance pour ses tâches décroît.

Types de résultats :

- ▶ Borne sur le rapport de performance avec/sans égoïsme
- ▶ Algorithmes d'approximation
- ▶ Utilisation de la théorie des jeux pour modéliser les organisations. . .

Plan général :

1. Méthodes et résultats classiques (~ 3 cours)
 - ▶ Problèmes classiques (ordonnancement de liste, etc.)
 - ▶ Modélisation des applications
 - ▶ NP-complétudes, algorithmes d'approximation
 - ▶ Ordonnancement à la volée et non-clairvoyant
2. Meilleurs modèles pour les plates-formes de calcul (~ 4 cours)
 - ▶ Introduction des coûts de communication
 - ▶ Ordonnancement de tâches divisibles, en régime permanent,
 - ▶ Interférences calculs/communications.
3. Nouveaux objectifs et contraintes (~ 6 cours)
 - ▶ Tolérance aux pannes, robustesse, consommation d'énergie
 - ▶ Ordonnancement multi-organisation
 - ▶ Apports de la théorie des jeux
 - ▶ Méthodes stochastiques
 - ▶ Optimisation multi-critères

Organisation

- ▶ cours en français ou anglais (sur demande)
- ▶ début des cours : ~ 20 octobre

Évaluation :

- ▶ DM (1/3) : analyse d'article guidée (questionnaire), à l'écrit.
- ▶ Examen final (2/3) : analyse d'article (fourni), à l'écrit et présentation orale.

Plus d'infos :

- ▶ email : Loris.Marchal@ens-lyon.fr
- ▶ bureau 326, à côté de l'amphi A
- ▶ Déroulé du cours en 2011 :
graal.ens-lyon.fr/~lmarchal/scheduling/