

Recherches arborescente et locale pour les problèmes d'ordonnancement avec contraintes de précédence et temps de préparation

B. Gacias, C. Artigues et P. Lopez

LAAS-CNRS, Université de Toulouse, 31077 Toulouse CEDEX 4
email: {bgacias,artigues,lopez}@laas.fr

ROADEF 2008

- 1 Introduction
- 2 Position du problème
- 3 Recherche basée sur les divergences et recherche locale
- 4 Schéma de branchement et évaluation des nœuds
- 5 Résultats expérimentaux et conclusions

- 1 Introduction
- 2 Position du problème
- 3 Recherche basée sur les divergences et recherche locale
- 4 Schéma de branchement et évaluation des nœuds
- 5 Résultats expérimentaux et conclusions

- Problème à machines parallèles
- Résolution par méthodes de recherche arborescente et locale [Néron et al., 2006]
- Nous traitons le problème avec temps de préparation et contraintes de précédence entre opérations
- Algorithme de liste [Hurink et Knust, 2001]

- 1 Introduction
- 2 Position du problème**
- 3 Recherche basée sur les divergences et recherche locale
- 4 Schéma de branchement et évaluation des nœuds
- 5 Résultats expérimentaux et conclusions

Position du problème

Données du problème

Variables de décision

Fonctions Objectif

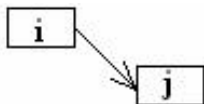
Données du problème

- Tâches : $J = \{1, \dots, n\}$
- Ressources : $M = \{M_1, \dots, M_m\}$
- Pour chaque tâche i : p_i, r_i, d_i
- Contraintes de précédence
- Temps de préparation s_{ij}

Position du problème

Données du problème

- Tâches : $J = \{1, \dots, n\}$
- Ressources : $M = \{M_1, \dots, M_m\}$
- Pour chaque tâche i : p_i, r_i, d_i
- Contraintes de précédence

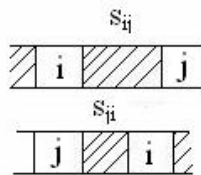


- Temps de préparation s_{ij}

Position du problème

Données du problème

- Tâches : $J = \{1, \dots, n\}$
- Ressources : $M = \{M_1, \dots, M_m\}$
- Pour chaque tâche i : p_i, r_i, d_i
- Contraintes de précédence
- Temps de préparation s_{ij}



Variables de décision

- S_i et C_i

Fonctions Objectif

- $\min \sum C_i$
- $\min L_{\max}$

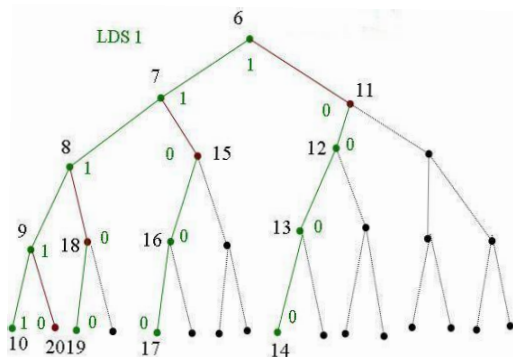
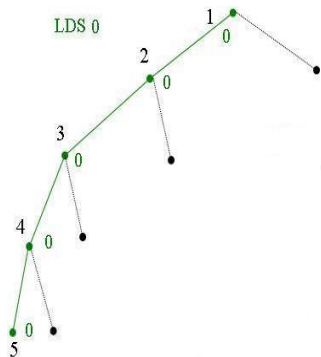
Plan

- 1 Introduction
- 2 Position du problème
- 3 Recherche basée sur les divergences et recherche locale**
- 4 Schéma de branchement et évaluation des nœuds
- 5 Résultats expérimentaux et conclusions

Recherche basée sur les divergences

Limited Discrepancy Search (LDS) [Harvey et Ginsberg, 1995]

- Méthode de recherche arborescente
- Ordre de recherche ; divergences à partir de l'heuristique initiale



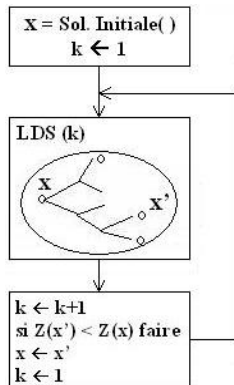
Variantes pour l'amélioration de l'efficacité de la méthode :

- Improved LDS (ILDS) [Korf, 1996]
- Depth-bounded Discrepancy Search (DDS) [Walsh, 1997]
- Discrepancy-bounded Depth-First Search (DDFS) [Beck et Perron, 2000]
- Yet Improved LDS (YIELDS) [Karoui et al., 2007]

Recherche locale basée sur LDS

Climbing Discrepancy Search (CDS) [Milano et Roli, 2002]

- Traiter l'exploration du voisinage avec LDS
- Les divergences sont définies par rapport à la solution x



CDDS : le voisinage est limité aussi par le niveau de profondeur [Ben Hmida et al., 2007]

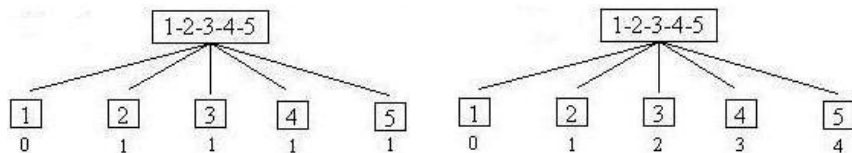
Nous proposons deux méthodes :

- Hybrid Discrepancy CDDS

- 1 Recherche CDS jusqu'à k divergences
- 2 $k \leftarrow k + 1$
- 3 Recherche CDDS, on restreint le voisinage

- Mix Counting CDS

- Comptages de divergences différents \rightarrow Niveau de profondeur

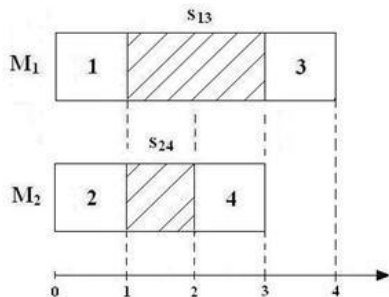


Plan

- 1 Introduction
- 2 Position du problème
- 3 Recherche basée sur les divergences et recherche locale
- 4 Schéma de branchement et évaluation des nœuds**
- 5 Résultats expérimentaux et conclusions

Définition de l'arbre de recherche

- Le problème ne peut pas être résolu efficacement par un algorithme de liste



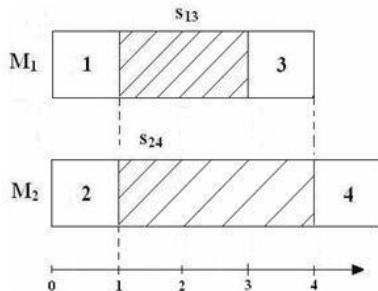
$$\begin{pmatrix} 0 & 10 & 2 & 10 \\ 10 & 0 & 0 & 1 \\ 10 & 10 & 0 & 10 \\ 10 & 10 & 10 & 0 \end{pmatrix}$$

$$\pi = (1, 2, 4, 3)$$

$$\pi' = (2, 1, 4, 3)$$

Définition de l'arbre de recherche

- Contrainte de précédence $3 \prec 4$



$$\begin{pmatrix} 0 & 10 & 2 & 10 \\ 10 & 0 & 0 & 1 \\ 10 & 10 & 0 & 10 \\ 10 & 10 & 10 & 0 \end{pmatrix}$$

$$\pi = (1, 2, 4, 3)$$

$$\pi' = (2, 1, 4, 3)$$

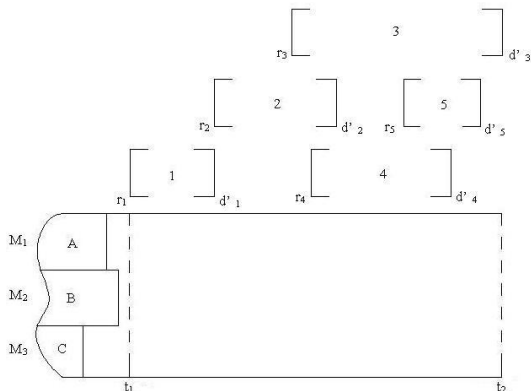
- π et π' ne respectent pas la contrainte de précédence
- Enumération complète : meilleur ordre d'opérations et meilleure allocation des ressources

Heuristiques d'initialisation

- Sélection des opérations
 - SPT, Shortest Processing Time
 - EDD, Earliest Due Date
- Affectation des opérations
 - ECT, Earliest Completion Time

Evaluation des nœuds

- $\min \sum C_i$: Borne inférieure [Chu et al., 2005]
- $\min L_{\max}$: Raisonnement énergétique
 - Dans un intervalle $\Delta = [t_1, t_2] \rightarrow E_{requis} + C_{setup} \leq E_{dispo}$
 - $[r_i, d'_i]$, fenêtres de temps pour chaque opération qu'il reste à affecter

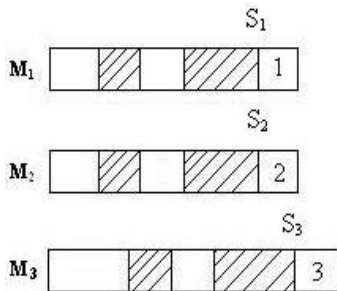


$$E_{dispo} = m \times (t_2 - t_1)$$
$$E_{requis} = \sum_i E_{min}, i \in F$$
$$C_{setup} \rightarrow k - m \text{ plus petits } s_{ij}, i, j \in F$$

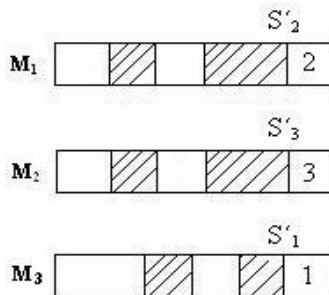
Evaluation des nœuds

Règles de dominance pour la recherche locale

- Trouver une affectation partielle dominante ($S'_i < S_i$ et $S'_j \leq S_j$ pour toutes les tâches du *Front*)
- La liste qui amène à cette affectation partielle doit respecter les contraintes de précédence et le nombre de divergences autorisées



Nœud évalué



Affectation partielle dominante

Plan

- 1 Introduction
- 2 Position du problème
- 3 Recherche basée sur les divergences et recherche locale
- 4 Schéma de branchement et évaluation des nœuds
- 5 Résultats expérimentaux et conclusions

Résultats expérimentaux

Comparaison pour le problème $P|r_i, q_i|C_{\max}$ [Néron et al.]

$T_{\max} = 30$ s	Meilleure Solution	Meilleure Sol. Strict	CPU(s)
$LDS_{z=1}^{TW}$	0	1	29.64
$LDS_{z=2}^{CHR}$	0	7	28.40
$BS_{w=3}^{TW}$	25	3	20.37
$BS_{w=4}^{CHR}$	22	0	28.40
<i>CDS</i>	35	6	30 (8.03)
<i>HD-CDDS</i>	38	9	30 (7.02)

Résultats expérimentaux

Evaluation borne, raisonnement énergétique et règles de dominance

$\sum C_i$	CPU(s) Meilleure Sol.	
<i>Sans Lb</i>	258	45.00 %
<i>Lb</i>	251	55.00 %
<i>Lb + RD</i>	214	90.00 %

L_{\max}	CPU(s) Meilleure Sol.	
<i>Sans NRJ</i>	256	68.33 %
<i>NRJ</i>	250	86.67 %
<i>NRJ + RD</i>	232	93.33 %

Résultats expérimentaux

- Evaluation du critère *ECT*
 - On trouve la solution optimale pour 85 % des instances
 - Différencier les divergences dans l'ordre des tâches des divergences d'affectation aux machines
- Comparaison entre les méthodes de recherche locale

$\sum C_i$	Meilleure Solution	Ecart moyen
<i>CDS</i>	12 (20.00 %)	1.36 %
<i>CDDS</i>	2 (3.33 %)	4.83 %
<i>HD-CDDS</i>	27 (45.00 %)	0.89 %
<i>MC-CDS</i>	40 (66.67 %)	1.02 %

L_{\max}	Meilleure Solution	Ecart moyen
<i>CDS</i>	18 (30.00 %)	2.53 %
<i>CDDS</i>	0 (0.00 %)	11.62 %
<i>HD-CDDS</i>	36 (60.00 %)	1.84 %
<i>MC-CDS</i>	34 (56.67 %)	1.69 %

Conclusions

- Les méthodes hybrides de recherche locale avec LDS s'avèrent efficaces pour les problèmes à machines parallèles
- Apports dans le traitement des nœuds : règles de dominance adaptées à la recherche locale et introduction des temps de préparation dans l'approche énergétique