

Partitioning of Embedded Applications with Variable Distribution of Calculation on a Reconfigurable Architecture

Fakhreddine GHAFFARI

{Email: ghaffari@i3s.unice.fr}

Michel AUGUIN

Maher BENJEMAA

Mohamed ABID



SSD 2003



Plan de l'Exposé

- Introduction
- Approche de partitionnement
- Exemple d'application
- Résultats de travaux
- Perspectives
- Conclusion

Introduction (1)

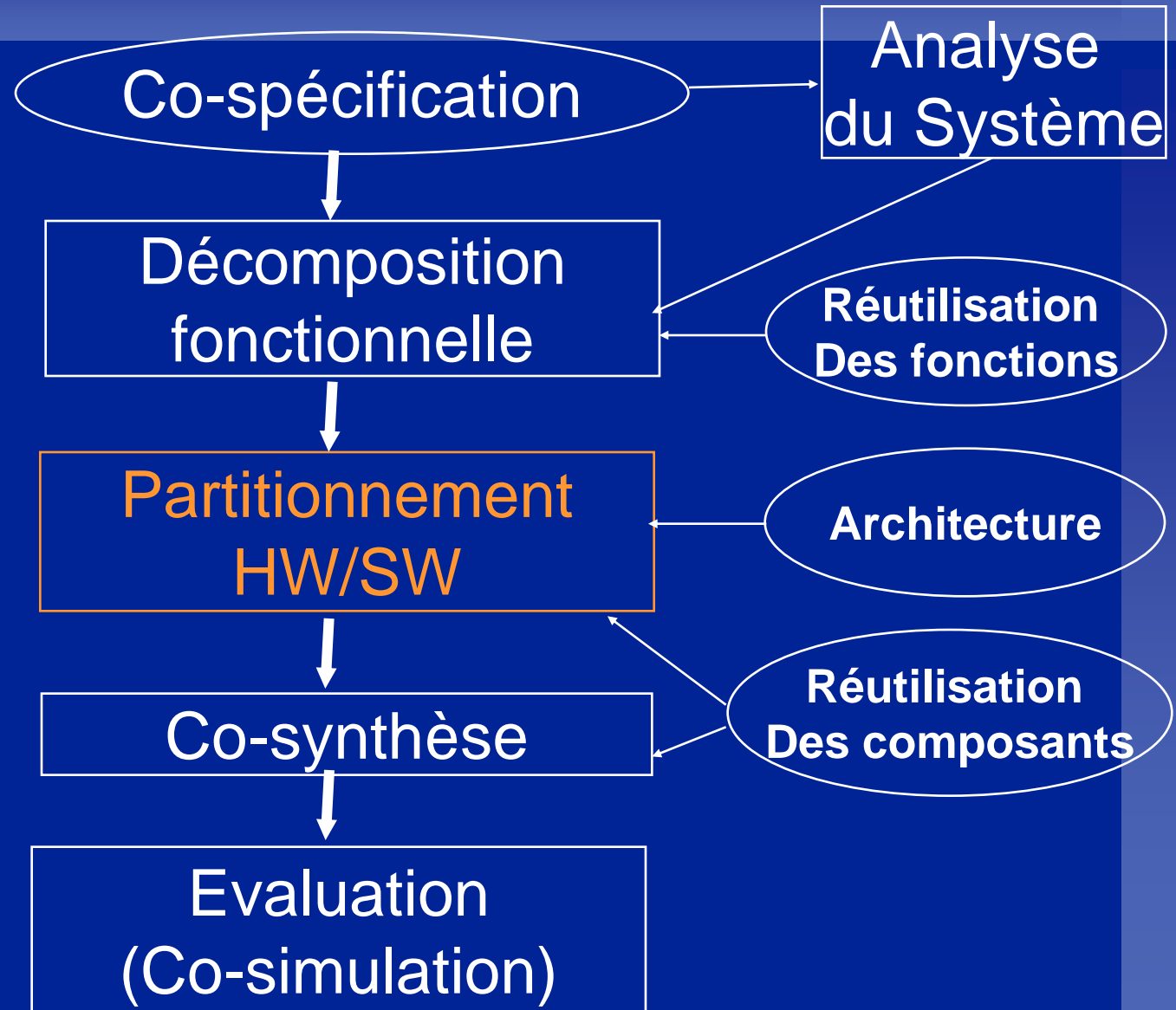
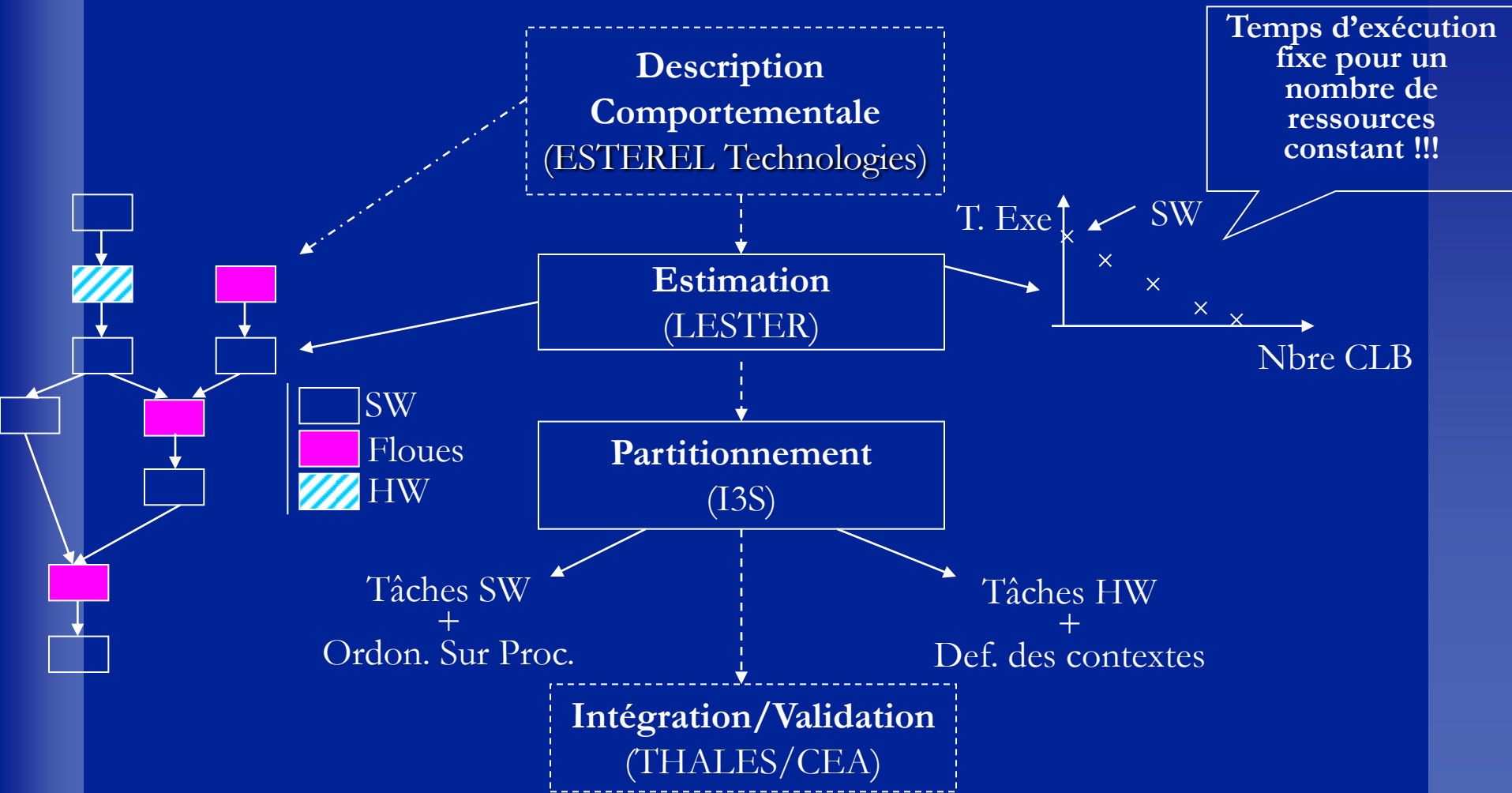


Fig. 1 : Processus d'exploration de l'espace de conception dans le codesign

Introduction (2)

● Organigramme du projet *EPICURE*



Introduction (3)

● Architecture cible : hétérogène

- Architecture bien adaptée au domaine d'applications
 - ❖ Parallélisme de traitement
 - ❖ Débit important d'échanges avec la mémoire
- Architecture permettant d'avoir une solution efficace en coût et performance
- Architecture assez générale, flexible et reconfigurable
- FPGAs : traitement sur mesure

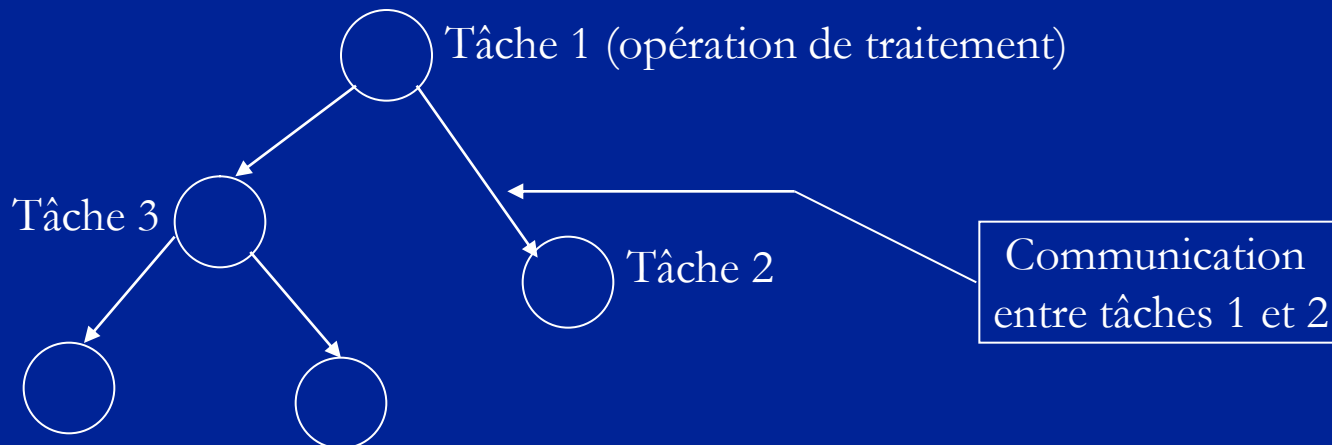


Fig 2 : Système reconfigurable à interface externe

Approche de partitionnement (1)

- **Modèle choisi**

- Les applications de traitement d'images sont modélisées par des DFGs
- niveau de granularité choisi : gros grain (niveau fonction)
- le modèle permet d'estimer le temps d'exécution et de communication



Approche de partitionnement (1)

● Objectif de partitionnement



Problème à résoudre : Partitionnement des tâches dont le temps d'exécution est variable en fonction du jeu de données

Exemple de fonctions : extraction du contours, étiquetages des objets dans une image....

Approche de partitionnement (2)

- Profiling d'une application et définition des paramètres de corrélation

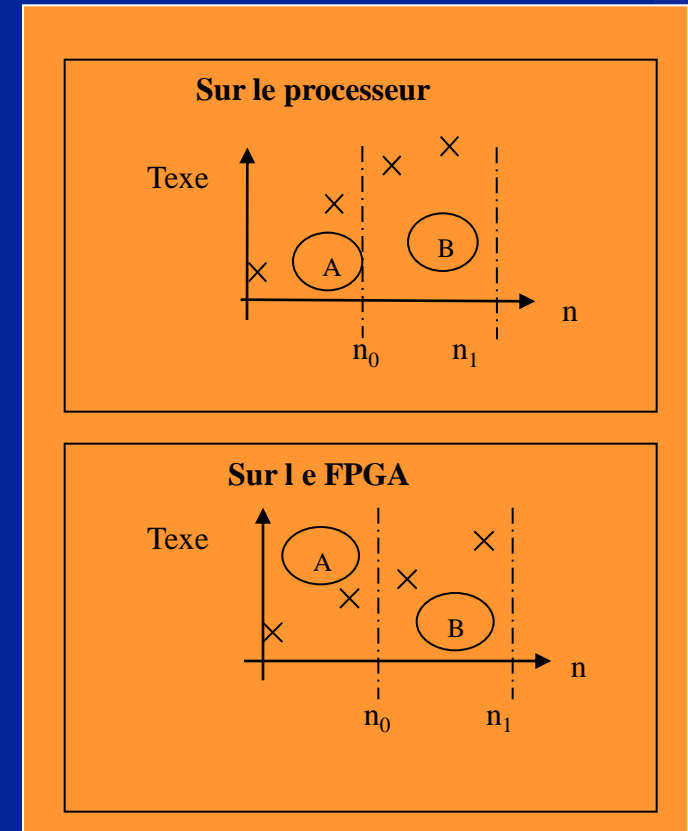
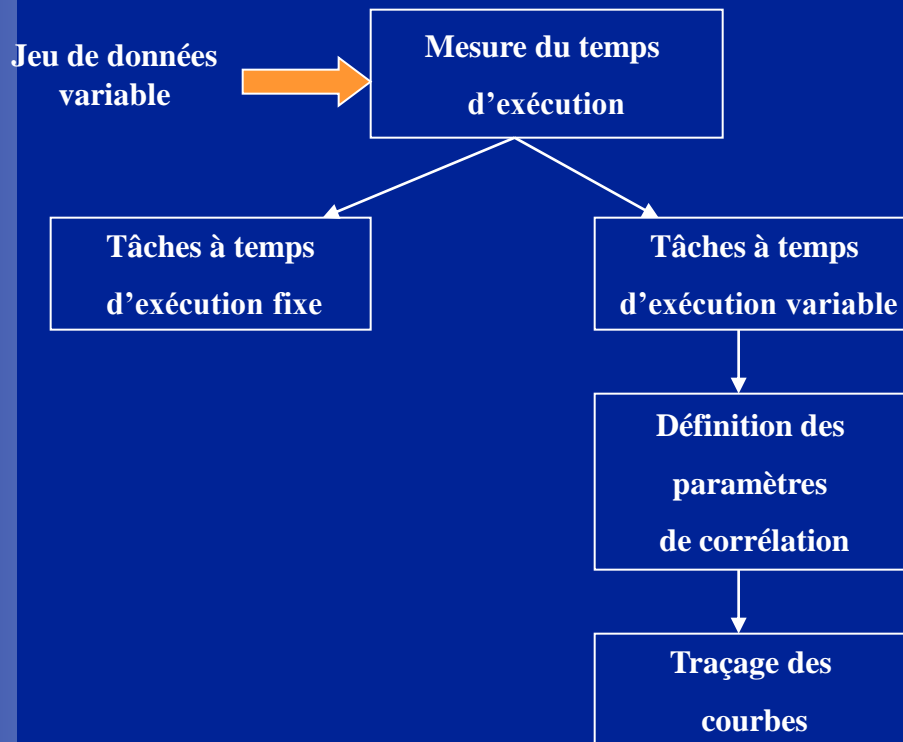


Fig2 : courbes du temps d'exécution en fonction du nombre d'objets dans l'image

Approche de partitionnement (3)

- Graphe de flot de données conditionné

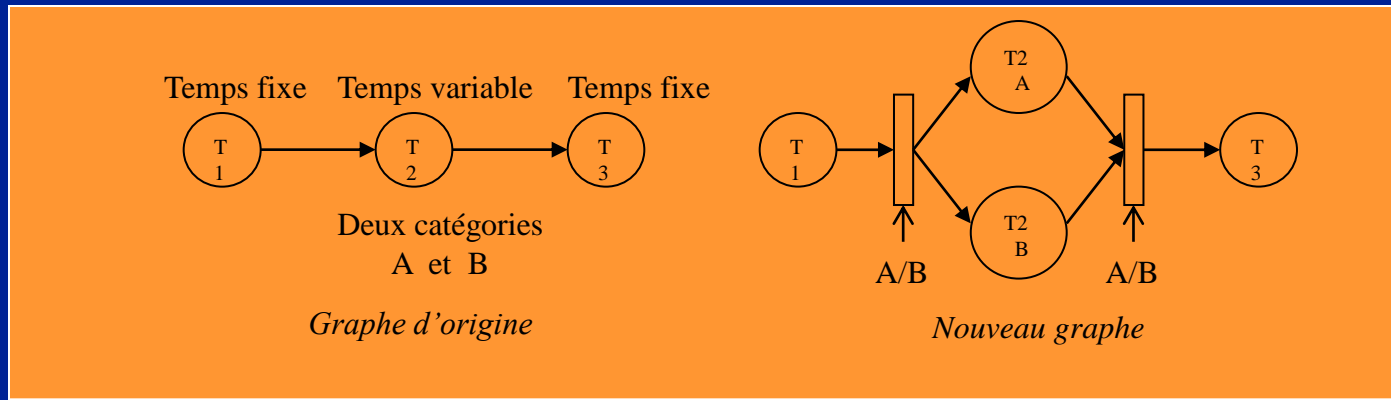


Fig 3 : Construction d'un graphe conditionné

- Combinaisons résultats

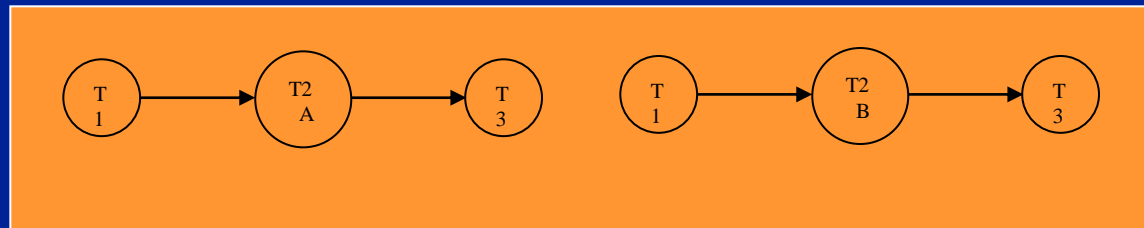
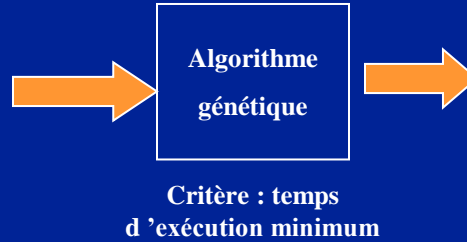


Fig 4 : Les deux configurations possibles du graphe conditionné de la fig 3

Approche de partitionnement (4)

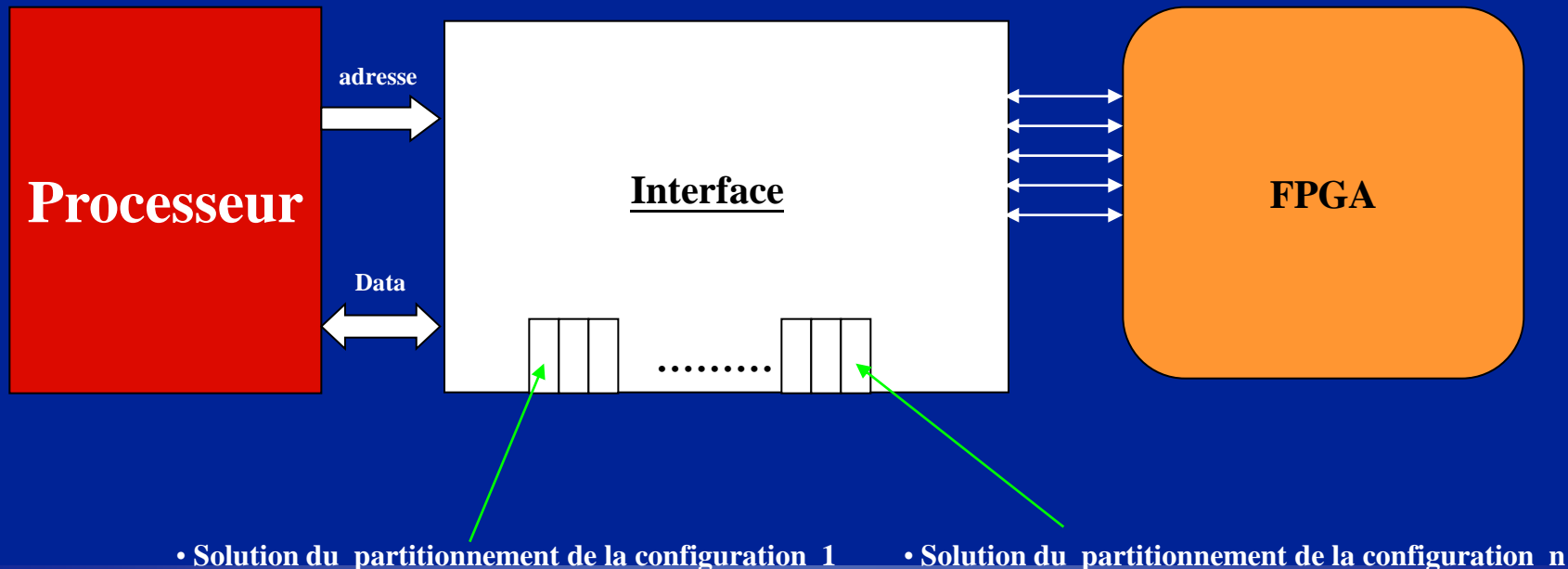
● Partitionnement

- Couples (Texte, Nbre CLB)
- Graphe de la configuration
- Caractéristiques de l'architecture
- Caractéristiques de bus, mémoires ...



- Solution du partitionnement de la configuration 1
- Solution du partitionnement de la configuration 2
-
-
- Solution du partitionnement de la configuration n

● Idée sur la réalisation : travaux en cours au CEA



Exemple d'application (1)

- Exemple d'application : Détection de mvt sur un fond d'image fixe



Image courante

Détection de mouvement



Image de référence

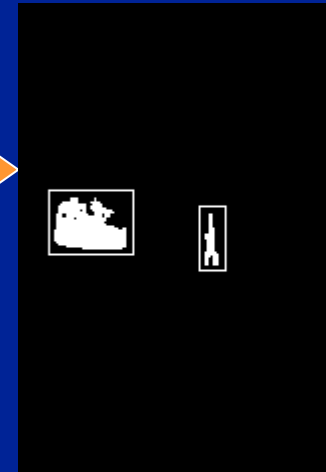
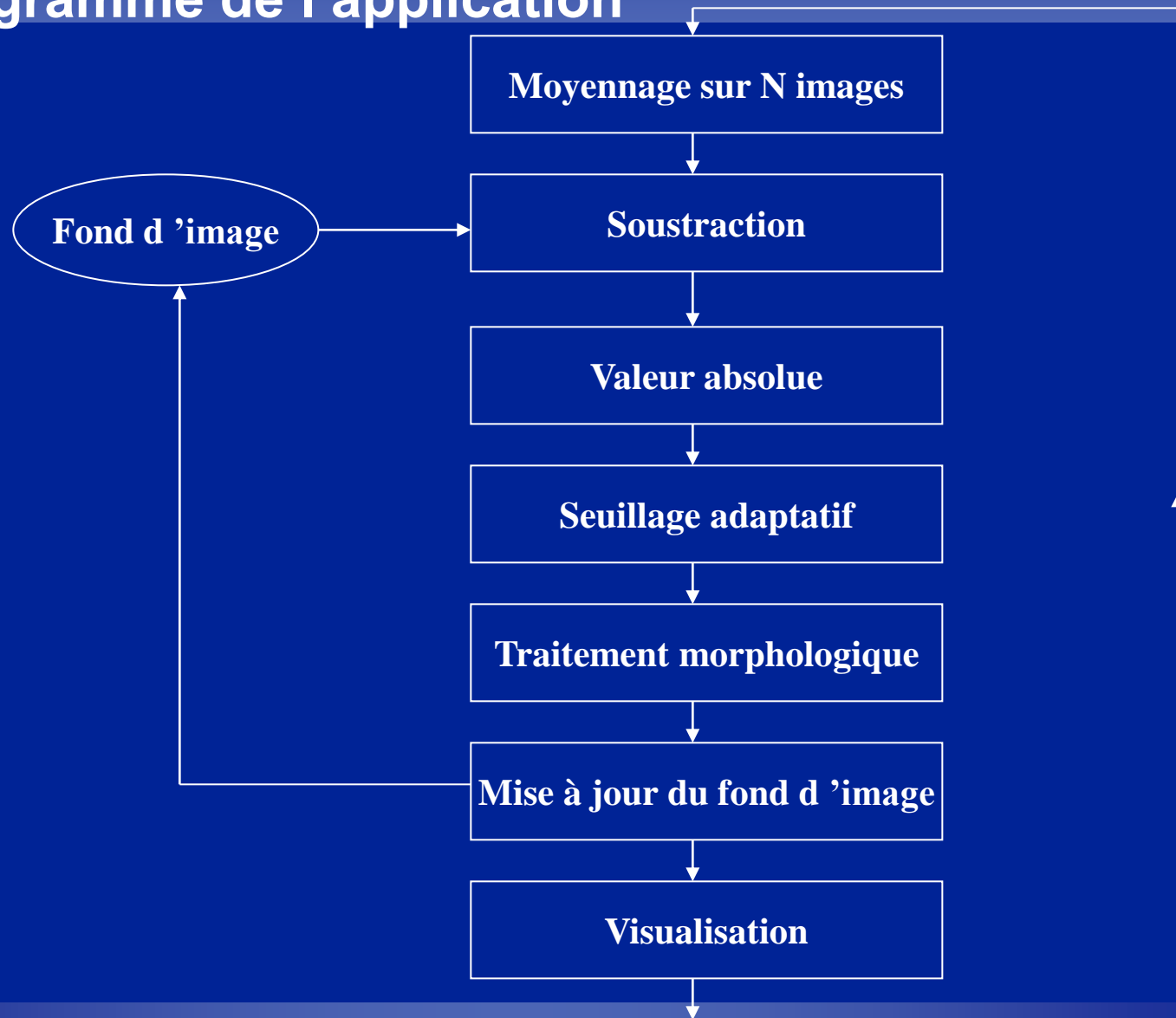


Image résultat

Exemple d'application (2)

- Organigramme de l'application



Exemple d'application (3)

● Exemple d'application : Détection de mvt sur un fond d'image fixe

➤ Traitement des résultats de Profiling

• Lors du choix des seuils nous tenons compte de :

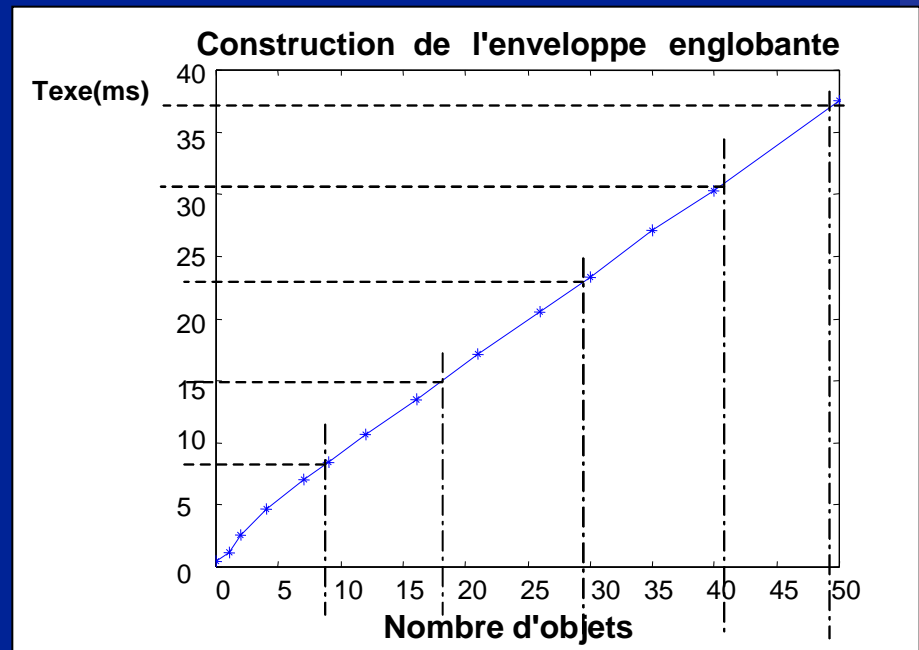
✎ les valeurs de la pente

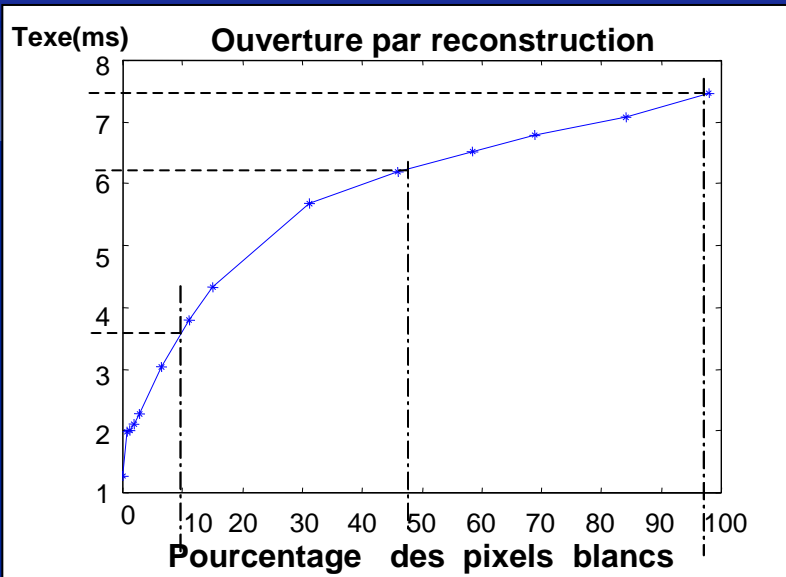
✎ nombre de configurations possibles à considérer

(temps de conception total, limite de l'espace mémoire)

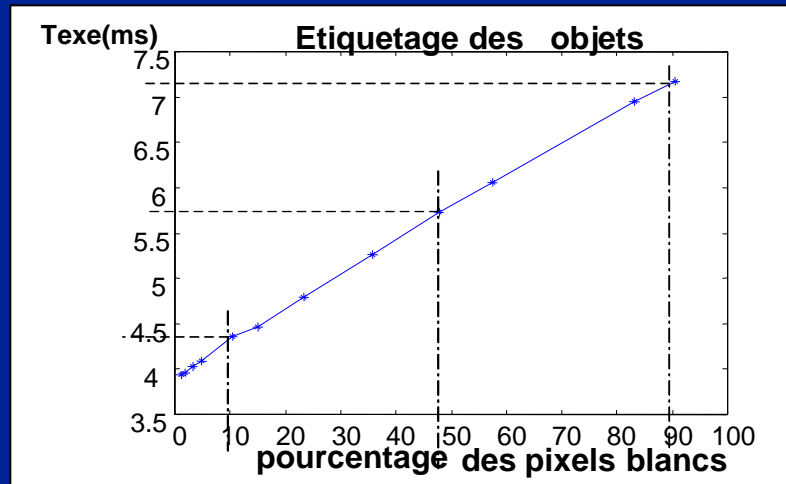
✎ les seuils sont identiques

pour le processeur et le FPGA





✂ le temps d'exécution affecté à chaque catégorie est le max des temps d'exécution de cette catégorie



✂ les fonctions qui ont le même paramètre de corrélation : nous choisissons les mêmes intervalles de catégories.

Exemple d'application (4)

● DFG conditionné de l'application

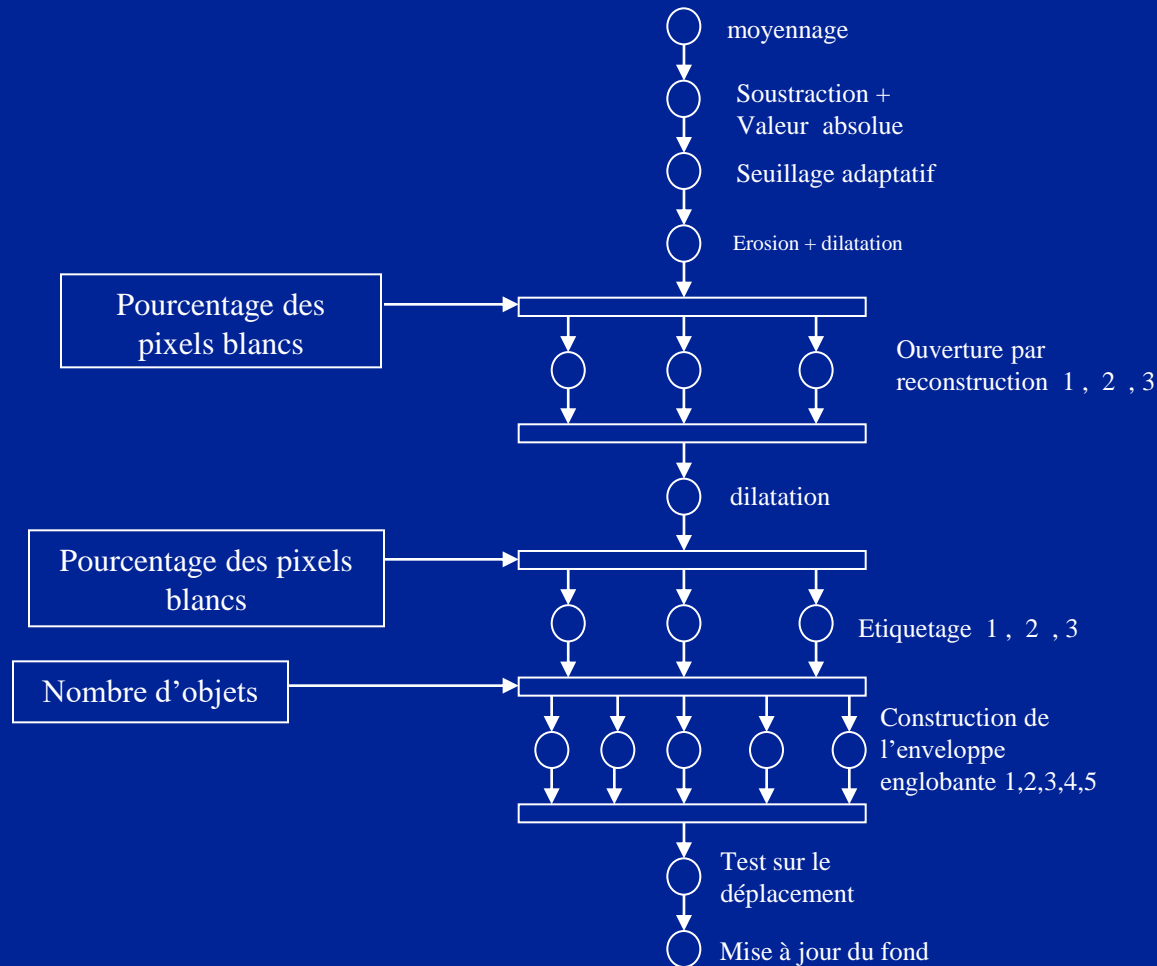
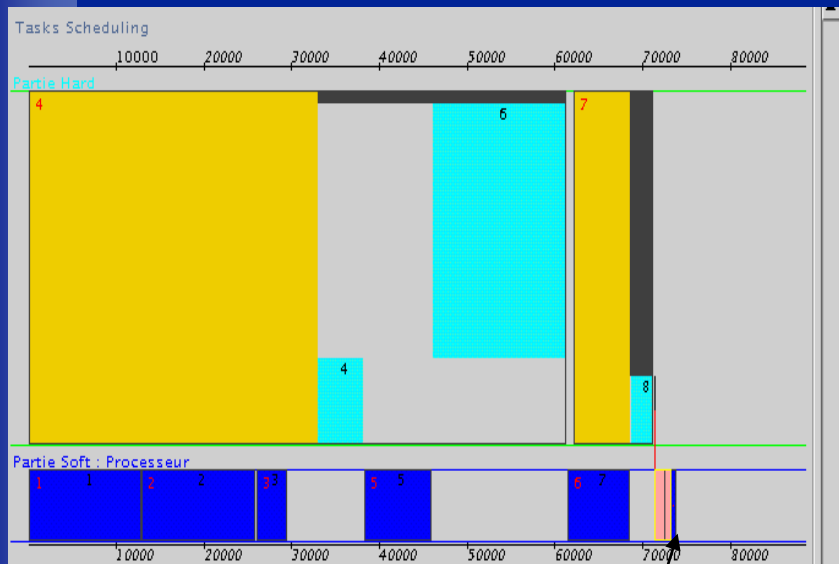


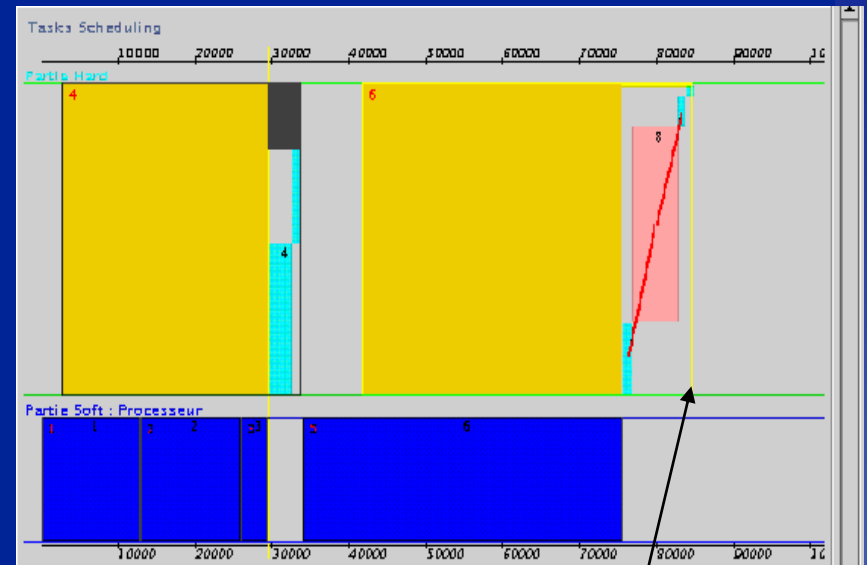
Fig 5 : DFG avec tâches à temps d'exécution variable

Résultats de travaux (1)

- Résultats de l'outil du partitionnement



Texe total = 72.455 ms



Texe total = 82.659 ms

Figure 13 : Résultats du partitionnement de la configuration 3

Figure 14 : Résultats du partitionnement de la configuration 6

Résultats de travaux (2)

- Temps d'exécution total

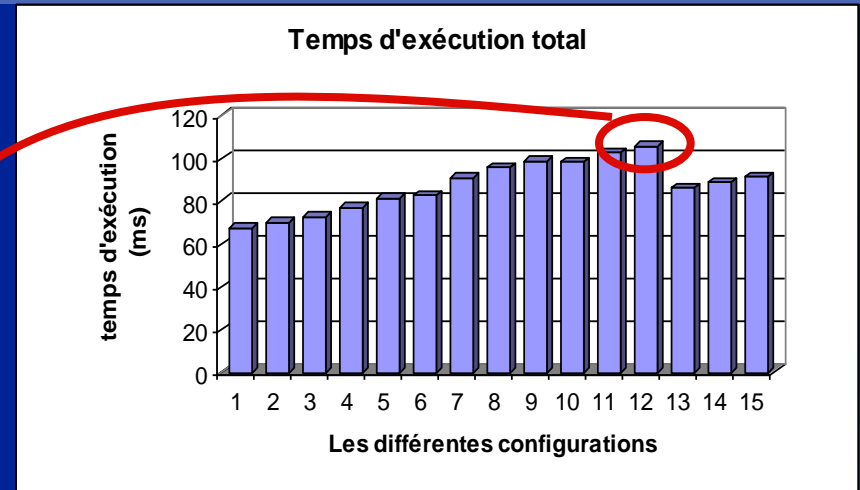


Fig 15: Temps d'exécution total de toutes les configurations

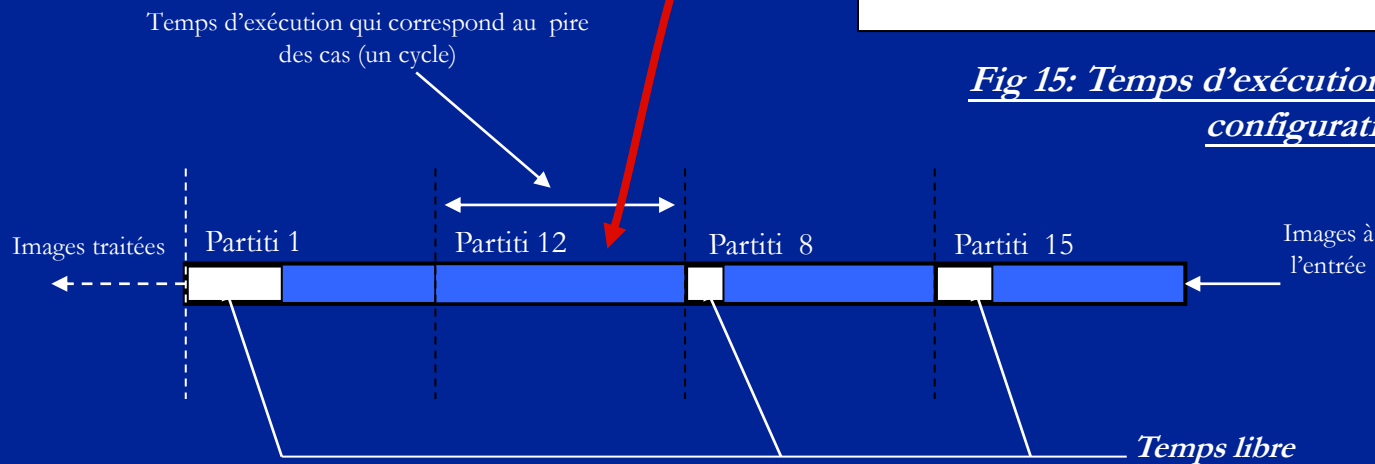


Fig 16 : Principe du temps libre pendant l'exécution de l'application

Résultats de travaux (3)

- Temps de communication

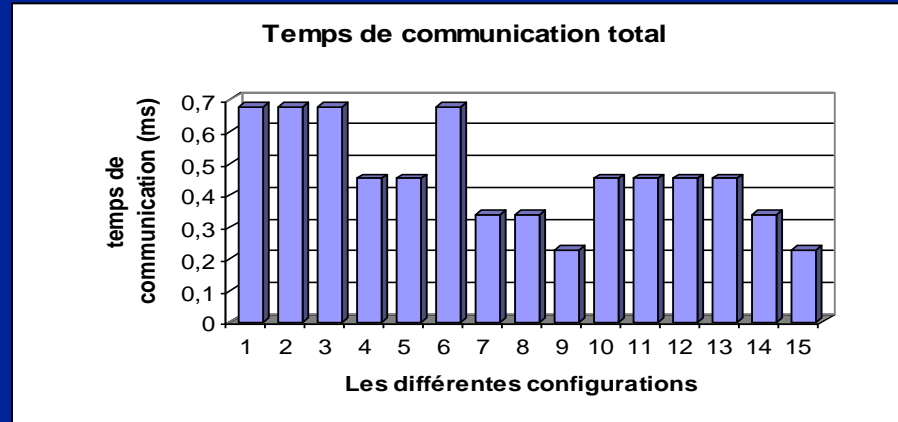


Fig 17: Temps de communication total pour toutes les configurations

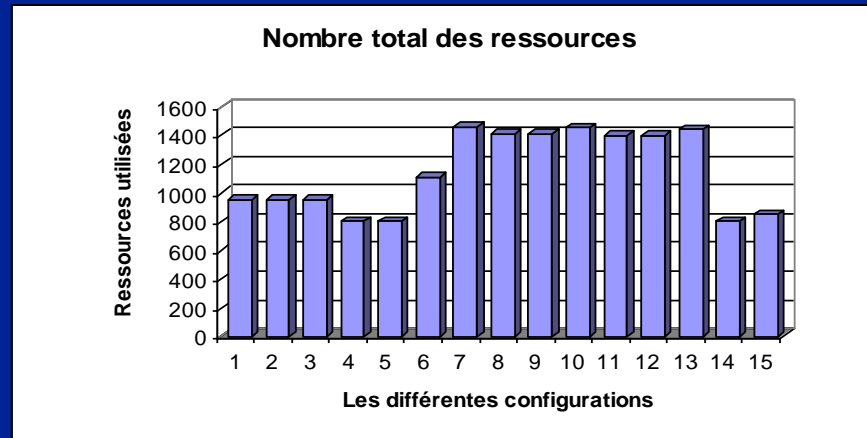


Fig 18: Nombre des ressources utilisées pour toutes les configurations

- Utilisation des ressources matérielles

Perspectives (1)

● Optimisation des résultats de partitionnement

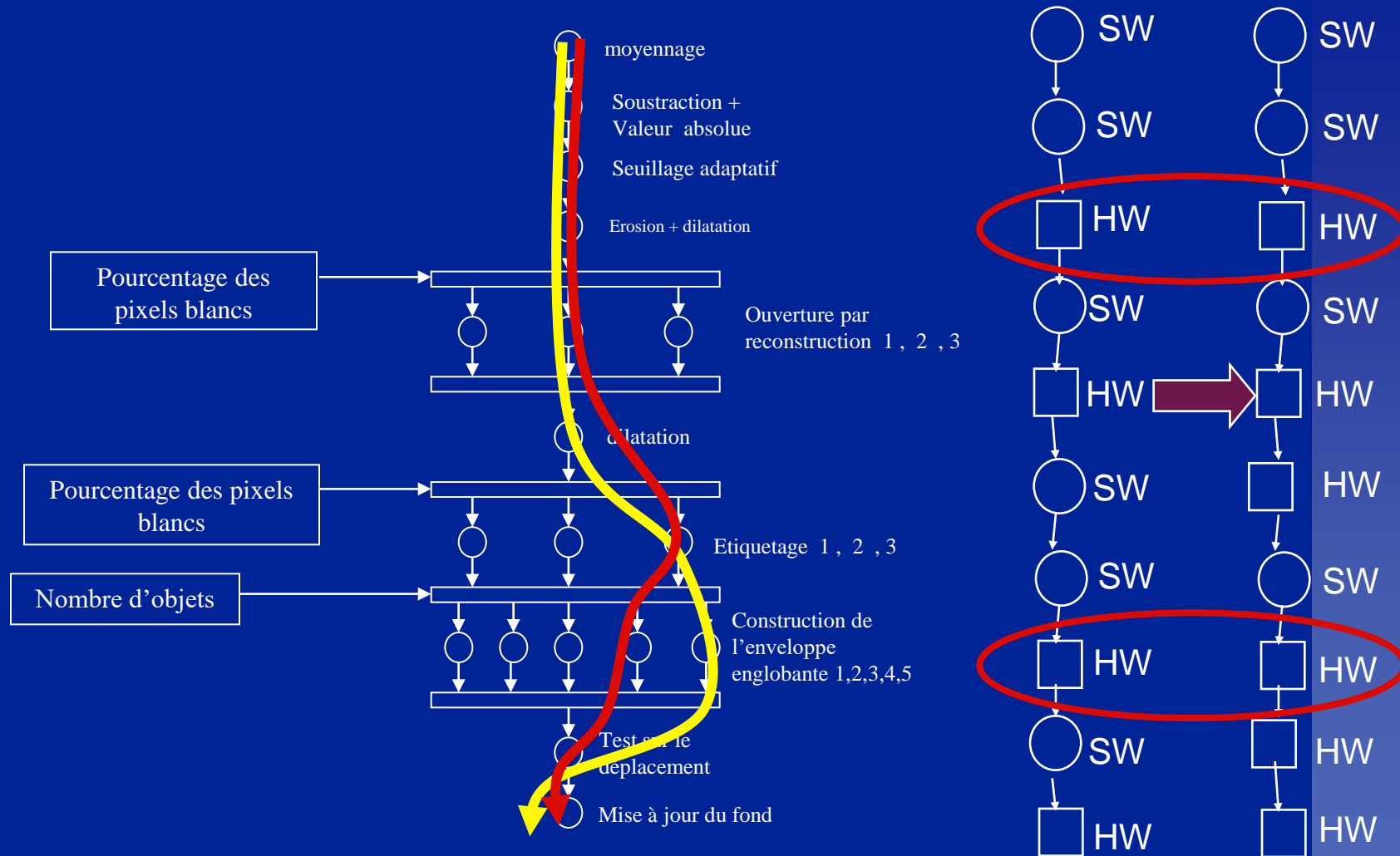


Fig 19 : DFG avec tâches à temps d'exécution variable

Perspective (2)

- Ordonnancement sur l'architecture cible

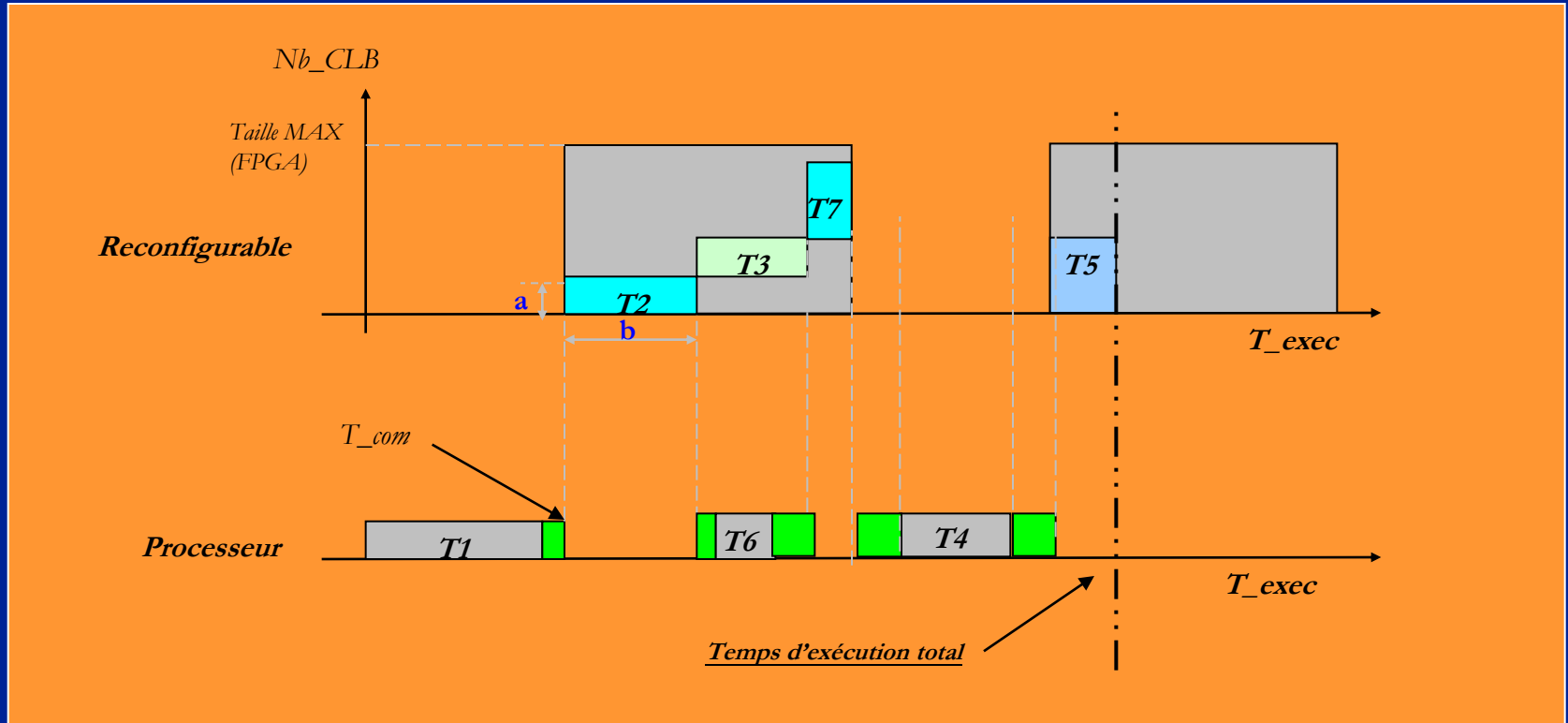
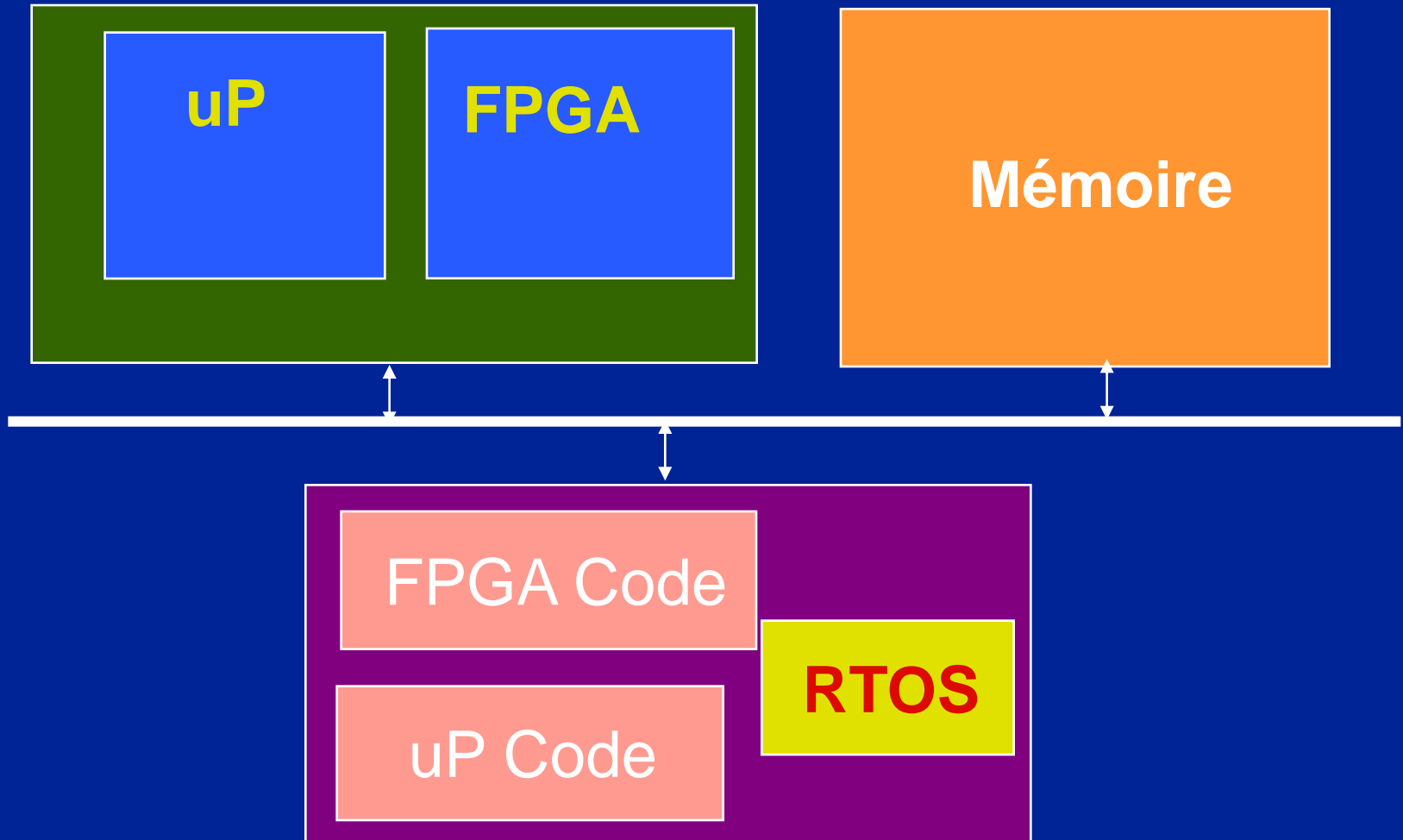


Fig 10: exemple d'ordonnement d'une solution sur l'architecture

Perspectives (3)

- Utilisation d'un RTOS



Conclusion

- ✓ L'approche permet d'avoir une meilleure exploitation des ressources de la composante matérielle de l'architecture
- ✓ Les résultats du profiling permettent de donner des configurations qui réduisent le pessimisme du cas pire

Pour un certains jeu des données nous gagnons :

- un temps libre que nous pouvons exploiter en optimisant d'autres critères tel que la consommation
- une diminution de temps de communication