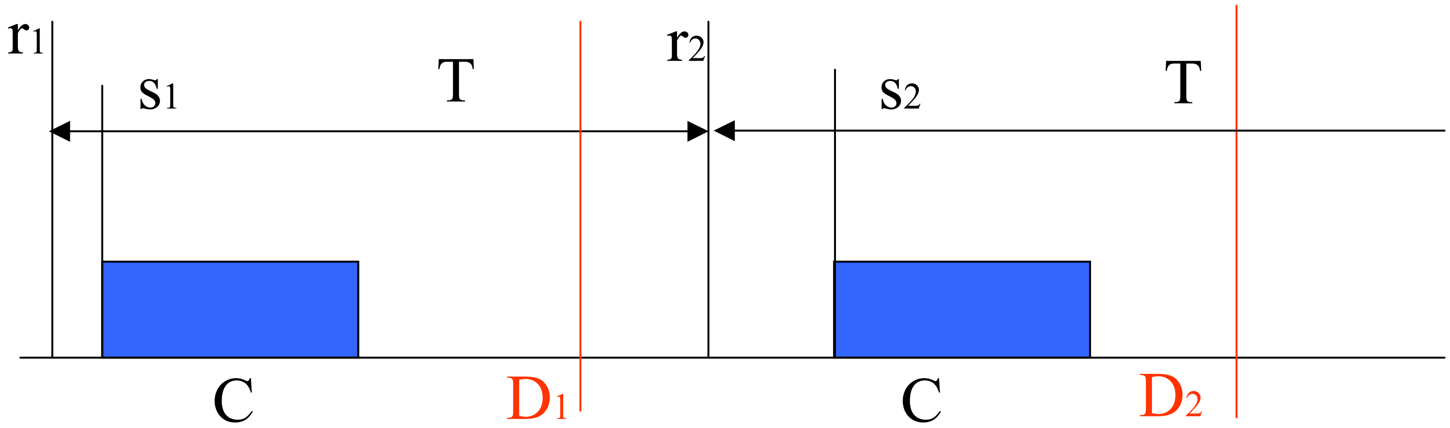
Ordonnancement non préemptif et condition d'ordonnançabilité pour systèmes embarqués à contraintes temps réel

Liliana CUCU, Yves SOREL, projet AOSTE
INRIA Rocquencourt

Plan de l’exposé

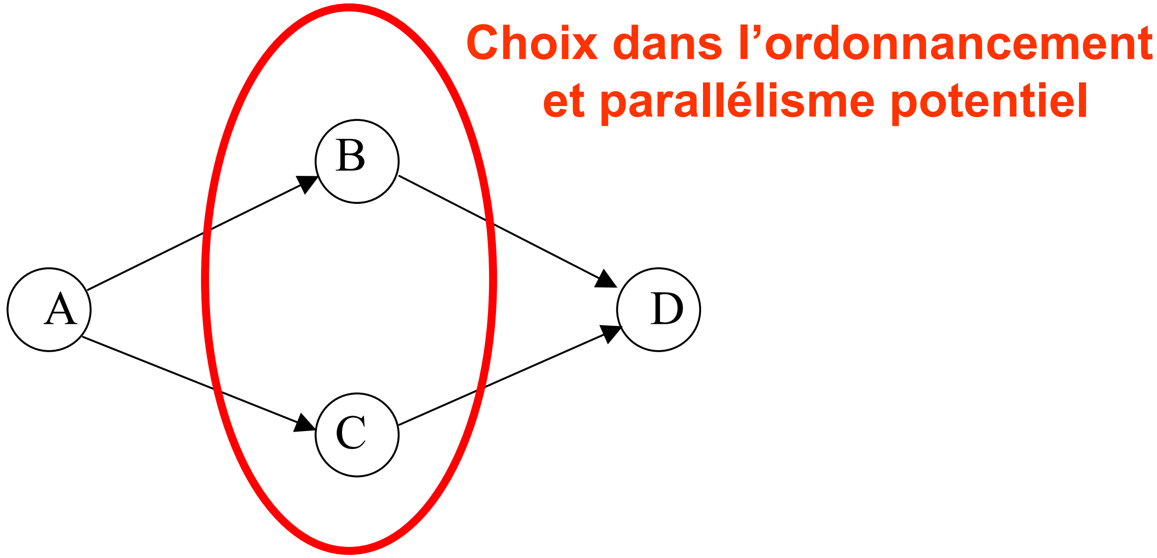
- Etat de l’art
- Modèle proposé
- Ordonnancement et ordonnançabilité
- Conclusion et perspectives

Modèle classique : contraintes temps réel



- Période : T
- Echéance : D
- Durée d'exécution : C
- Date de réveil : r
- Date de début : s

Modèle classique : contraintes de précédences



Tâche

Précédence

Graphe orienté acyclique

En-ligne vs. hors-ligne / Statique vs. dynamique

Quand est réalisé l'ordonnancement ?

- En-ligne
  - pendant l'exécution
- Hors-ligne
  - avant l'exécution

Quel type de choix fait l'algorithme d'ordonnancement ?

- Statique
  - le choix dépend de l'ensemble total fixe des tâches et de l'ensemble des tâches disponibles
- Dynamique
  - le choix dépend uniquement de l'ensemble des tâches disponibles

Etat de l'art : contraintes de périodicités

Processeurs / caractéristiques / but

- 1 / T = D / - RMS : optimal dans le cas statique
- 1 / T ≤ D / - DM : optimal dans le cas dynamique
- 1 / T ≤ D, r / - NP-difficile (non préemptif)

Etat de l'art : contraintes de précédences

- 1 / préc, D / min L_{max} EDD – optimal
- 1 / préc, r, D / - NP-difficile
- 1 / préc / f_{max} Lawler – optimal
- 1 / préc, r / f_{max} Baker – $O(n^2)$ (préemptif)
- 1 / préc, r / min L_{max} NP-difficile

Etat de l’art : contraintes de précédences et de périodicités

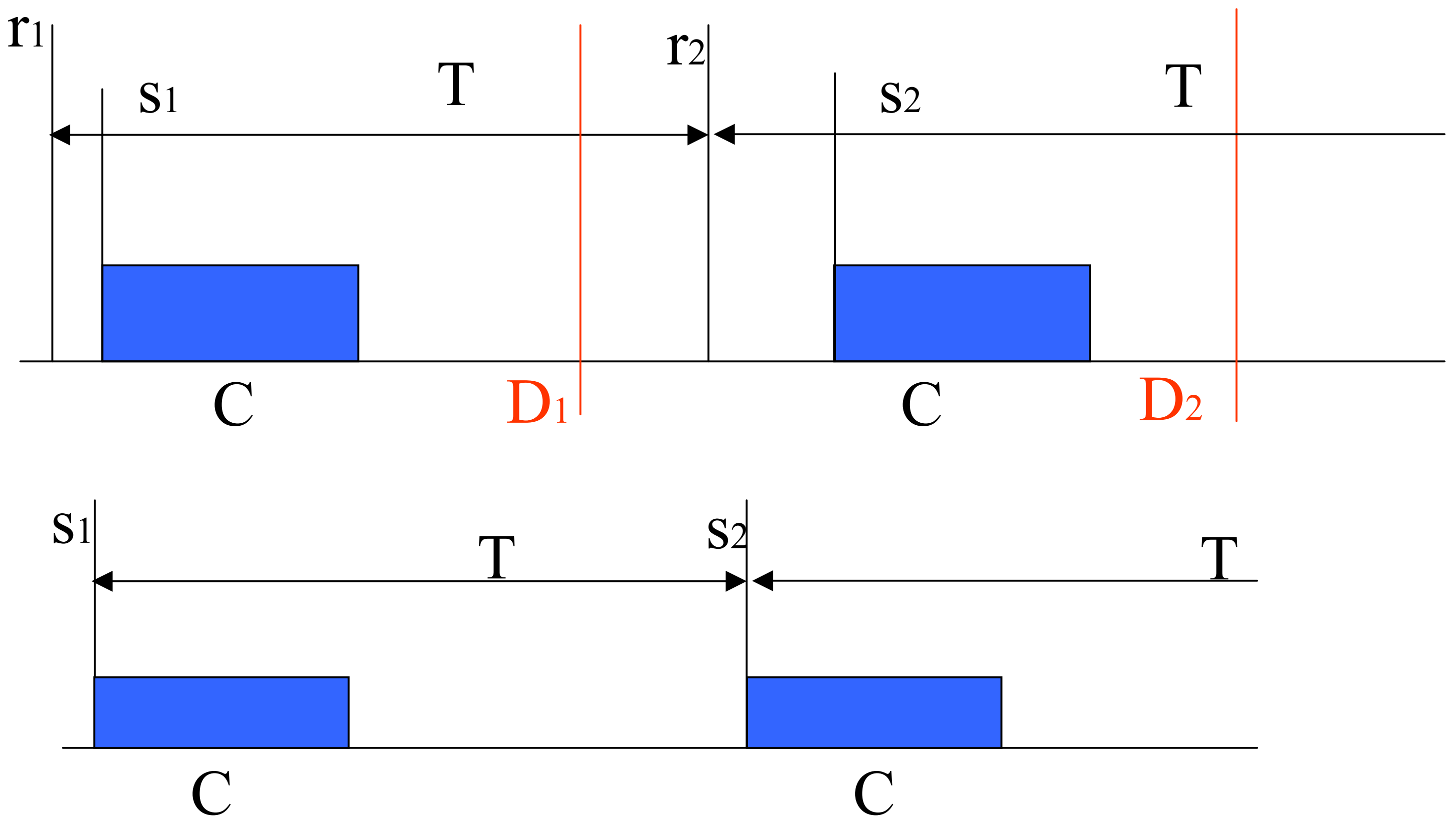
- 1 / r, D cohérent ordre partiel, T / - EDF modifié
- 1 / préc. sous-tâches / - condition d’ordonnançabilité
- - / préc. généralisées / T

Modèle proposé : non-préemptif

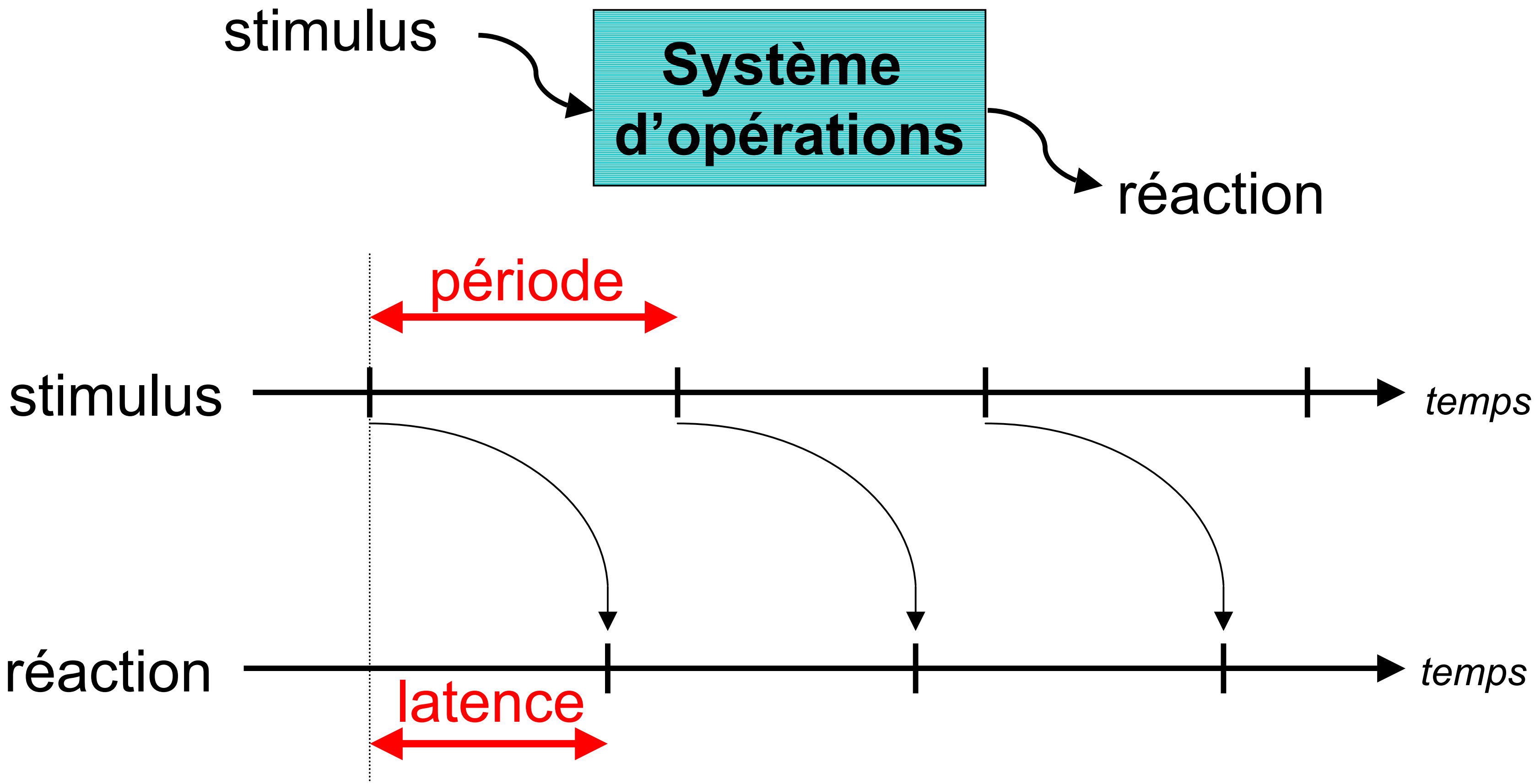
- Modèle classique vs. modèle proposé
- Contrainte de périodicité stricte
- Contrainte de latence
- Graphe répétitif

Modèle classique vs. modèle proposé

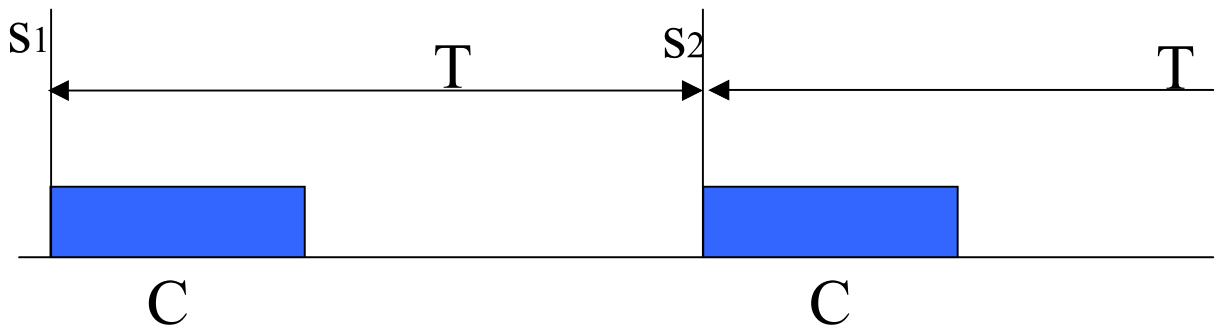
Opération à la place de tâche afin d'être indépendant des aspects d'implantation

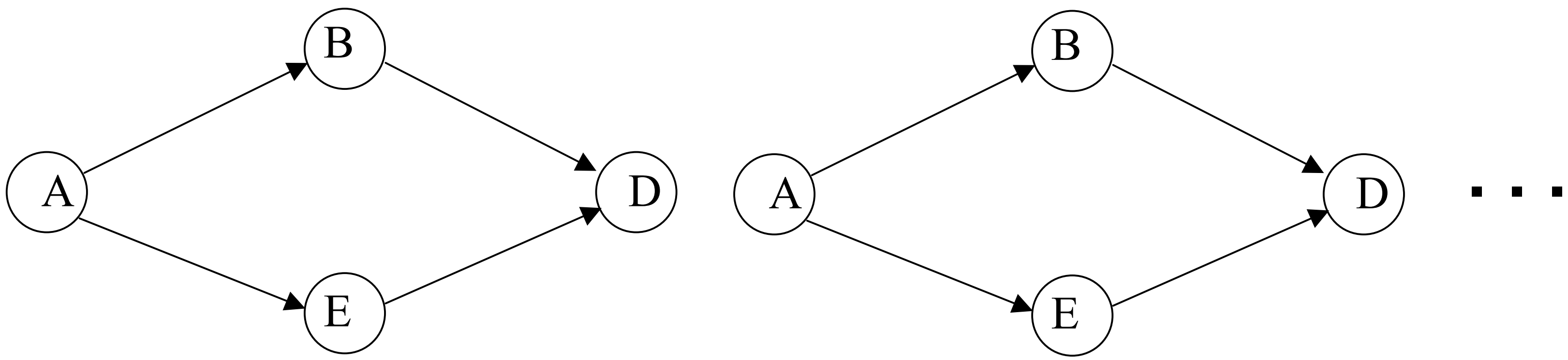


Système réactif



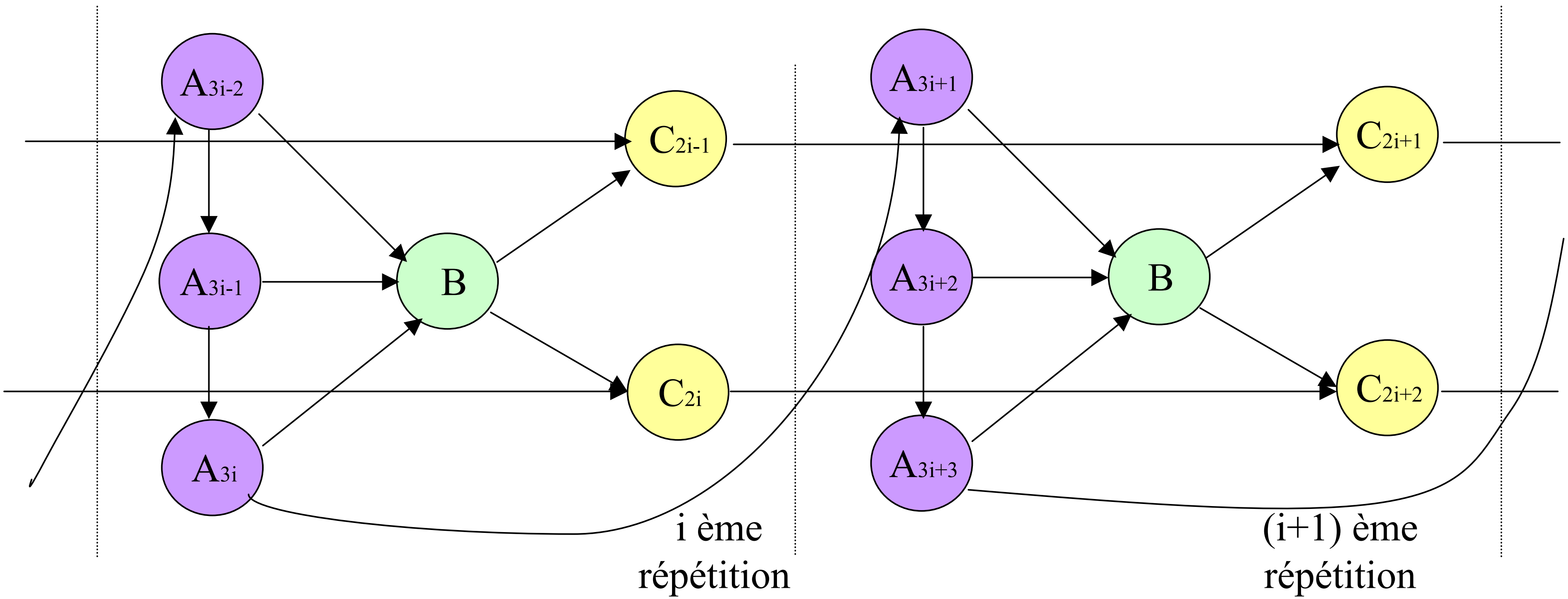
Graphe : répétition infinie



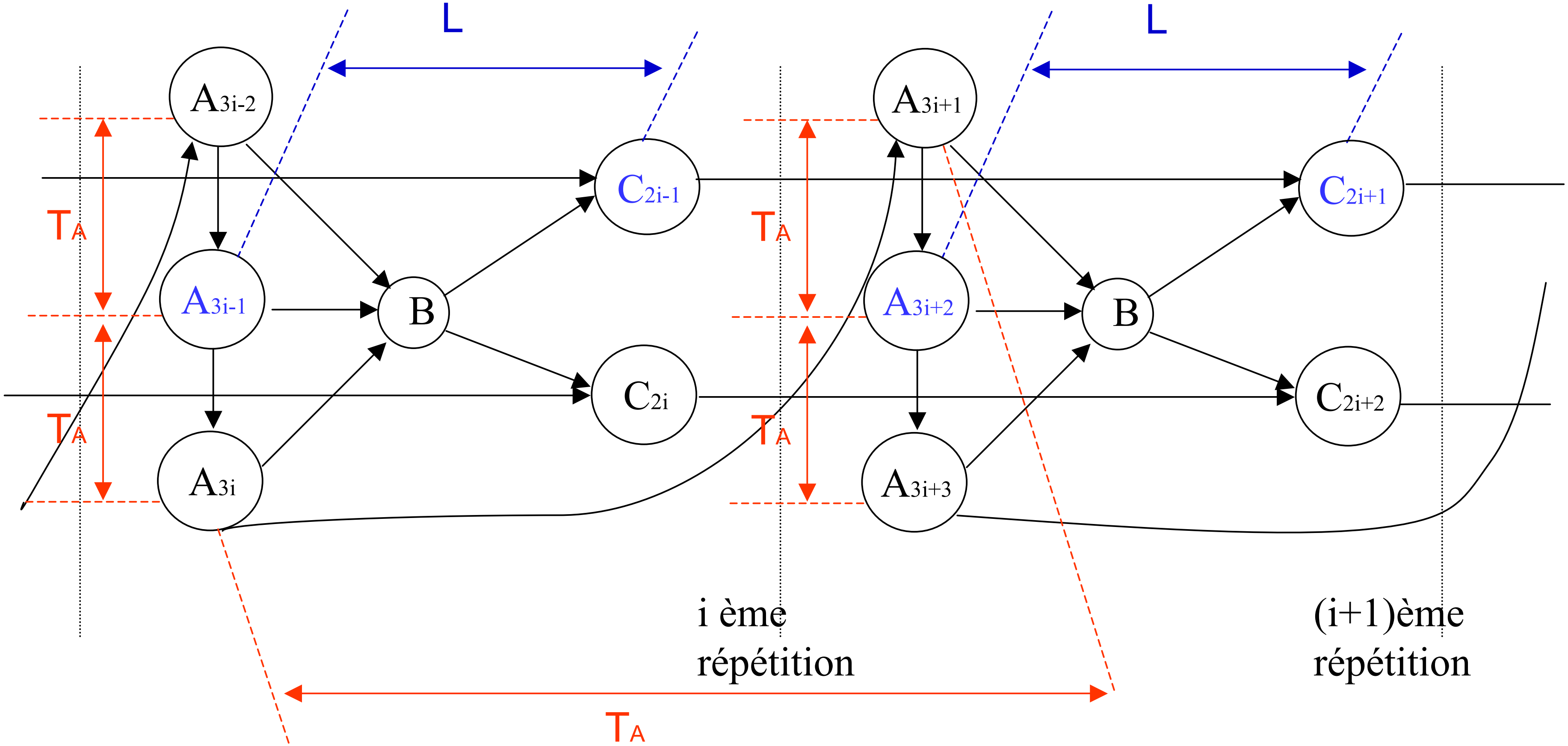


Motif du graphe répété infiniment

Graphe : répétition infinie et finie



Contrainte de périodicité stricte et contrainte de latence



$$s_{C2i-1} - s_{A3i-1} + C_C \le L \qquad s_{Ai+1} - s_{Ai} = T_A, \ \forall i \in \mathbb{N}^*$$

Relation entre la latence et la périodicité

Théorème : la contrainte de périodicité est une contrainte de latence

$$s_{Ai+1} - s_{Ai} = T_A \,,\ \forall\, i \in \mathbb{N}^* \quad \Rightarrow \quad s_{Ai+1} - s_{Ai} \leq L - C_A,\ \forall\, i \in \mathbb{N}^*$$

Ordonnancement et ordonnançabilité dans le cas monoprocesseur

- Condition d'ordonnançabilité : contraintes de précédences et de périodicités
- Condition d'ordonnançabilité : contraintes de précédences et de latences
- Condition d'ordonnançabilité : cas général
- Algorithme d'ordonnancement : cas général
- Extension – modèle classique

Condition d’ordonnançabilité : contraintes de précédences et de périodicités

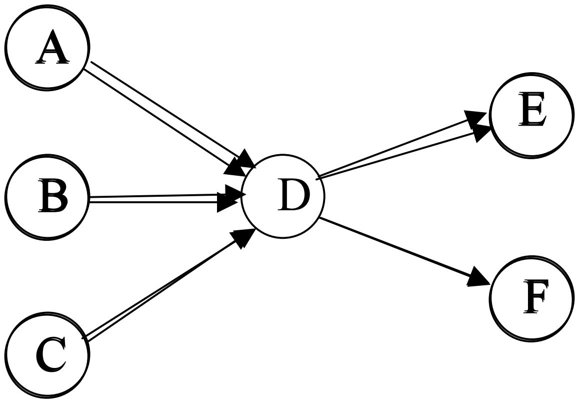
Théorème : le problème d’ordonnancement de *n* opérations avec périodes et durées d’exécution quelconques est NP-difficile.

↓

On considère le problème d’ordonnancement de *n* opérations avec périodes et durées d’exécution multiples entre elles.

Condition d'ordonnançabilité : contraintes de précédences et de périodicités

Théorème : héritage de périodes pour une opération non-périodique



$T_D = \max\{T_A, T_B, T_C\}$ $T_D = \min\{T_E, T_F\}$

$T_D = \min\{T_E, T_F\}$

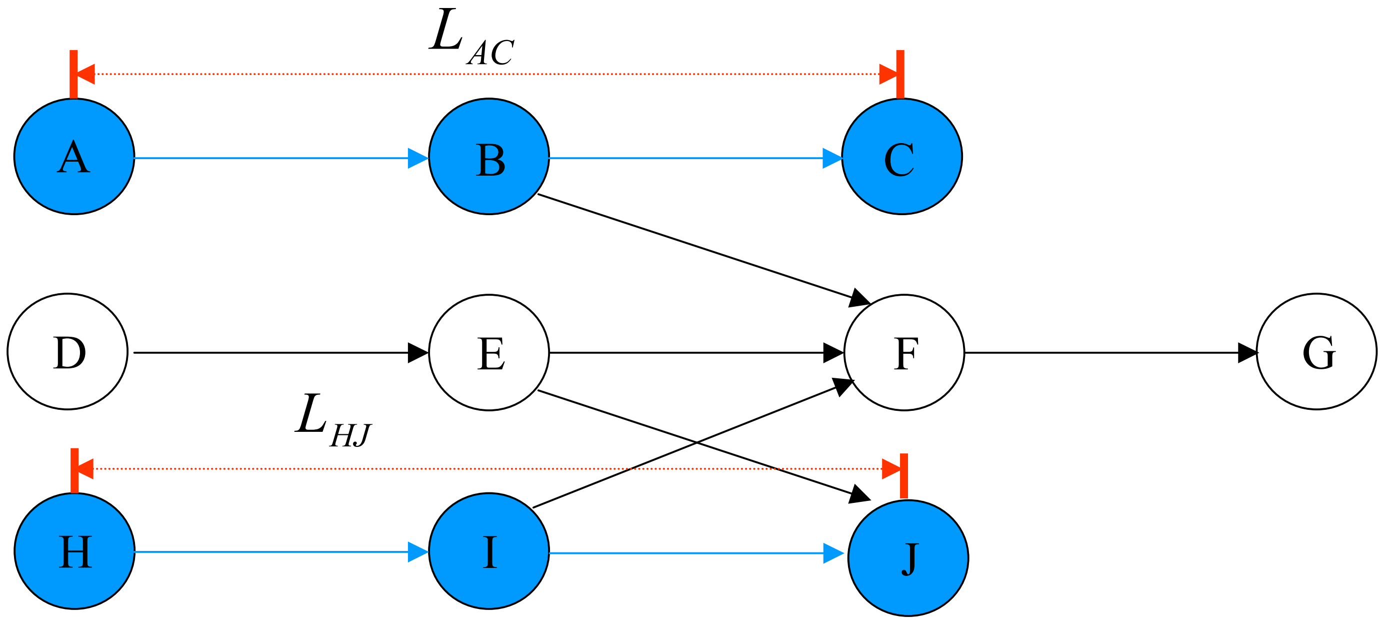
Condition d'ordonnançabilité : contraintes de précédences et de périodicités

- **Théorème :** pour un système avec contraintes de précédences et de périodicités
  - l'existence d'une hyperpériode de s_{max} à $s_{max} + T$, où T est PPCM de toutes les périodes
  - si le système est ordonnançable alors $\sum_{A \in V} \frac{C_A}{T_A} \leq 1$

Condition d'ordonnançabilité : contraintes de précédences et de latences

- Relations entre paires d'opérations
  - **II** : condition d'ordonnançabilité pour des paires en relation **II**
  - **Z** : condition d'ordonnançabilité pour des paires en relation **Z**
  - **X** : condition d'ordonnançabilité pour des paires en relation **X**
- Condition générale d'ordonnançabilité

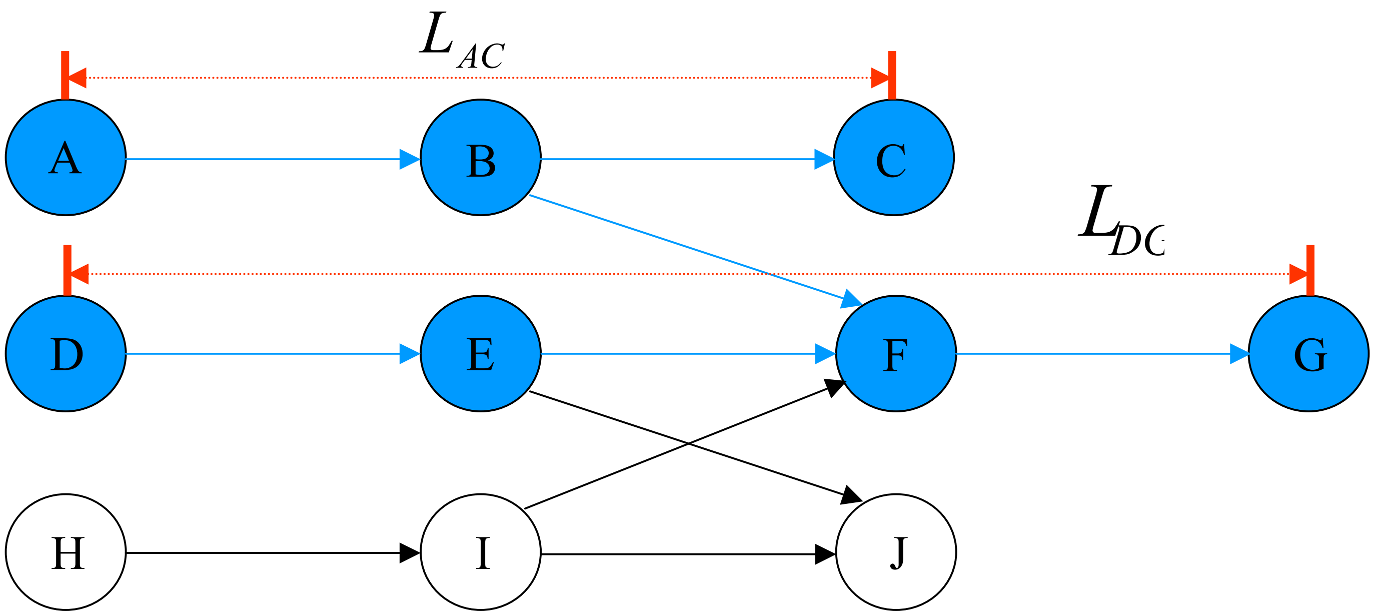
Relations entre paires d'opérations : II



(A,C) II *(H,J)*

Théorème : le système est ordonnançable ssi $L_{AC} \geq \sum_{M \in I(A,C)} C_M$ et $L_{HJ} \geq \sum_{M \in I(H,J)} C_M$

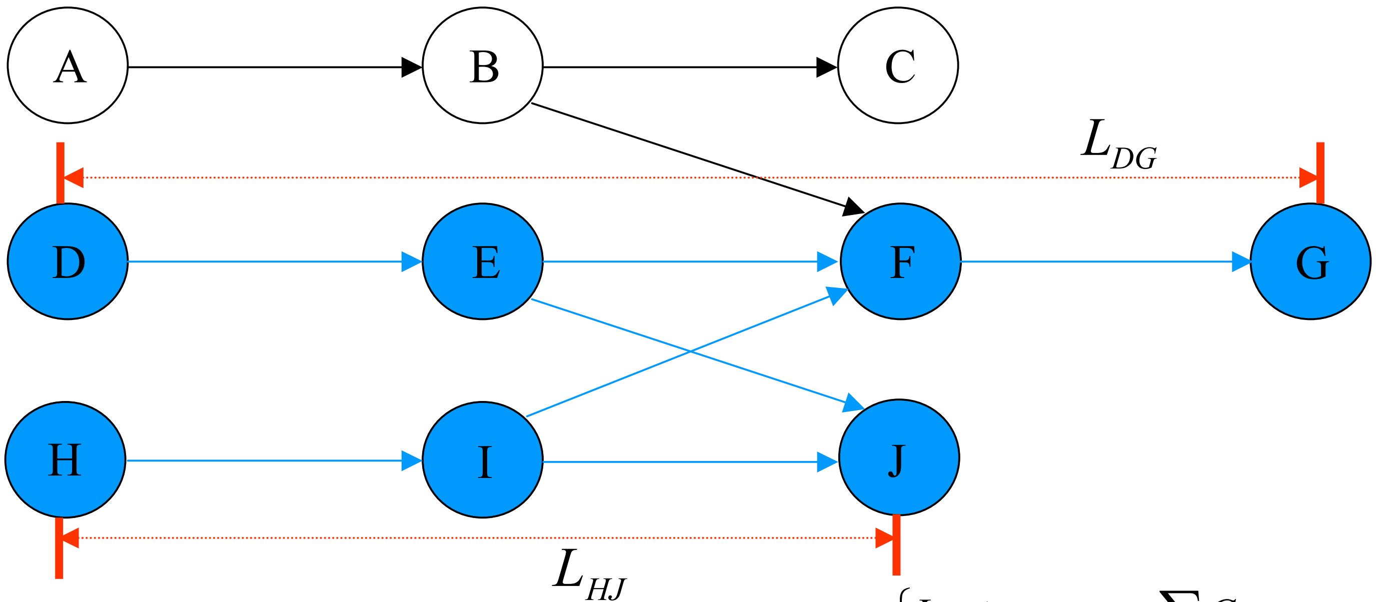
Relations entre paires d'opérations : Z



(A,C) Z (D,G)

Théorème : le système est ordonnaçable ssi $L_{AC} \geq \sum_{M \in I(A,C)} C_M$ et $L_{DG} \geq \sum_{M \in I(D,G)} C_M$

Relations entre paires d'opérations : X



(D,G) X *(H,J)*

$$
\begin{cases}
L_{DG} \geq \sum\limits_{M \in \left\{ I(D,G) \bigcup \bigcup\limits_{N \in \Gamma^{-}_{DG}(H,J)} I(H,E) \right\}} C_M \quad et\ L_{HJ} \geq \sum\limits_{M \in \left\{ I(H,J) \bigcup \bigcup\limits_{N \in \Gamma^{+}_{HJ}(D,G)} I(N,G) \right\}} C_M \\
L_{DG} \geq \sum\limits_{M \in \left\{ I(D,G) \bigcup \bigcup\limits_{N \in \Gamma^{+}_{DG}(H,J} I(N,J) \right\}} C_M \quad et\ L_{HJ} \geq \sum\limits_{M \in \left\{ I(H,J) \bigcup \bigcup\limits_{N \in \Gamma^{-}_{HJ}(D,G)} I(D,N) \right\}} C_M \\
L_{DG} = \sum\limits_{M \in I(D,G)} C_M \ et\ L_{HJ} \geq \sum\limits_{M \in I(D,G) \bigcup I(H,J)} C_M \\
L_{DG} \geq \sum\limits_{M \in I(D,G) \bigcup I(H,J)} C_M \quad et\ L_{HJ} = \sum\limits_{M \in I(H,J)} C_M
\end{cases}
$$

Théorème : le système est ordonnançable ssi une des équations suivantes est satisfaite

Condition d'ordonnançabilité : contraintes de précédences et de latences

Théorème : le système est ordonnançable ssi:

- pour toutes les paires *(A,C)* II *(H,J)*, $L_{AC} \geq \sum_{M \in I(A,C)} C_M$ et $L_{DG} \geq \sum_{M \in I(D,G)} C_M$
- pour toutes les paires *(A,C)* Z *(D,G)*, $L_{AC} \geq \sum_{M \in I(A,C)} C_M$ et $L_{HJ} \geq \sum_{M \in I(H,J)} C_M$
- pour toutes les paires *(D,G)* X *(H_i,J_i)*, une des équations suivantes est satisfaite :

$$\begin{cases} L_{DG} = \sum_{M \in I(D,G)} C_M \quad et \quad L_{H_iJ_i} \geq \sum_{M \in I(D,G) \bigcup I(H_i,J_i)} C_M \\ L_{DG} \geq \sum_{M \in I(D,G) \bigcup \bigcup_{i \in \{1,j\}; N \in \Gamma^{-}_{DG}(H_i,J_i)} I(H_i,N) \bigcup \bigcup_{i \in \{j,k\}, N \in \Gamma^{+}_{DG}(H_i,J_i)} I(N,J_i) \bigcup \bigcup_{i \in [k,n]} I(Hi,Ji)} C_M \\ \begin{cases} L_{Hi,Ji} \geq \sum_{M \in \left\{ I(Hi,Ji) \bigcup \bigcup_{N \in \Gamma^{+}_{HiJi}(D,G)} I(N,G) \right\}} C_M, \forall i \in \{1,\ldots, j\}; \\ L_{Hi,Ji} \geq \sum_{M \in \left\{ I(Hi,Ji) \bigcup \bigcup_{N \in \Gamma^{-}_{HiJi}(D,G)} I(Hi,N) \right\}} C_M, \forall i \in \{j,\ldots, k\}; \\ L_{Hi,Ji} = \sum_{M \in I(hi,Ji)} C_M, \forall i \in \{k,\ldots, n\}. \end{cases} \end{cases}$$

Condition d'ordonnançabilité : cas général

- **Théorème :** pour un système avec contraintes de précédences, de périodicités et de latences

  - l'existence d'une hyperpériode de s_{max} à s_{max} +T, où T est le PPCM de toutes les périodes

Condition d'ordonnançabilité : cas général

- **Théorème :** si le système est ordonnançable alors

$\sum_{A \in V} \frac{C_A}{T_A} \leq 1$ et

$$
\begin{cases}
L_{AB} \geq \sum_{M \in W_1} C_H + \sum_{H \in V, T_H < T_A} C_H \frac{L_{AB}}{T_H} \\
W_1 = M(A,B) \cup \bigcup_{i \in \{1,j\}; E \in lt_{AB}(C_i D_i)} M(C_i, E) \cup \bigcup_{i \in \{j+1,k\}, E \in ft_{AB}(C_i D_i)} M(E, D_i) \cup \bigcup_{i \in \{k+1,m\}} M(Ci, Di) \cup \bigcup_{i \in \{m+1,n\}} M(Ci, Di) \\
\begin{cases}
L_{CiDi} \geq \sum_{H \in \left\{ M(CiDi) \bigcup \bigcup_{E \in ft_{CiDi}(AB)} M(E,B) \right\}} C_H, \forall i \in \{1,\ldots, j\}; \\
L_{CiDi} \geq \sum_{H \in \left\{ M(Ci,Di) \bigcup \bigcup_{N \in lt_{CiDi}(AB)} M(A,E) \right\}} C_H, \forall i \in \{j+1,\ldots, k\}; \\
L_{CiDi} \geq \sum_{H \in M(CiDi)} C_H, \forall i \in \{k+1,\ldots, m\} \\
L_{CiDi} \geq \sum_{H \in \left\{ M(Ci,Di) \bigcup \bigcup M(A,B) \right\}} C_H, \forall i \in \{m+1,\ldots, n\}
\end{cases}
\end{cases}
$$

Algorithme d'ordonnancement

- Etape 1 : Algorithme de marquage
  - la marque d'une opération est la plus petite de toutes les latences pour lesquelles il y a au moins un chemin partant de l'opération et allant jusqu'à la deuxième opération de chaque latence
- Etape 2 : Algorithme d'ordonnancement
  - régime transitoire : l'étape d'initialisation ordonnance d'abord les opérations sans aucune contrainte, puis les opérations avec la marque $\neq 0$, et finalement l'opération périodique avec la plus petite période
  - régime permanent : une fois qu'une opération périodique est ordonnancée, l'ordonnancement est le contraire de celui de l'étape d'initialisation

Algorithme d’ordonnancement

- **Théorème :** l’algorithme est optimal (s’il y a un ordonnancement, l’algorithme le trouvera)
- **Corollaire :** en appliquant l’algorithme d’ordonnancement de 0 à 2T, on obtient un test d’ordonnançabilité, T est le PPCM de toutes les périodes

Extension vers le modèle classique

- **Théorème :**
  - L(A,B) contrainte de latence <=> échéance de B une fois A ordonnancée
    - $s_B \leq L + s_A - C_B$ <=> $s_B \leq D_B$ une fois A ordonnancée
- Algorithme de marquage par rapport aux échéances
  - la marque d'une opération est la plus petite de toutes les échéances de ses prédécesseurs

Conclusion

- Nouveau modèle
- Relations entre
  - les contraintes de latences et de périodicités
  - la contrainte de latence et l'échéance
- Condition générale d'ordonnançabilité
- Algorithme optimal d'ordonnancement
- Test d'ordonnançabilité

Perspectives

- Impact de l'héritage des périodes dans le modèle classique
- Relaxation période stricte - prise en compte de la gigue
- Prise en compte de la préemption et de son coût
- Utilisation du problème central répétitif afin de trouver des résultats pour le problème d'optimisation associé à notre problème

Questions