

# Non-cooperative scheduling of multiple bag-of-task applications

Arnaud Legrand et Corinne Touati

Laboratoire ID-IMAG, CNRS-INRIA Grenoble, France

Universite de Tsukuba, Japon

Juin 2006

# Théorie des jeux : Quezaco ?



- ▶ 2 joueurs
- ▶ symétrique
- ▶ somme nulle
- ▶ séquentiel
- ▶ information parfaite
- ▶ nombre de coups finis

# Théorie des jeux : Quezaco ?



- ▶ 2 joueurs
- ▶ symétrique
- ▶ somme nulle
- ▶ séquentiel
- ▶ infomation parfaite
- ▶ nombre de coups finis

# Théorie des jeux : Quezaco ?



- ▶ 2 joueurs
- ▶ symétrique
- ▶ somme nulle
- ▶ séquentiel
- ▶ infomation parfaite
- ▶ nombre de coups finis

# Théorie des jeux : Quezaco ?



- ▶ 2 joueurs
- ▶ symétrique
- ▶ somme nulle
- ▶ séquentiel
- ▶ infomation parfaite
- ▶ nombre de coups finis

# Théorie des jeux : Quezaco ?



- ▶ 2 joueurs
- ▶ symétrique
- ▶ somme nulle
- ▶ séquentiel
- ▶ infomation parfaite
- ▶ nombre de coups finis

# Théorie des jeux : Quezaco ?



- ▶ 2 joueurs
- ▶ symétrique
- ▶ somme nulle
- ▶ séquentiel
- ▶ infomation parfaite
- ▶ nombre de coups finis

# Théorie des jeux : Quezaco ?



- ▶ 2 joueurs
- ▶ symétrique
- ▶ somme nulle
- ▶ séquentiel
- ▶ infomation parfaite
- ▶ nombre de coups finis

## ▶ Meilleure stratégie

- ▶ réponse à une attaque terroriste
- ▶ sauvegarde de l'environnement...

## ▶ Equilibres entre populations

- ▶ génétique
- ▶ <hawk-dove>

## ▶ Partage de ressources

- ▶ héritages
- ▶ enchères
- ▶ Applications concurrentes dans un système informatique (BP, CPU, mémoire...)

## ▶ Meilleure stratégie

- ▶ réponse à une attaque terroriste
- ▶ sauvegarde de l'environnement...

## ▶ Equilibres entre populations

- ▶ génétique
- ▶ «hawk-dove»

## ▶ Partage de ressources

- ▶ héritages
- ▶ enchères
- ▶ Applications concurrentes dans un système informatique (BP, CPU, mémoire...)

## ▶ Meilleure stratégie

- ▶ réponse à une attaque terroriste
- ▶ sauvegarde de l'environnement...

## ▶ Equilibres entre populations

- ▶ génétique
- ▶ «hawk-dove»

## ▶ Partage de ressources

- ▶ héritages
- ▶ enchères
- ▶ Applications concurrentes dans un système informatique (BP, CPU, mémoire...)

## ▶ Meilleure stratégie

- ▶ réponse à une attaque terroriste
- ▶ sauvegarde de l'environnement...

## ▶ Equilibres entre populations

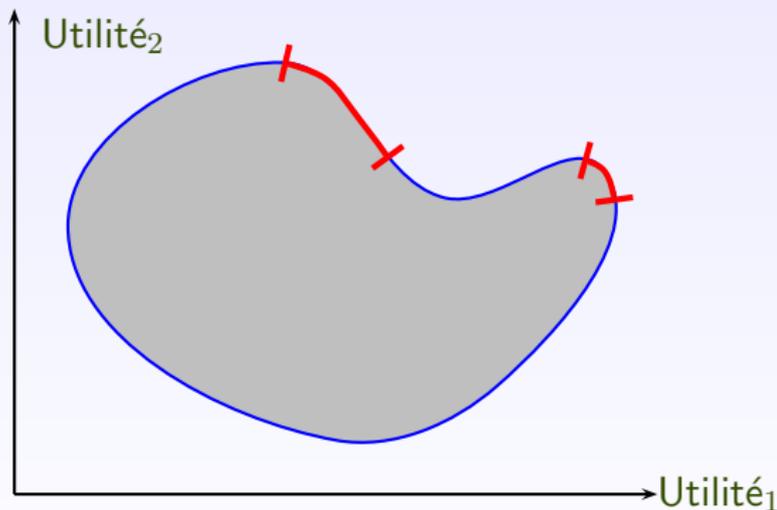
- ▶ génétique
- ▶ «hawk-dove»

## ▶ Partage de ressources

- ▶ héritages
- ▶ enchères
- ▶ **Applications concurrentes** dans un système informatique (BP, CPU, mémoire...)

# Théorie des jeux : Plus formellement...

- ▶ 1 joueur  $\longleftrightarrow$  1 fonction d'utilité
- ▶ Nécessité de notion d'optimalité globale du système: optimalité de **Pareto**



- ▶ 1 partage dépend de la décision de tous les joueurs

$\rightsquigarrow$  Optimisations **multi-critères**

# Théorie des jeux: coopérer ou non ?

Jeux coopératifs

institution, règles  
(pénalités)

Jeux non-coopératifs

comportement individuel  
prise de décision individuelle

Exemple: Croisement routier:

- ▶ Approche **coopérative** : je respecte la signalisation
- ▶ Approche **non-coopérative** : je traverse aussi vite que possible

# Théorie des jeux: coopérer ou non ?

Jeux coopératifs

institution, règles  
(pénalités)

Jeux non-coopératifs

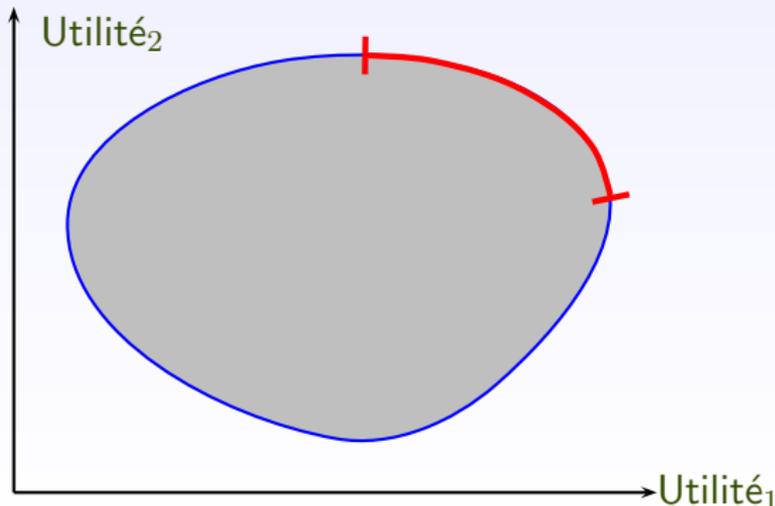
comportement individuel  
prise de décision individuelle

Exemple: Croisement routier:

- ▶ Approche **coopérative** : je respecte la signalisation
- ▶ Approche **non-coopérative** : je traverse aussi vite que possible

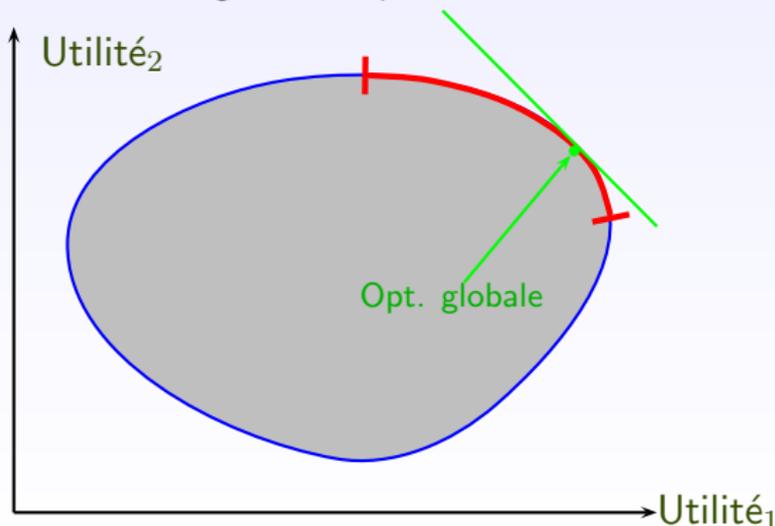
# Jeux coopératifs

- ▶ Les joueurs collaborent pour obtenir un équilibre acceptable pour tous: marchandage ou **équité**
- ▶ Propriété des partages  $\rightsquigarrow$  **règles** de marchandage
- ▶ Défis
  - ▶ Quelles règles choisir pour un système donné
  - ▶ Algorithmes pour les mettre en œuvre dans les systèmes



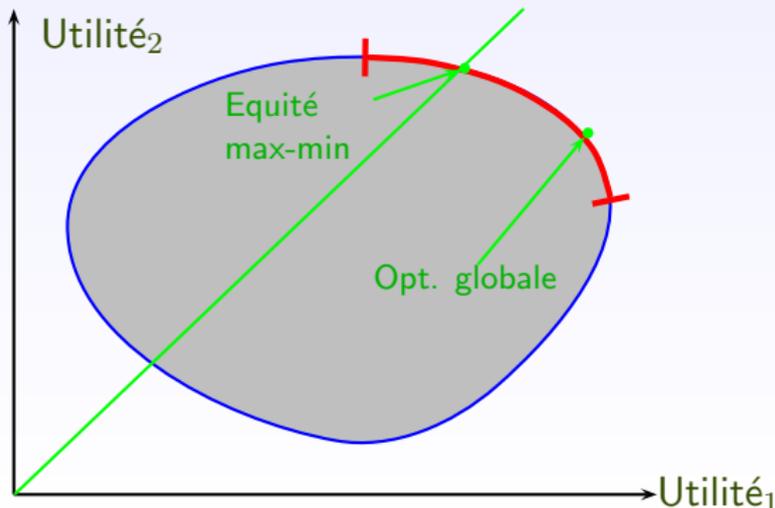
# Jeux coopératifs

- ▶ Les joueurs collaborent pour obtenir un équilibre acceptable pour tous: marchandage ou **équité**
- ▶ Propriété des partages  $\rightsquigarrow$  **règles** de marchandage
- ▶ Défis
  - ▶ Quelles règles choisir pour un système donné
  - ▶ Algorithmes pour les mettre en œuvre dans les systèmes



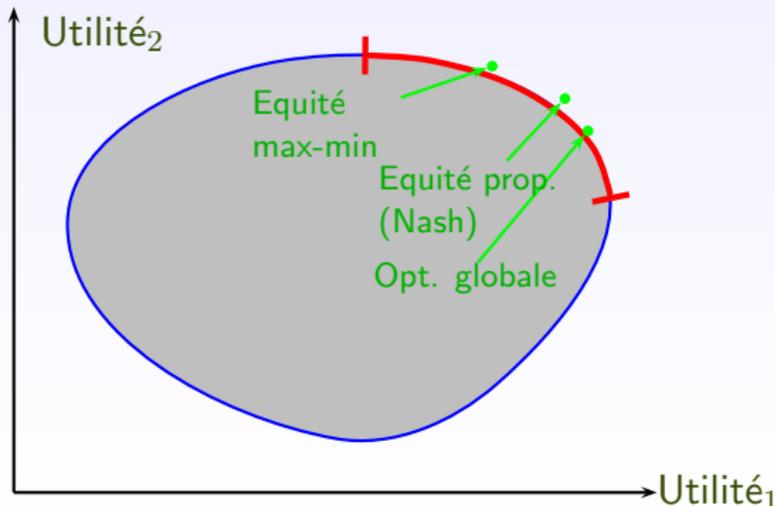
# Jeux coopératifs

- ▶ Les joueurs collaborent pour obtenir un équilibre acceptable pour tous: marchandage ou **équité**
- ▶ Propriété des partages  $\rightsquigarrow$  **règles** de marchandage
- ▶ Défis
  - ▶ Quelles règles choisir pour un système donné
  - ▶ Algorithmes pour les mettre en œuvre dans les systèmes

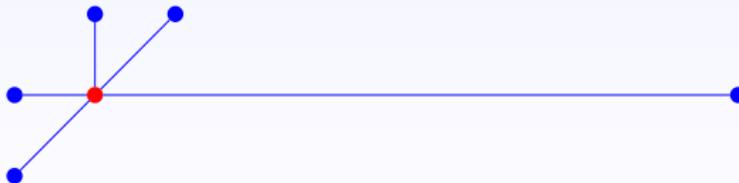


# Jeux coopératifs

- ▶ Les joueurs collaborent pour obtenir un équilibre acceptable pour tous: marchandage ou **équité**
- ▶ Propriété des partages  $\rightsquigarrow$  **règles** de marchandage
- ▶ Défis
  - ▶ Quelles règles choisir pour un système donné
  - ▶ Algorithmes pour les mettre en œuvre dans les systèmes



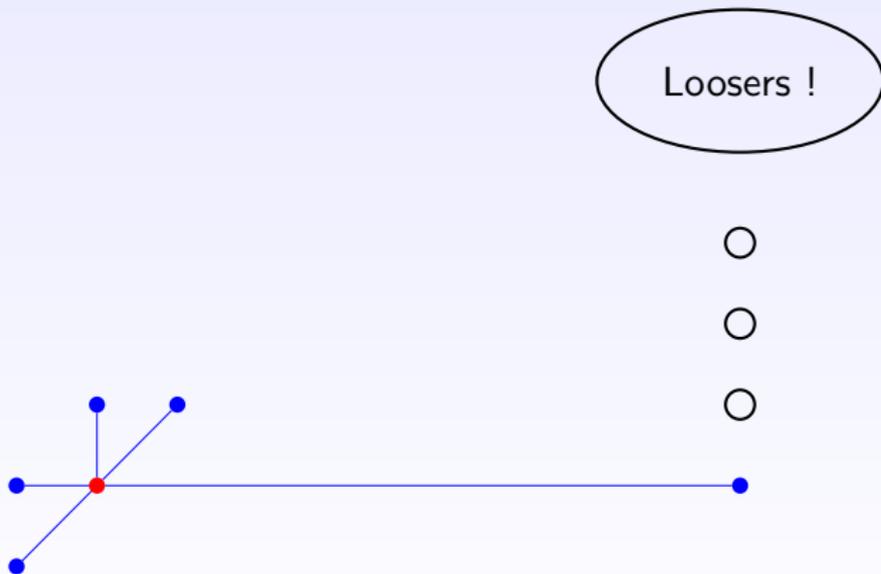
## Critique de l'équité max-min



## Critique de l'équité max-min

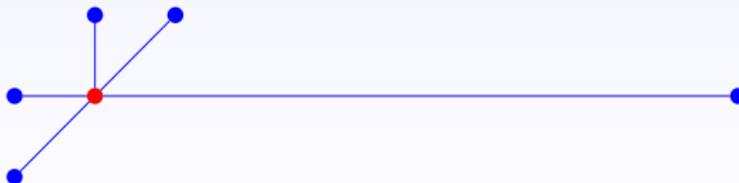


## Critique de l'équité max-min

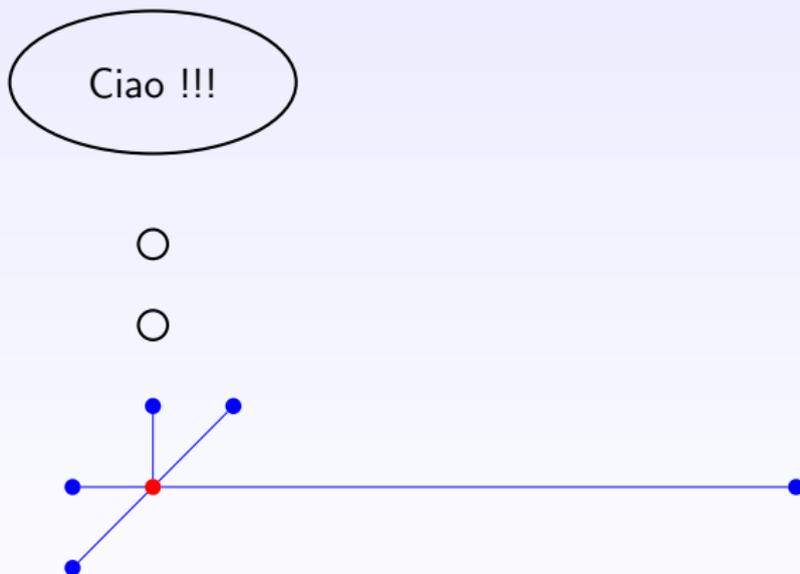


Faible utilisation des ressources du réseau ...

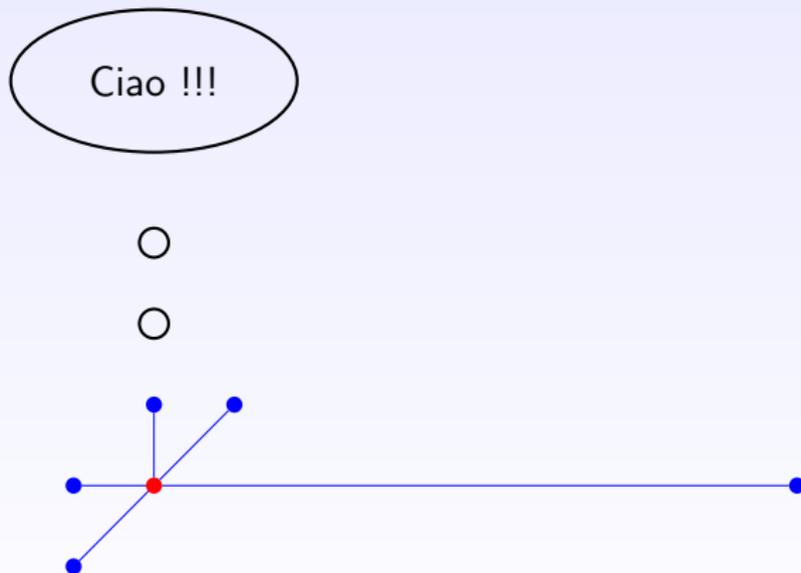
## Critique de l'optimalité globale



## Critique de l'optimalité globale



## Critique de l'optimalité globale



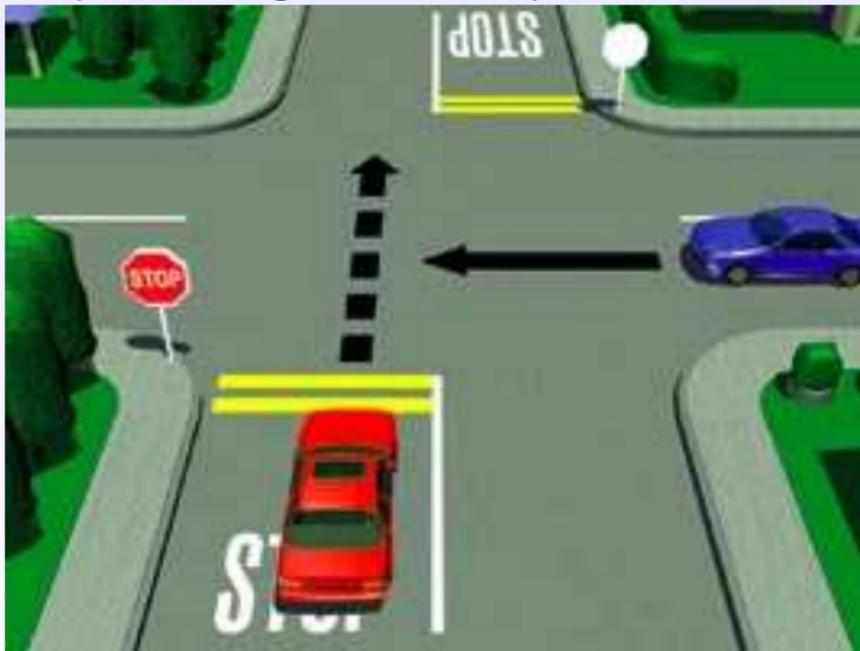
Peut exclure certains utilisateurs ...

## Critères d'équité (classiques)

- ▶ Sous forme d'axiomes ( $\Rightarrow$  justification)
- ▶ Formule d'optimisation ( $\Rightarrow$  calcul)
- ▶ Règles de marchandage ( $\Rightarrow$  protocoles)

# Jeux coopératifs VS non coopératifs

Coopératifs : signalisations + police



# Jeux coopératifs VS non coopératifs

Coopératifs : signalisations + police



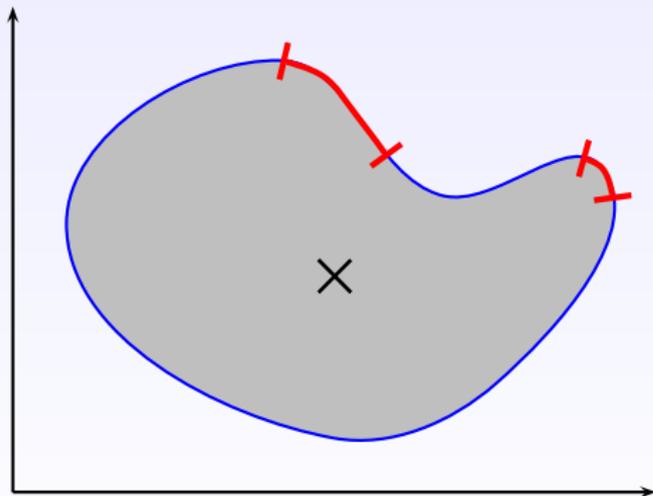
# Jeux coopératifs VS non coopératifs

NON - coopératifs : on enlève la police  $\Rightarrow$  chacun pour soi



Les joueurs optimisent **individuellement** leur fonction d'utilité

- ▶ généralement Pareto inefficaces



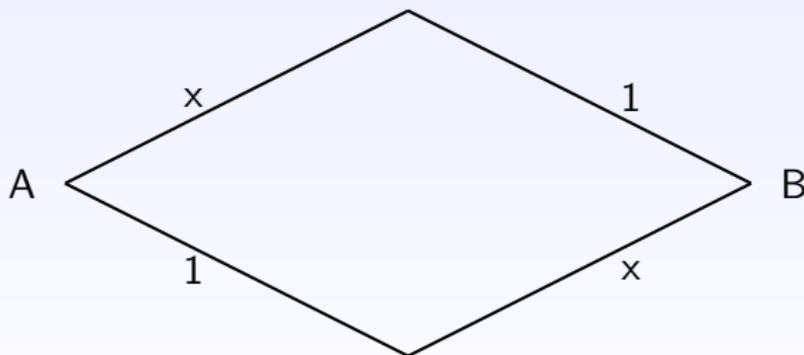
Les joueurs optimisent **individuellement** leur fonction d'utilité

- ▶ généralement Pareto inefficaces
- ▶ conséquence: **paradoxes de Braess**

# Jeux non-coopératifs

Les joueurs optimisent **individuellement** leur fonction d'utilité

- ▶ généralement Pareto inefficaces
- ▶ conséquence: **paradoxes de Braess**



Coût pour chaque connexion :

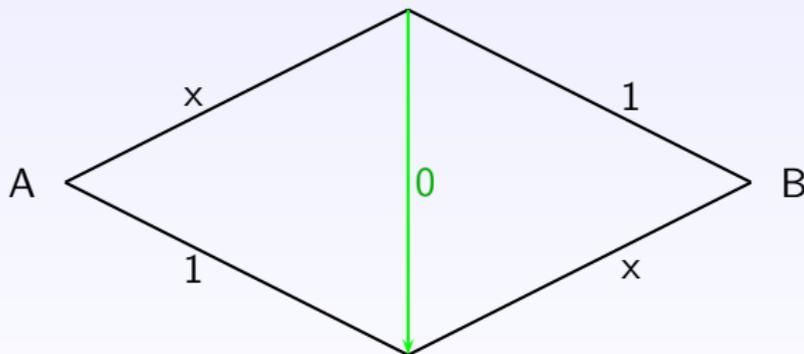
Avant ajout lien : 1.5

Après : 2

# Jeux non-coopératifs

Les joueurs optimisent **individuellement** leur fonction d'utilité

- ▶ généralement Pareto inefficaces
- ▶ conséquence: **paradoxes de Braess**



Coût pour chaque connexion :

Avant ajout lien : 1.5

Après : 2

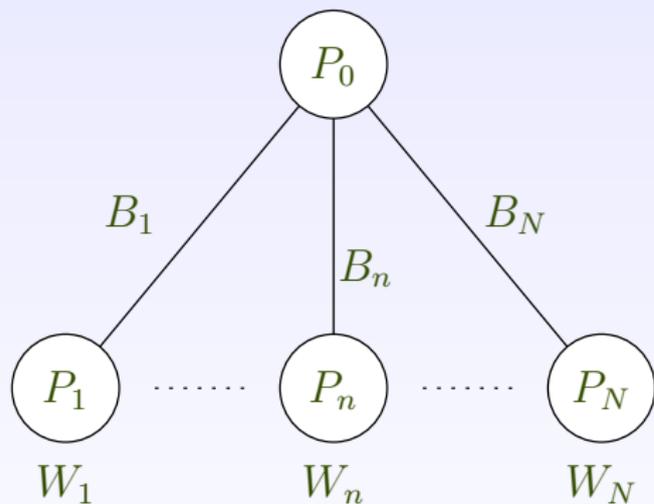
Grande plateforme  $\Rightarrow$  multi-utilisateurs / multi-applications  
 $\Rightarrow$  on veut un partage **optimal** et **équitable** des ressources

Notre résultat :

Même si le **système** effectue un **partage parfait** des ressources entre les demandes d'accès concurrents,  
le comportement **non-coopératif** des utilisateurs entraîne une **mauvaise utilisation** de la plateforme

Équité au niveau système  $\nRightarrow$  équité au niveau applicatif

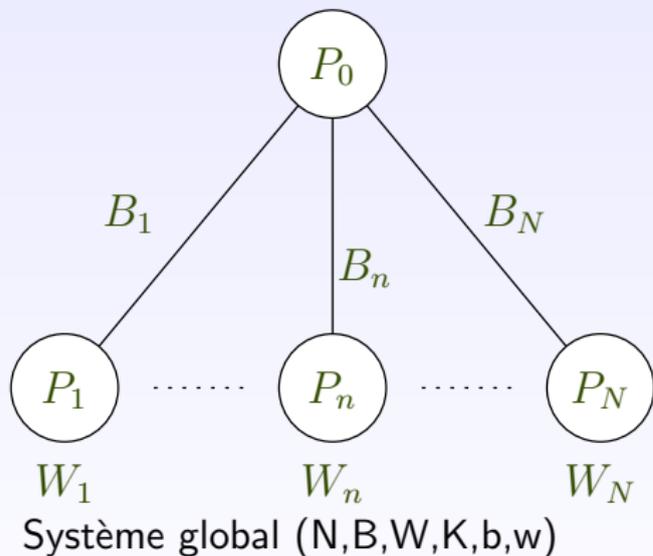
# Système considéré - 1



Hypothèses : multi-port  
partage instantané parfait des ressources

Plateforme maître/esclave

- ▶  $N$  processeurs de puissance  $W_n$  (en  $\text{Mflop.s}^{-1}$ )
- ▶ sur des liens de capacité  $B_n$  (en  $\text{Mb.s}^{-1}$ )
- ▶ système physique  $(N, B, W)$



### Multi-applications

- ▶  $K$  applications
- ▶ grand nombre de tâches identiques et indépendantes
- ▶ caractérisées par  $(w_k, b_k)$  (CPU, BP)
- ▶ système utilisateur (K,b,w)

$\alpha_{n,k}$  : nombre de tâches de type  $k$  effectuées sur la machine  $n$  par unité de temps

Contraintes physiques :

$$\text{Calcul } \sum_{k=1}^K \alpha_{n,k} w_k \leq W_n$$

$$\text{Communication } \sum_{k=1}^K \alpha_{n,k} b_k \leq B_n$$

Utilité :

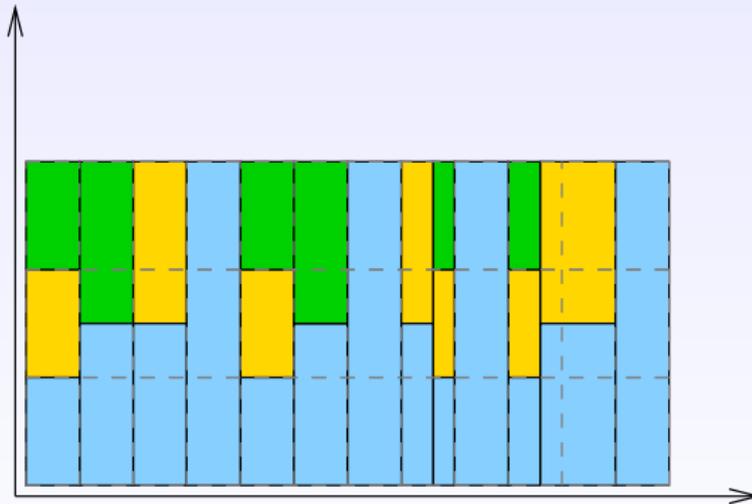
$$\sum_n \alpha_{n,k}$$

Stratégie : ordonnancement

Si 1 application seule :

$$\alpha_{n,1} = \min\left(\frac{W_n}{w}, \frac{B_n}{b}\right)$$

# Ordonnements typiques



- ▶ Pour un système donné  $(N, B, W, K, b, w)$  il **existe exactement** un équilibre de Nash (et on en connaît une formule close)
- ▶ **UNE** machine = Nash est **Pareto optimal**
- ▶ **Plusieurs** machines = **ca dépend** (et on peut écrire les CNS)

Exemple : 2 machines  $\begin{cases} B_1 = 50 & B_2 = 100 \\ W_1 = 100 & W_2 = 50 \end{cases}$

2 applications  $\begin{cases} b_1 = 1 & b_2 = 2 \\ w_1 = 2 & w_2 = 1 \end{cases}$

Avec collaboration :  $\alpha_1 = \alpha_2 = 50$

Sans collaboration :  $\alpha_1 = \alpha_2 = 37.5$

- ▶ Pour un système donné  $(N, B, W, K, b, w)$  il **existe exactement** un équilibre de Nash (et on en connaît une formule close)
- ▶ **UNE** machine = Nash est **Pareto optimal**
- ▶ **Plusieurs** machines = **ca dépend** (et on peut écrire les CNS)

Exemple : 2 machines  $\begin{cases} B_1 = 50 & B_2 = 100 \\ W_1 = 100 & W_2 = 50 \end{cases}$

2 applications  $\begin{cases} b_1 = 1 & b_2 = 2 \\ w_1 = 2 & w_2 = 1 \end{cases}$

Avec collaboration :  $\alpha_1 = \alpha_2 = 50$

Sans collaboration :  $\alpha_1 = \alpha_2 = 37.5$

- ▶ Pour un système donné  $(N, B, W, K, b, w)$  il **existe exactement** un équilibre de Nash (et on en connaît une formule close)
- ▶ **UNE** machine = Nash est **Pareto optimal**
- ▶ **Plusieurs** machines = **ca dépend** (et on peut écrire les CNS)

Exemple : 2 machines  $\begin{cases} B_1 = 50 & B_2 = 100 \\ W_1 = 100 & W_2 = 50 \end{cases}$

2 applications  $\begin{cases} b_1 = 1 & b_2 = 2 \\ w_1 = 2 & w_2 = 1 \end{cases}$

Avec collaboration :  $\alpha_1 = \alpha_2 = 50$

Sans collaboration :  $\alpha_1 = \alpha_2 = 37.5$

- ▶ Pour un système donné  $(N, B, W, K, b, w)$  il **existe exactement** un équilibre de Nash (et on en connaît une formule close)
- ▶ **UNE** machine = Nash est **Pareto optimal**
- ▶ **Plusieurs** machines = **ca dépend** (et on peut écrire les CNS)

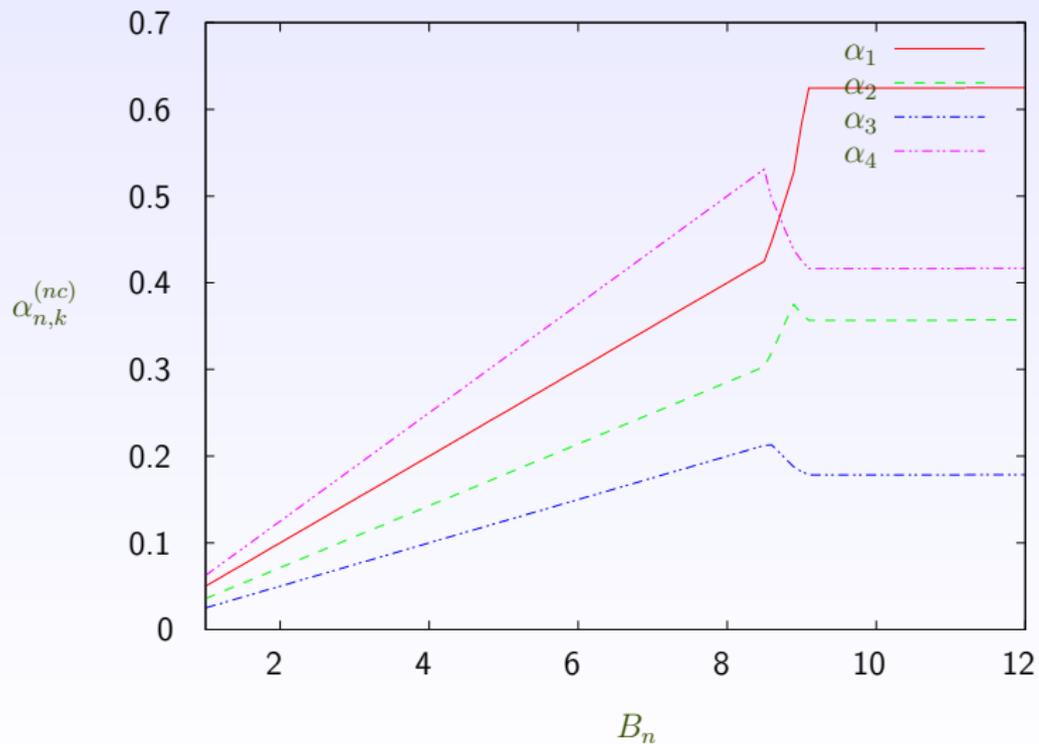
Exemple : 2 machines  $\begin{cases} B_1 = 50 & B_2 = 100 \\ W_1 = 100 & W_2 = 50 \end{cases}$

2 applications  $\begin{cases} b_1 = 1 & b_2 = 2 \\ w_1 = 2 & w_2 = 1 \end{cases}$

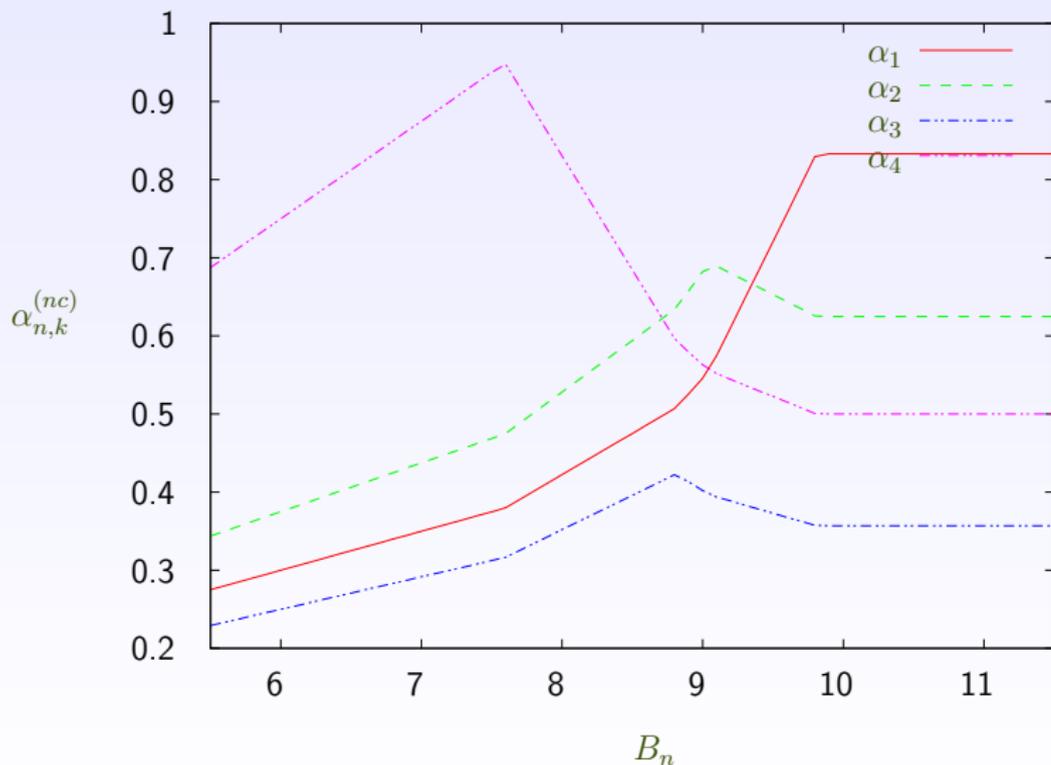
Avec collaboration :  $\alpha_1 = \alpha_2 = 50$

Sans collaboration :  $\alpha_1 = \alpha_2 = 37.5$

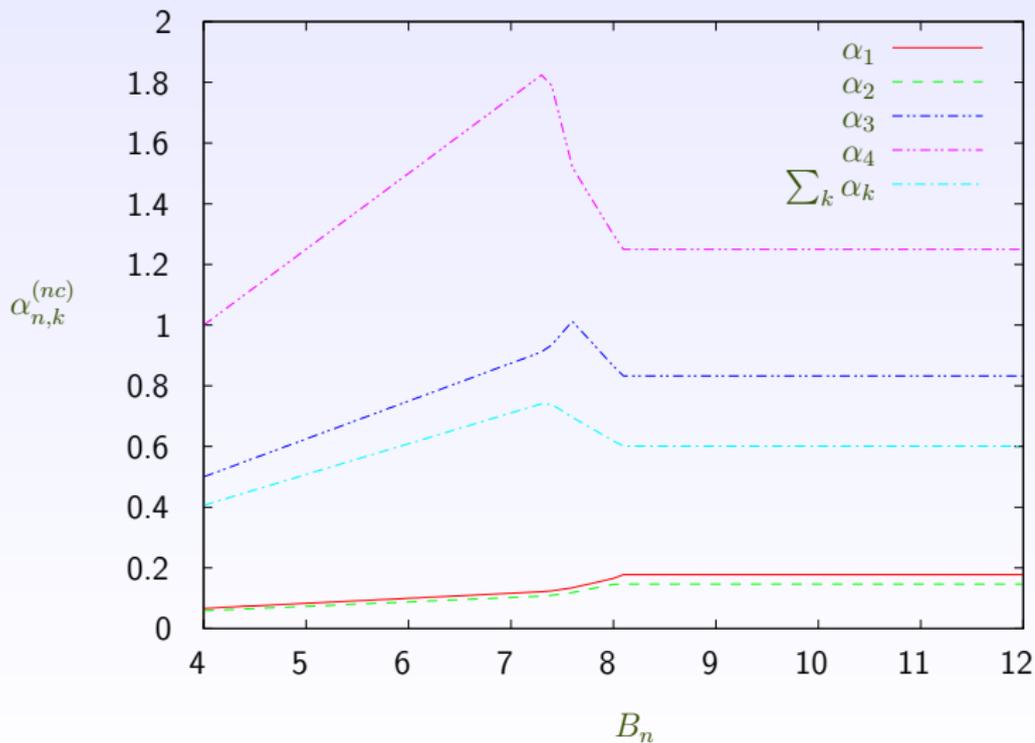
## Débit minimal



## Débit maximal

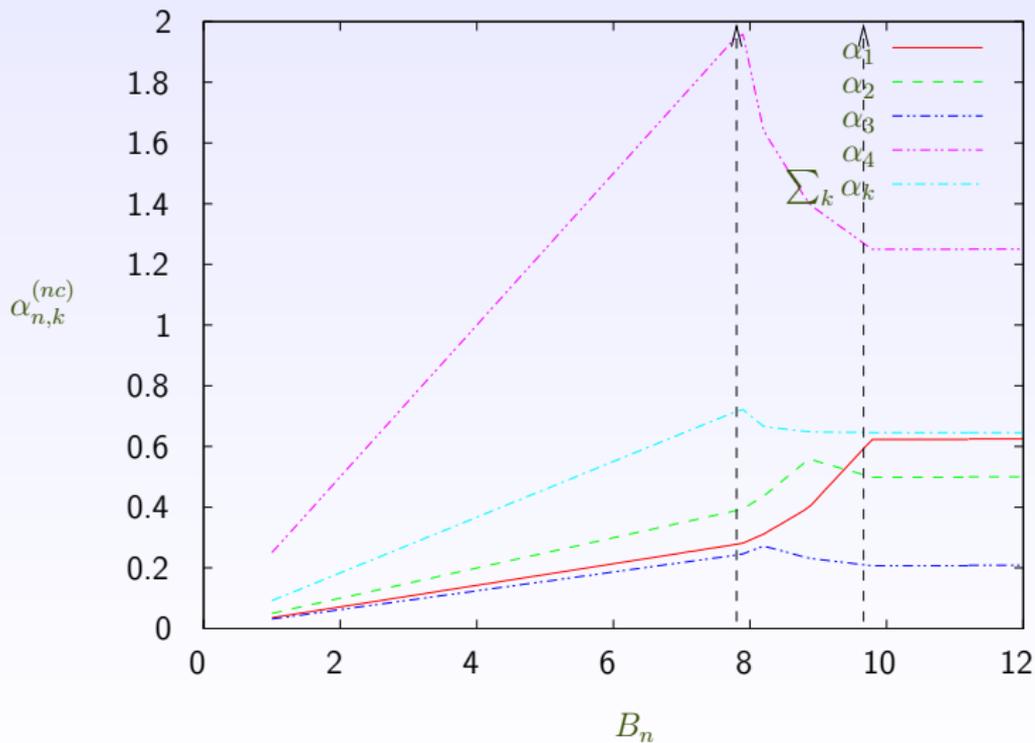


## Performance moyenne



# Résultats

Toutes les mesures à la fois !



Quid des paradoxes de Braess ?

Et les systèmes plus complexes ?