Agrégation optimale de sessions multicast

Nathalie Faure,

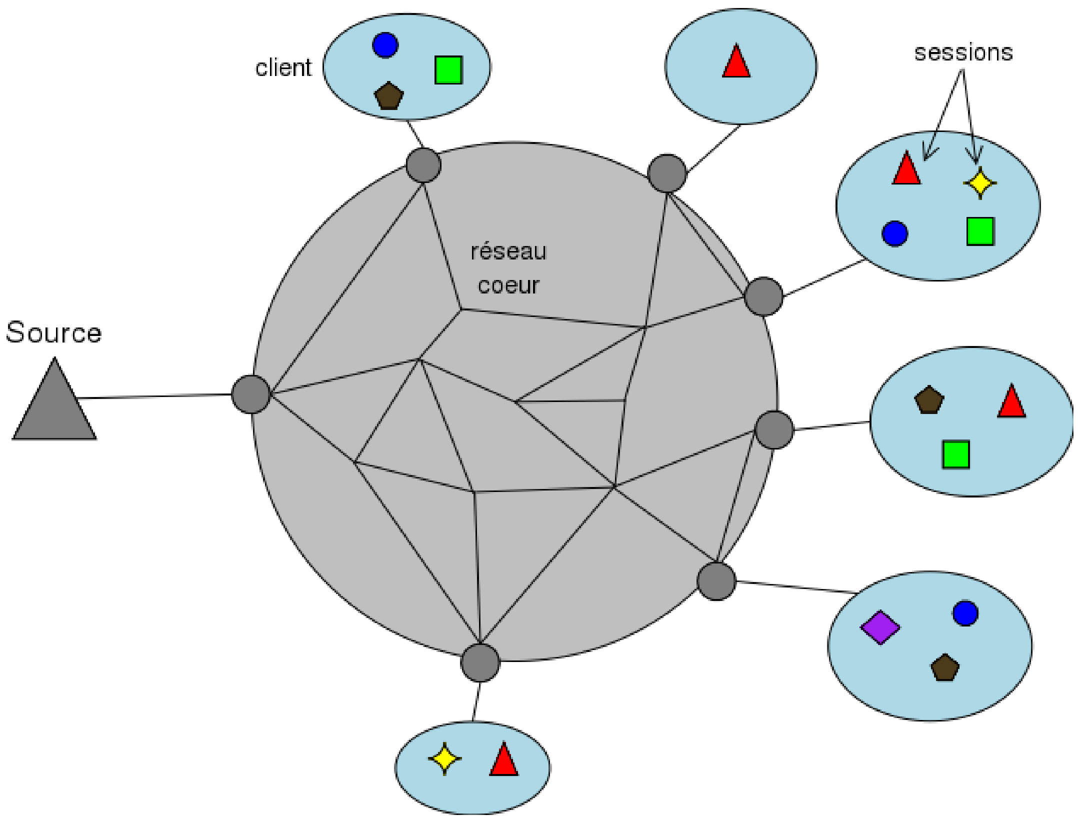
France Télécom R&D, LIP6

Plan

- Introduction au problème
- Modélisations du problème
  - PLNE et Renforcement
  - Énumération explicite
- Utilisation de la méthode de génération de colonnes
- Résultats expérimentaux
- Conclusions

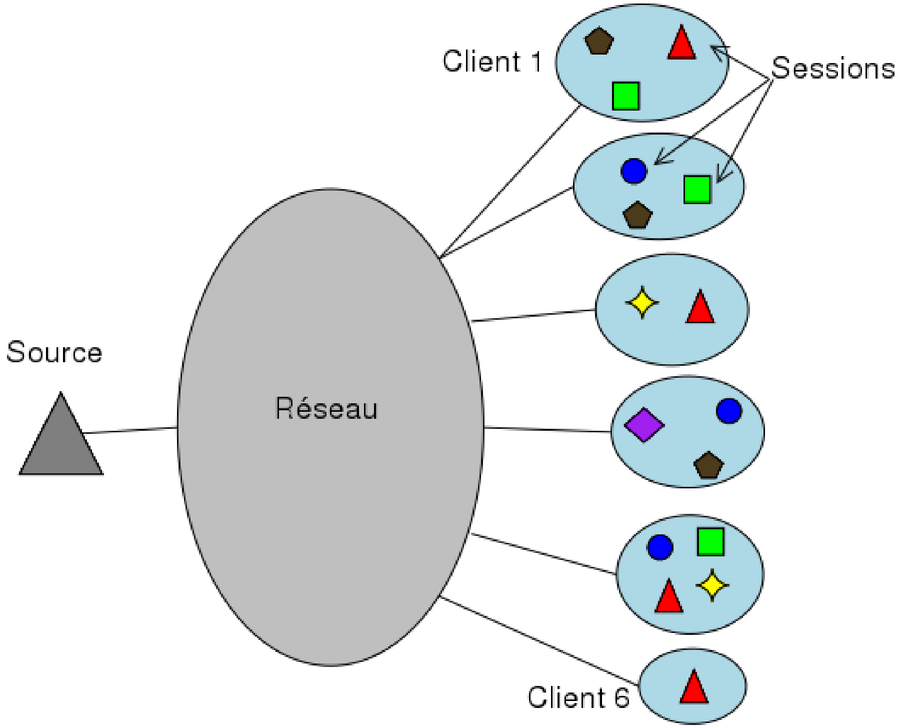
Sessions multicast

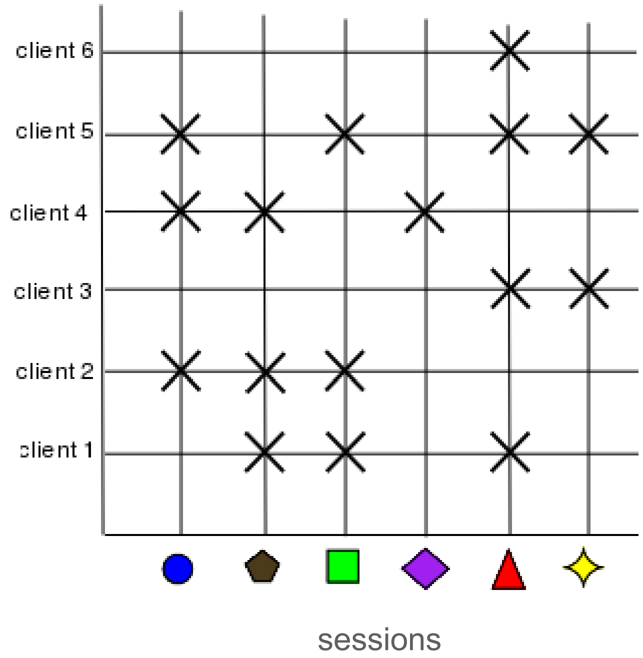
- **Session** : Un contenu et son ensemble de clients
- Exemple : 6 sessions et 6 clients



Réseaux étoilés

- Le cœur de réseau est agrégé et simplifié
- On ne considère que les feuilles : relation directe avec la source



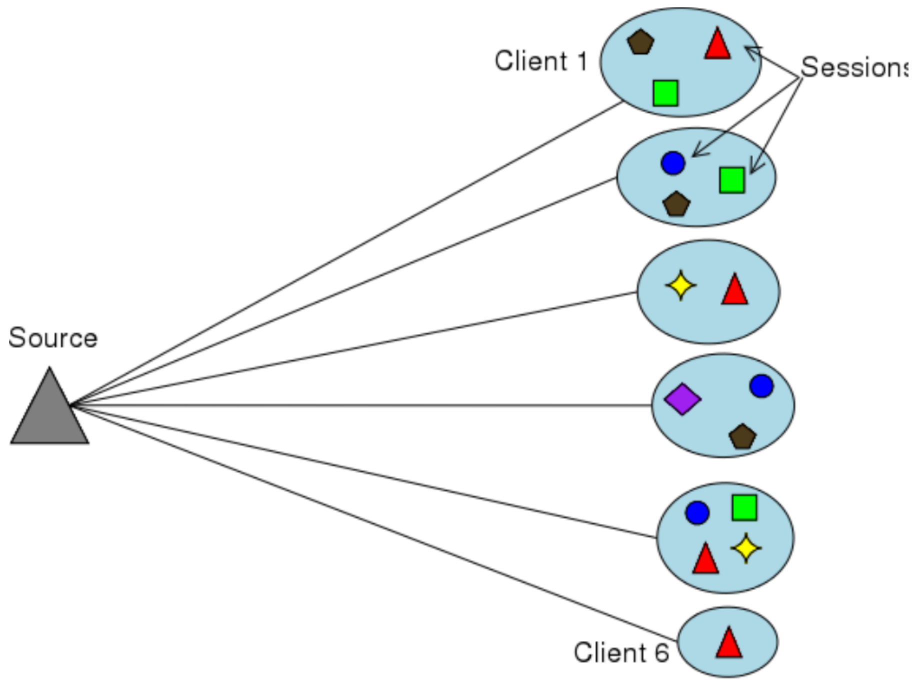


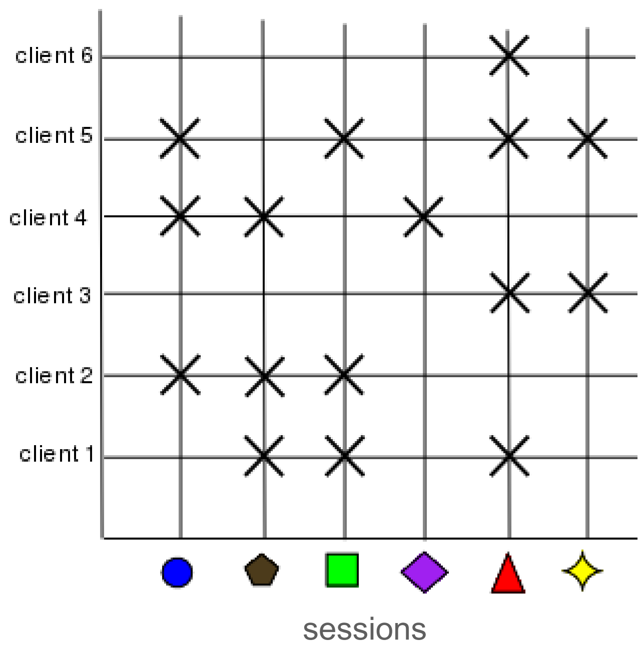
Partitionnement de sessions

- Beaucoup de sessions à router mais peu d'arbres pour le faire
  => Impossibilité d'utiliser un arbre par session
- Plusieurs sessions différentes par arbre => Information envoyée inutilement!!
- Problème d'optimisation: Partitionner les sessions de manière à minimiser le nombre d'informations envoyées inutilement
- Problème NP-difficile

Exemple de partitionnement de sessions

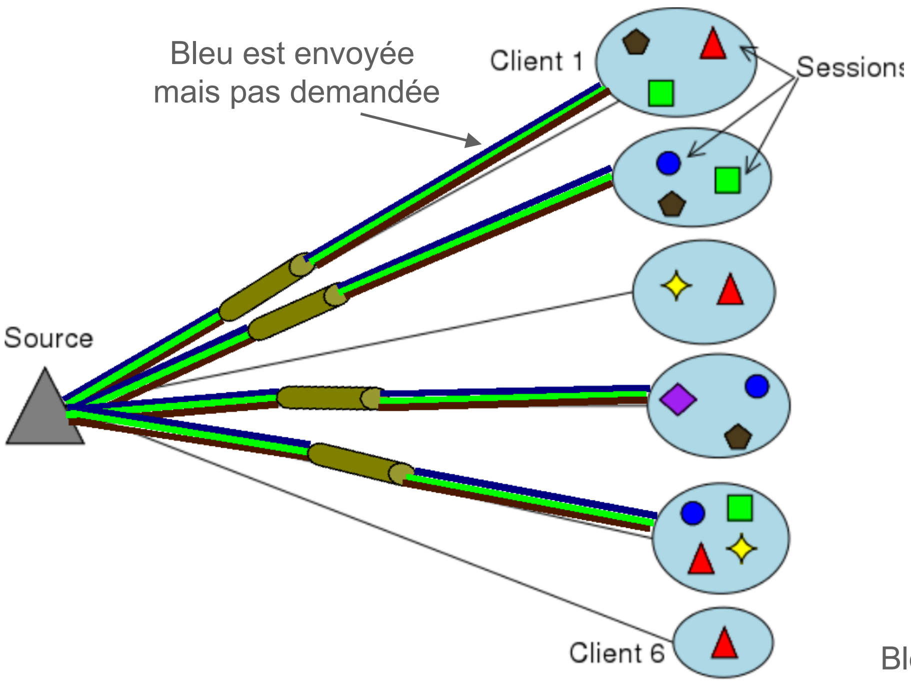
➔ 6 sessions à partitionner dans 3 arbres

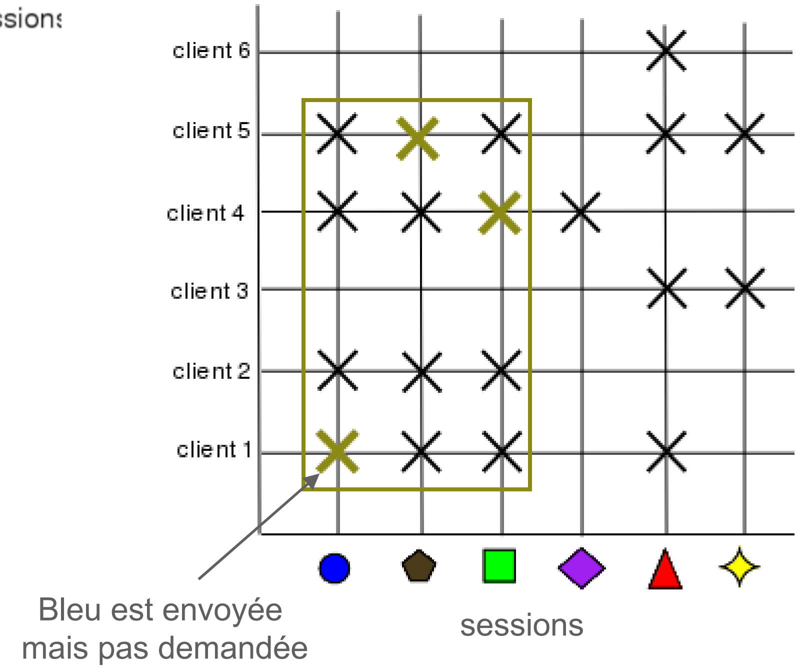




Exemple de partitionnement de sessions

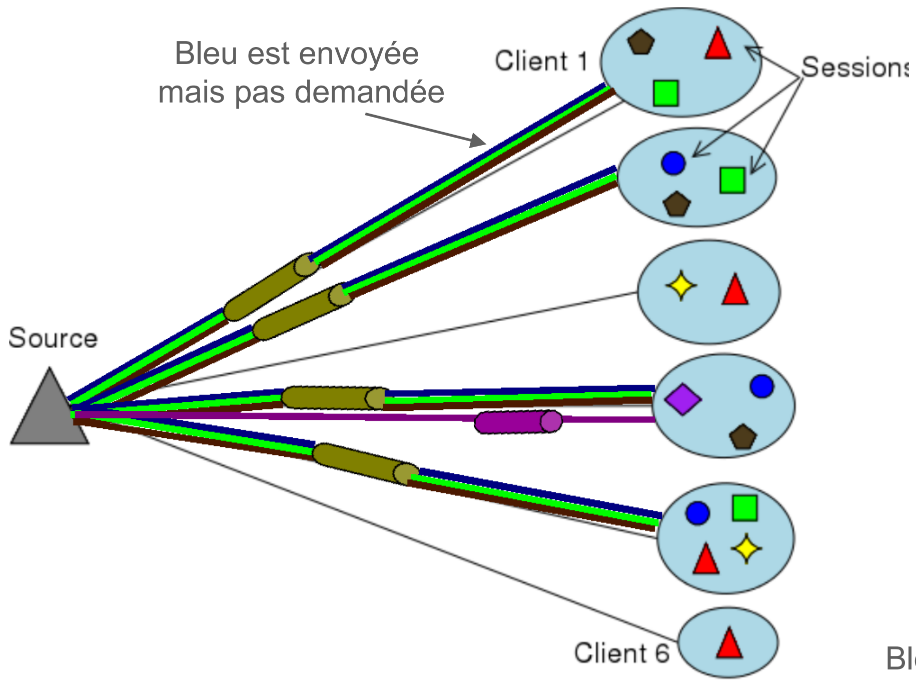
➔ 6 sessions à partitionner dans 3 arbres

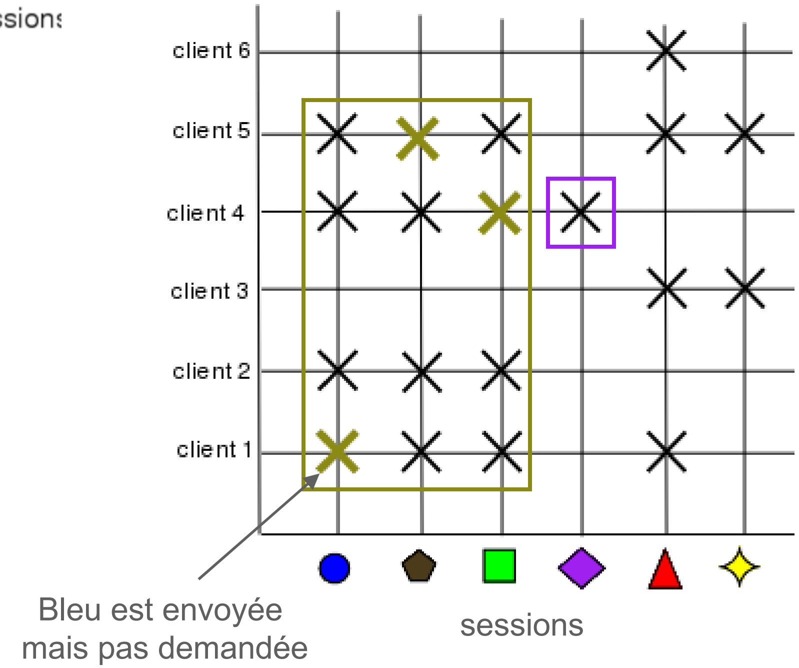




Exemple de partitionnement de sessions

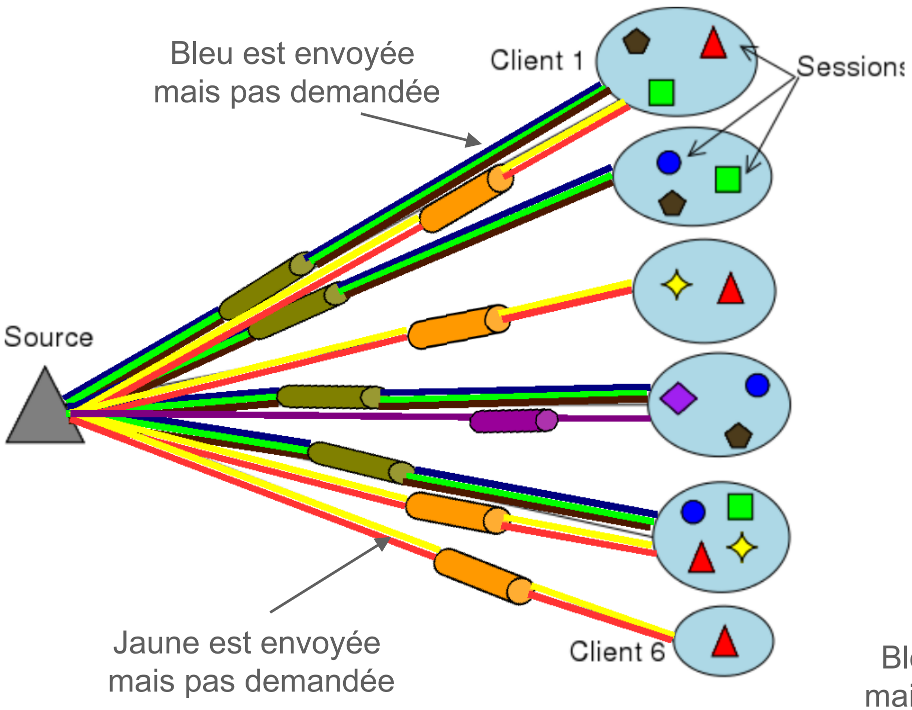
➔ 6 sessions à partitionner dans 3 arbres

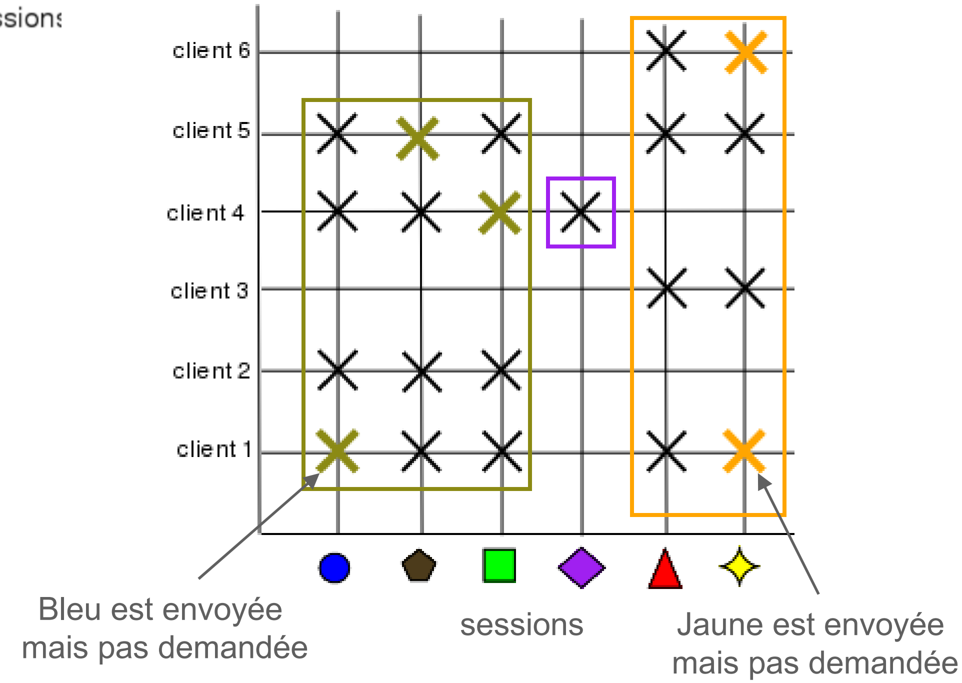




Exemple de partitionnement de sessions

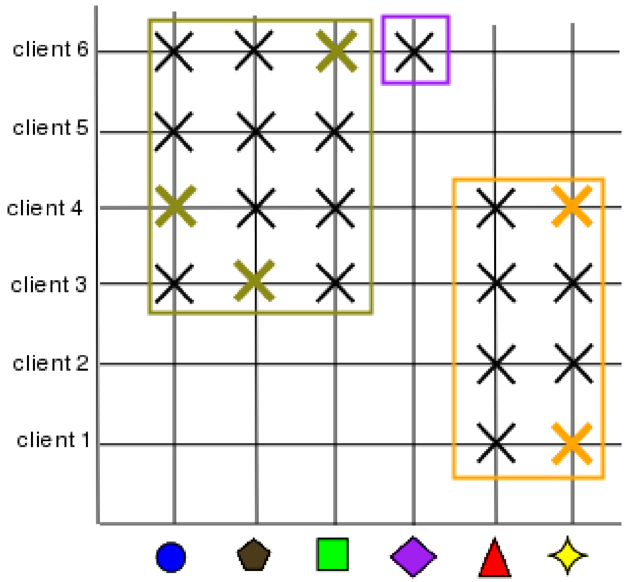
➔ 6 sessions à partitionner dans 3 arbres => surcoût égal à 5





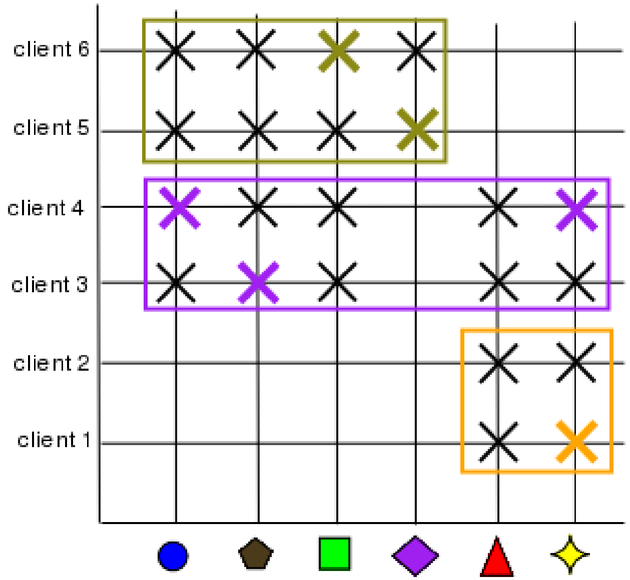
Différentes stratégies d'agrégation

➔ Partitionnement de sessions, de clients ou regroupement libre



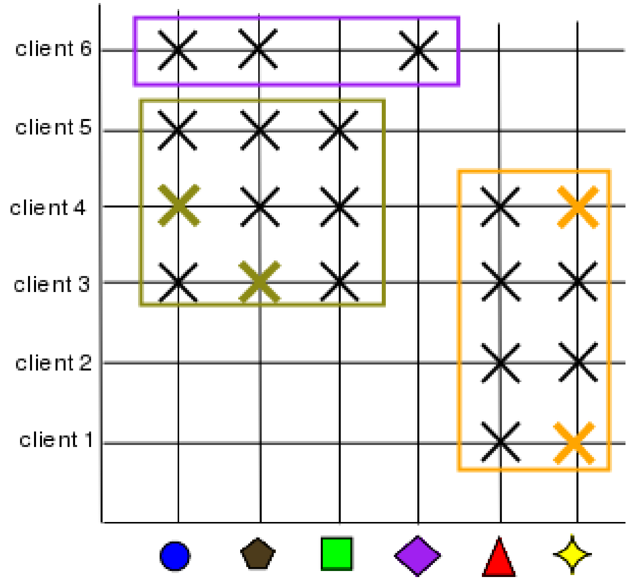
Stratégie partition de sessions

Surcoût = 5



Stratégie partition de clients

Surcoût = 6



Stratégie regroupement libre

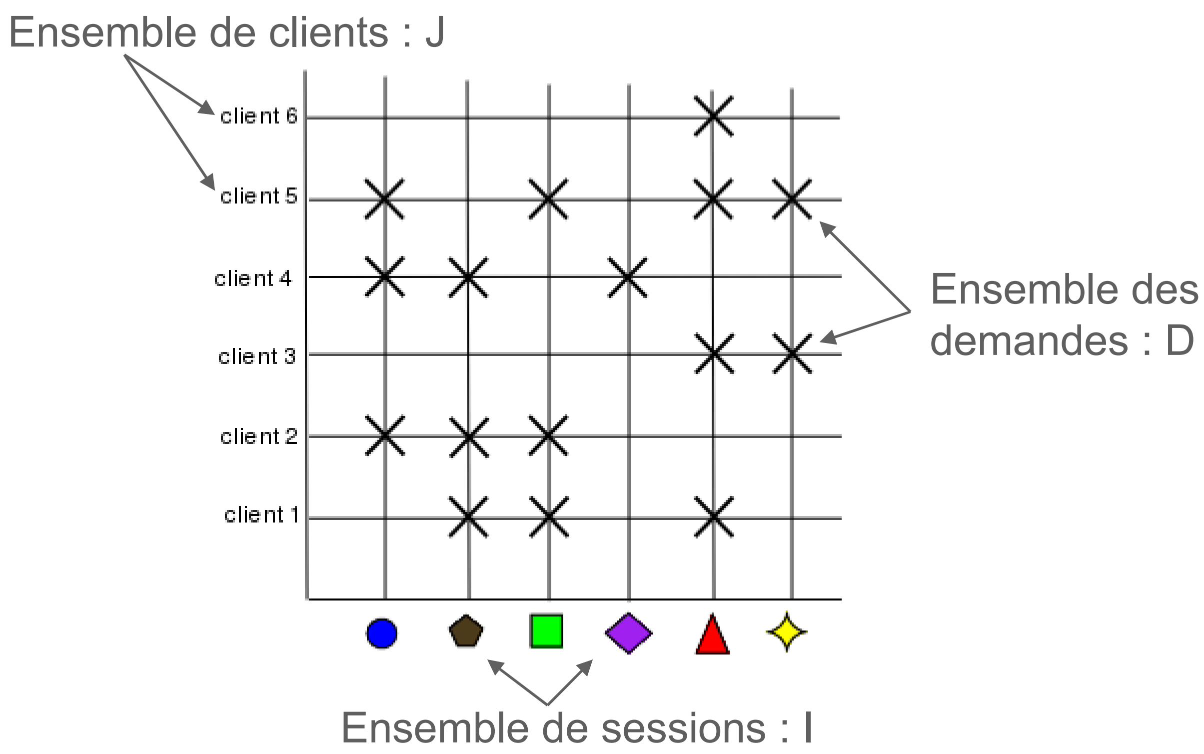
Surcoût = 4

Plan

- Introduction au problème
- Modélisations du problème
  - PLNE et Renforcement
  - Énumération explicite
- Utilisation de la méthode de génération de colonnes
- Résultats expérimentaux
- Conclusions

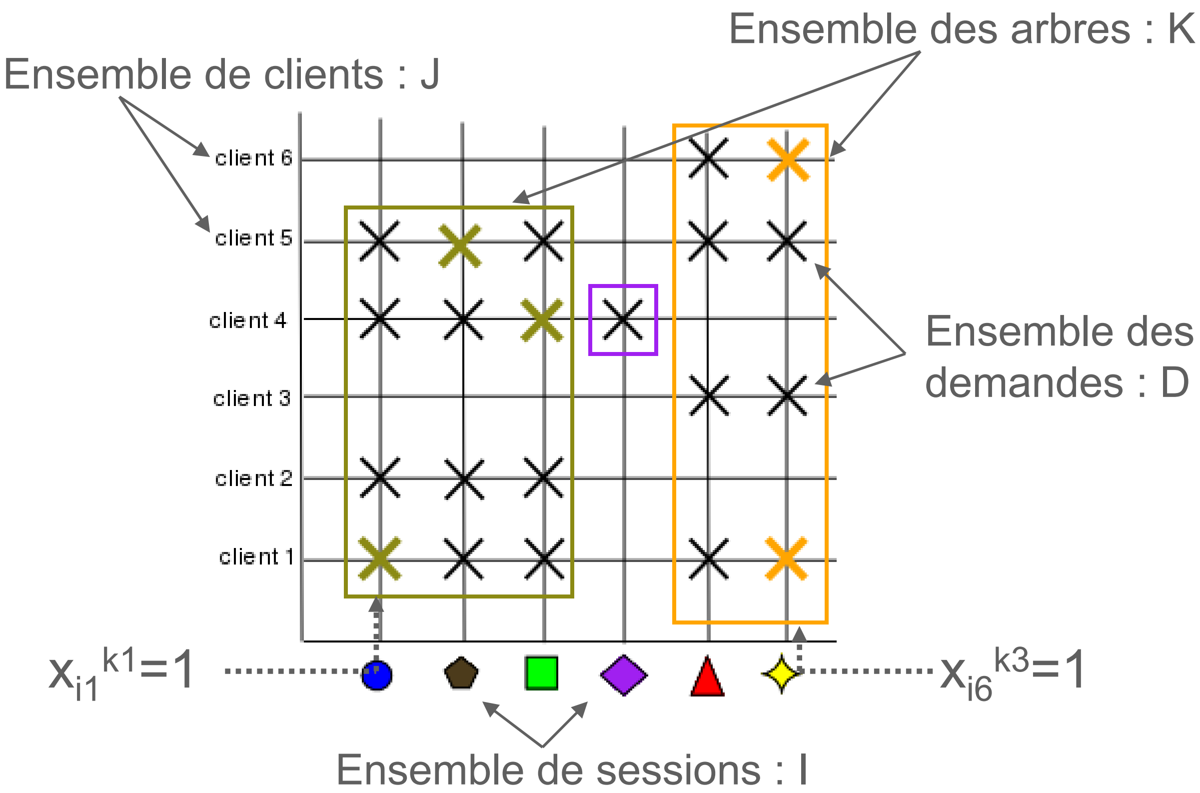
Formulation 1 : notations

➔ Donnée : une demande d entre une session i et un client j



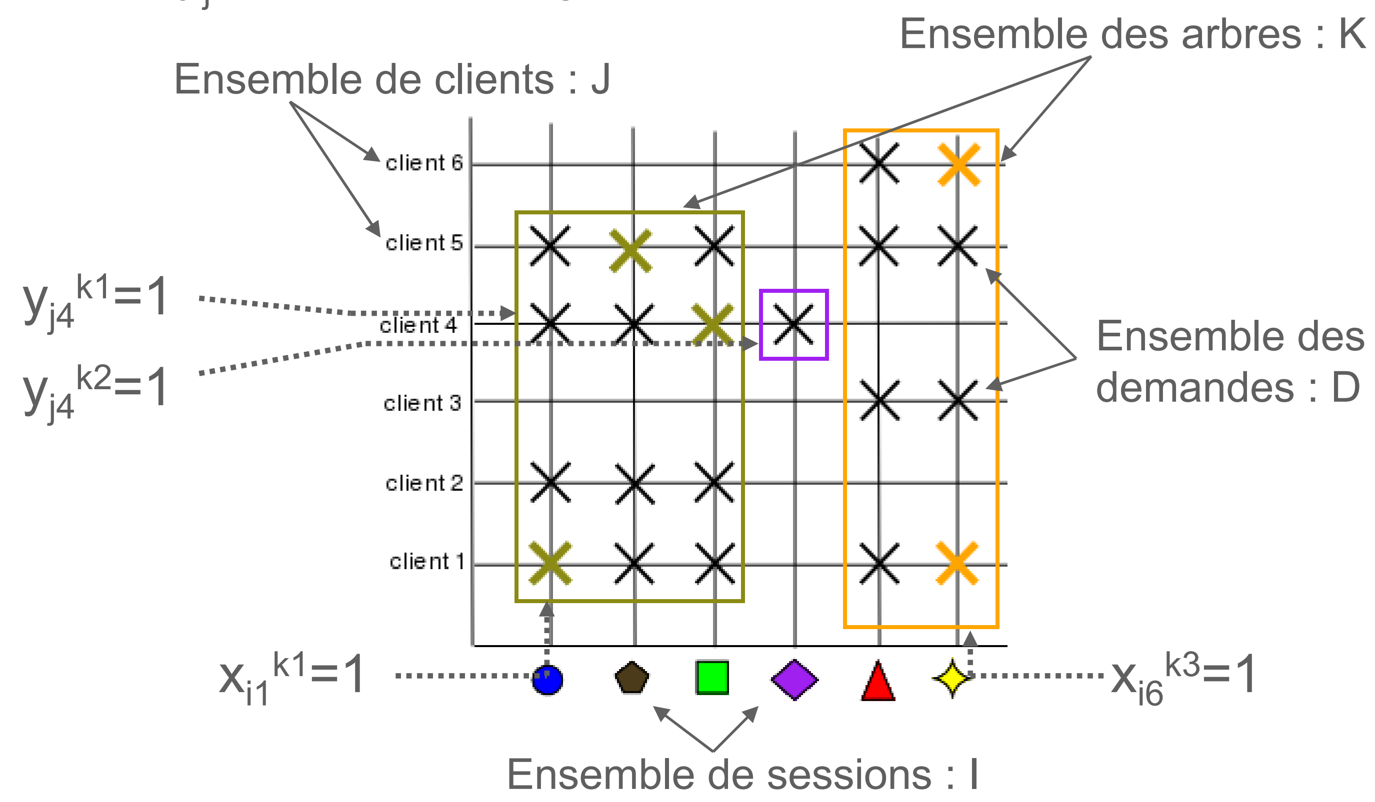
Formulation 1 : notations

➔ Variable x_i^k=1ssi la session i est dans l'arbre k



Formulation 1 : notations

- Variable x_i^k=1ssi la session i est dans l'arbre k
- Variable y_j^k=1ssi le client j est dans l'arbre k



Formulation 1

- Regroupement libre

Minimisation du trafic inutile

Satisfaire les demandes

$$z^* = Min \sum_{k\in K}\sum_{i\in I}\sum_{j\in J} x_i^k \cdot y_j^k$$

$$s.t. \quad \sum_{k\in K} x_i^k \cdot y_j^k \geq 1, \quad i\in I, j\in J, (i,j)\in D,$$

$$x_i^k, y_j^k \in \{0,1\}, \quad i\in I, j\in J, k\in K.$$

$x_i^k = 1$ *ssi* *la session* i *appartient à l'arbre* k,

$y_j^k = 1$ *ssi* *le client* j *appartient à l'arbre* k.

- Partitionnement de sessions : $\sum_{k\in K} x_i^k = 1, \quad i\in I.$

- Partitionnement de clients : $\sum_{k\in K} y_j^k = 1, \quad j\in J.$

Formulation 1 : linéarisation
clusters de session

- Linéarisation : $z_{i,j}^k = x_i^k \cdot y_j^k, \quad i \in I,\ j \in J, k \in K.$
- Donnée : $D^c = (I \times J) \setminus D$

$$
\begin{aligned}
& z^* = Min \sum_{k \in K} \sum_{(i,j) \in D^c} z_{i,j}^k && \text{Minimisation du trafic inutile} \\
s.c.\ & \sum_{k \in K} z_{i,j}^k = 1, \quad (i,j) \in D, && \text{Satisfaire les demandes} \\
& \sum_{k \in K} x_i^k = 1, \quad i \in I, && \text{Clusters de sessions} \\
& z_{i,j}^k \le x_i^k, \quad (i,j) \in D, k \in K, && \\
& z_{i,j}^k \le y_j^k, \quad (i,j) \in D, k \in K, && \text{Linéarisation} \\
& z_{i,j}^k \ge x_i^k + y_j^k - 1, \quad (i,j) \in D^c, k \in K, && \\
& x_i^k, y_j^k, z_{i,j}^k \in \{0,1\}, \quad i \in I, j \in J, k \in K.
\end{aligned}
$$

Formulation 1 : renforcement
clusters de session

- ➔ Propriété d'intégrité : relaxation possible des variables y_j^k et z_{ij}^k dans [0,1]
- ➔ Clusters de sessions : tous les clients d'une session sont dans le même arbre qu'elle

$$x_i^k \leq y_j^k, \quad (i,j) \in D, k \in K,$$

- ➔ Classification des arbres : $\sum_{i \in I} x_i^k \geq \sum_{i \in I} x_i^{k+1}, \quad k = 1,\ldots, p-1,$
  - Borne supérieure et inférieure sur la taille des arbres

- ➔ Monotonie : au moins une session par arbre

Plan

- Introduction au problème
- Modélisations du problème
  - PLNE et Renforcement
  - Énumération explicite
- Utilisation de la méthode de génération de colonnes
- Résultats expérimentaux
- Conclusions

Formulation 2 : notations

- Ensemble T : sous-ensembles de tous les regroupements de sessions possibles
- Exemple pour 3 sessions s_1, s_2, s_3 :
  - $T = \{(s_1),(s_2),(s_3),(s_1, s_2),(s_1,s_3), (s_2,s_3), (s_1, s_2,s_3)\}$
- Cardinalité de T : $2^{|S|} - 1$

- Matrice binaire $A=(a_{s,t})_{s\in S,t\in T}$: appartenance des sessions s aux sous-ensembles t
- Vecteur de coût c : surplus de chaque sous-ensemble t
- Variable binaire $z^t = 1$ ssi le sous-ensemble t est choisi

Formulation 2

$$
\begin{aligned}
z^* = \min \sum_{t \in T} c^t z^t, \\
s.c.: \sum_{t \in T} a_{s,t} \cdot z^t \geq 1, \quad s \in S, \\
\sum_{t \in T} z^t \leq p, \\
z^t \in \{0,1\}, \quad t \in T.
\end{aligned}
$$

Minimisation du coût des sous-ensembles choisis

Couvrir toutes les sessions

Nombre de sous-ensembles limité

- Énumération explicite possible jusqu'à 19 sessions
- Mémoire insuffisante dès 20 sessions (2^{20} variables)
- Intérêt du modèle par rapport à une formulation sans énumération (précédemment étudiée) : mettre en œuvre la génération de colonnes

Plan

- Introduction au problème
- Modélisations du problème
  - PLNE et Renforcement
  - Énumération explicite
- Utilisation de la méthode de génération de colonnes
- Résultats expérimentaux
- Conclusions

Algorithme de génération de colonnes

➔ Initialisation d'un sous-ensemble T' de T

Algorithme de génération de colonnes

- Initialisation d'un sous-ensemble T' de T
- Résolution du PL en continu sur T'

Algorithme de génération de colonnes

- Initialisation d'un sous-ensemble T' de T
- Résolution du PL en continu sur T'
- Expression du sous problème à l'aide des variables duales (u_s et v)
- Résolution exacte du sous problème pour trouver la colonne à ajouter

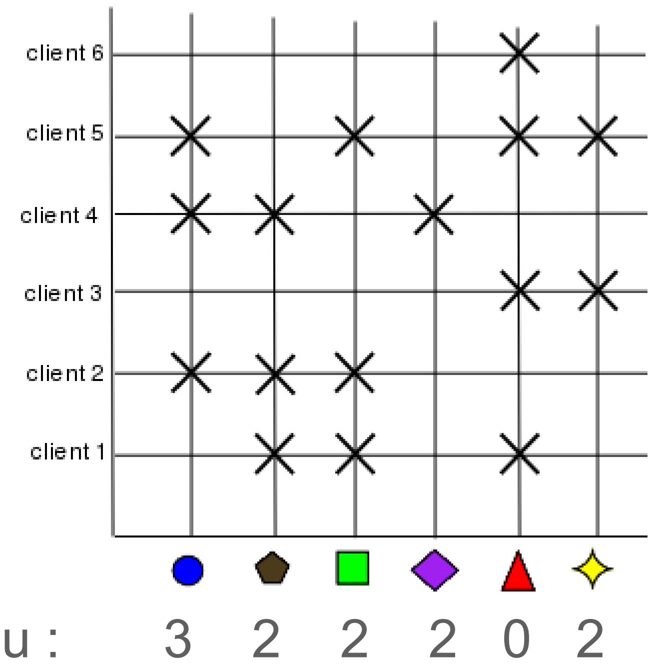
Sous problème

- Entrée : vecteur u (dual calculé)
- Nouvelles variables :
  - Variable binaire x_s : choix de la session s
  - Variable binaire y_i : choix du client i
  - Variable binaire de coût $c_{i,s}$: 1 si surplus engendré
- Recherche du sous ensemble (= groupe de sessions) de coût réduit minimum

$$
\begin{aligned}
&\Phi = Min \sum_{i \in I, s \in S} c_{i,s} - \sum_{s \in S} u_s x_s \\
&s.c.\ \ x_s \le y_i, \qquad\qquad\qquad (i,s)\ demande, \\
&\qquad c_{i,s} \ge x_s + y_i - 1, \quad (i,s)\ non\ demande, \\
&\qquad c_{i,s}, x_s, y_i \in \{0,1\},\ \ i \in I,\ s \in S.
\end{aligned}
$$

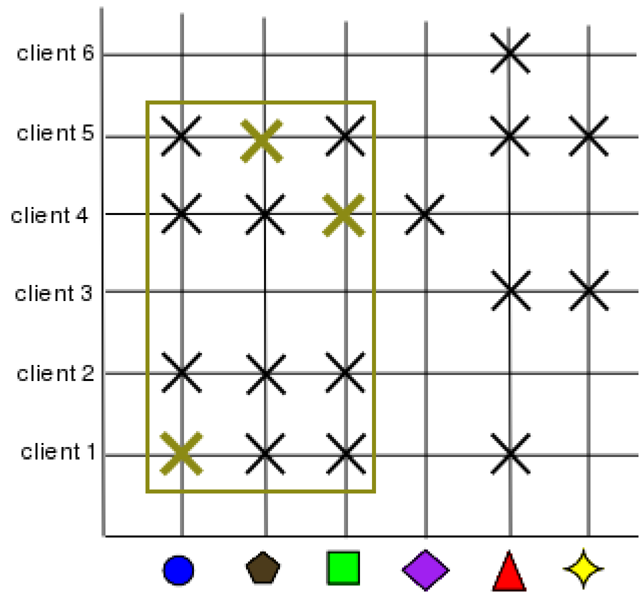
Sous problème : exemple

➔ 6 sessions, 6 clients



Sous problème : exemple

➔ 6 sessions, 6 clients



u : 3 2 2 2 0 2

➔ $\Phi = 3 - 3 - 2 - 2 = -4$

Algorithme de génération de colonnes

- Initialisation d'un sous-ensemble T' de T
- Résolution du PL en continu sur T'
- Expression du sous problème à l'aide des variables duales (u_s et v)
- Résolution exacte du sous problème pour trouver la colonne à ajouter
- Si Φ+v < 0 : ajout du nouveau sous-ensemble dans T'

Algorithme de génération de colonnes

- Initialisation d'un sous-ensemble T' de T
- Résolution du PL en continu sur T'
- Expression du sous problème à l'aide des variables duales (u_s et v)
- Résolution exacte du sous problème pour trouver la colonne à ajouter
- Si Φ+v < 0 : ajout du nouveau sous-ensemble dans T'

Algorithme de génération de colonnes

- ➔ Initialisation d'un sous-ensemble T' de T
- ➔ Résolution du PL en continu sur T'
- ➔ Expression du sous problème à l'aide des variables duales (u_s et v)
- ➔ Résolution exacte du sous problème pour trouver la colonne à ajouter
- ➔ Si Φ+v < 0 : ajout du nouveau sous-ensemble dans T'
- ➔ Sinon : Résolution du PL en nombre entier sur le dernier T'

Algorithme de génération de colonnes

- Initialisation d'un sous-ensemble T' de T
- Résolution du PL en continu sur T'
- Expression du sous problème à l'aide des variables duales (u_s et v)
- Résolution exacte du sous problème pour trouver la colonne à ajouter
- Si Φ+v < 0 : ajout du nouveau sous-ensemble dans T'
- Sinon : Résolution du PL en nombre entier sur le dernier T'
- Comparaison et estimation de la solution :
  - Borne supérieure : Résolution du PL en nombre entier
  - Borne inférieure : Dernière résolution du PL en continu

Plan

- Introduction au problème
- Modélisations du problème
  - PLNE et Renforcement
  - Énumération explicite
- Utilisation de la méthode de génération de colonnes
- Résultats expérimentaux
- Conclusions

Expérimentations

- Implémentation dans Xpress-MP 2005
- Instances générées aléatoirement - Moyenne sur 50 instances
- |I| = nb de clients, |S| = nb de sessions, p = nb d'arbres

Résolution / Problème |I|, |S| ->p	Formulation 1 renforcée	Formulation 2 énumération explicite	Génération de colonnes	Instances optimales	Erreur sur les instances non optimales
10, 10 ->6	7.9s	0.1s	0.1s	100 %	-
15, 15 ->8	> 5h	6s	0.9s	96 %	4%
19, 19 ->10	*	175s	4s	96 %	2%
20, 20 ->10	*	*	7s	90 %	2 %
30, 30 ->15	*	*	3min	83 %	1 %
40, 40 ->20	*	*	16min	70 %	0.4 %

Conclusions

- Problème NP-difficile de partitionnement de sessions multicast
- Résolution exacte jusqu'à 19 sessions
- Résolution approchée par génération de colonnes
  - Optimum atteint très souvent
  - Pourcentage d'erreur très faible
  - Sous problème aussi NP-difficile => Perte de temps pour la résolution exacte