



Méthodes PARAFAC généralisées pour l'extraction aveugle de sources. Application aux systèmes DS-CDMA.

Thèse soutenue par **Dimitri Nion**

Encadrants: Lieven De Lathauwer et Inbar Fijalkow

Financement: DGA

Laboratoire: ETIS, CNRS UMR 8051

Université de Cergy-Pontoise, le 18 octobre 2007

Plan de la présentation

Introduction

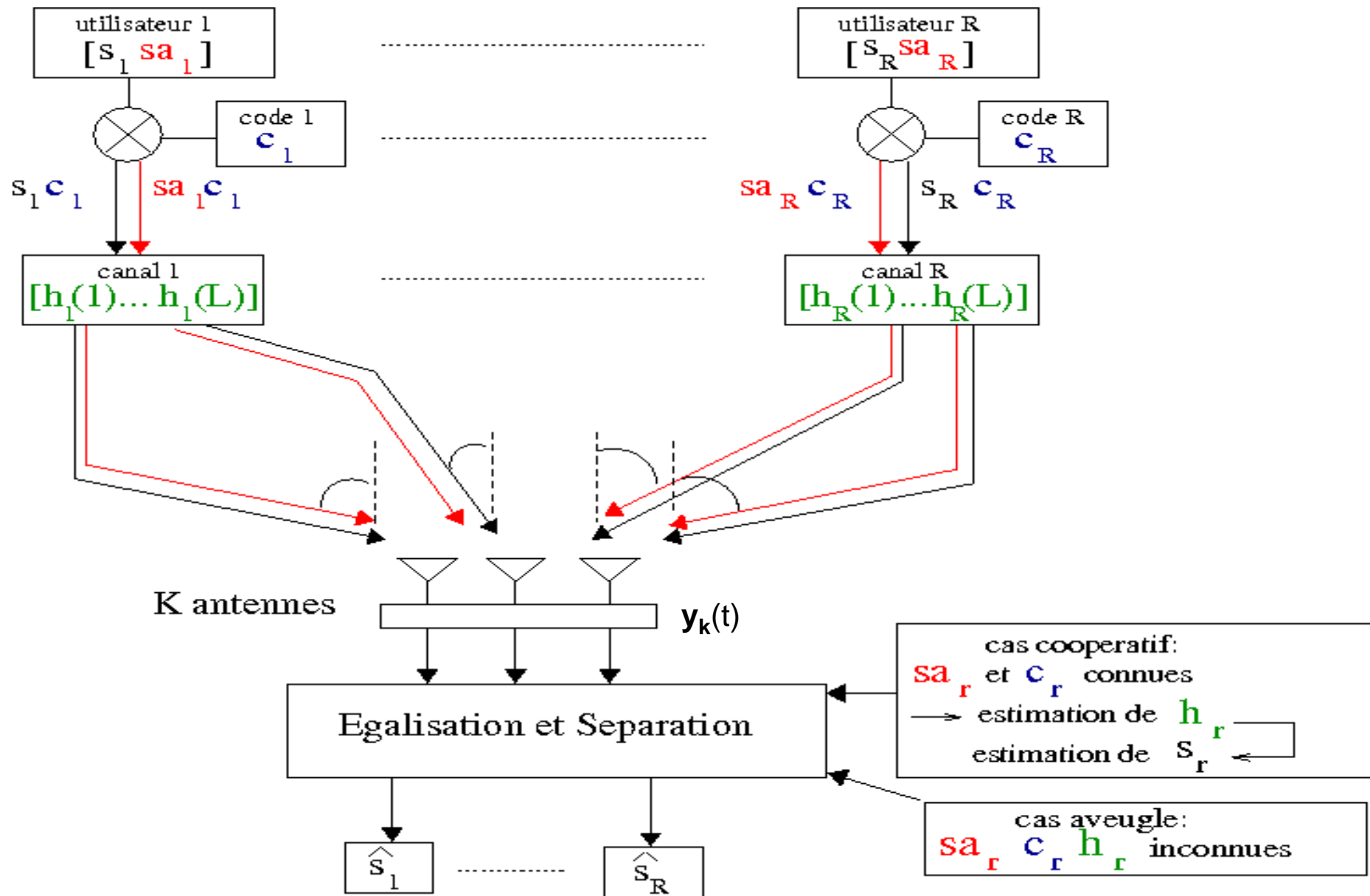
- I. Les décompositions tensorielles
- II. Algorithmes de calcul des décompositions tensorielles
- III. Unicité des décompositions tensorielles

Conclusion et Perspectives

Contexte

- Domaine: Séparation aveugle de sources
- Application: Communications sans fil
- Système: DS-CDMA Multi-Utilisateurs, uplink, réseau d'antennes réceptrices
- Propagation: Trajets multiples engendrant de l'IES
- Hypothèses: R.I. du canal non connue, codes CDMA non connus, angles d'arrivée non connus: approche **Aveugle**
- Objectif: Estimer la séquence de symboles émise par chaque utilisateur
- Approche: Déterministe, s'appuie sur l'algèbre tensorielle
Comment? Stocker les observations dans un tenseur d'ordre 3, puis le **décomposer en une somme de contributions**, une par utilisateur
- Idée sous-jacente: Modèle tensoriel plus riche qu'un modèle matriciel

Système CDMA considéré: coopératif vs. aveugle

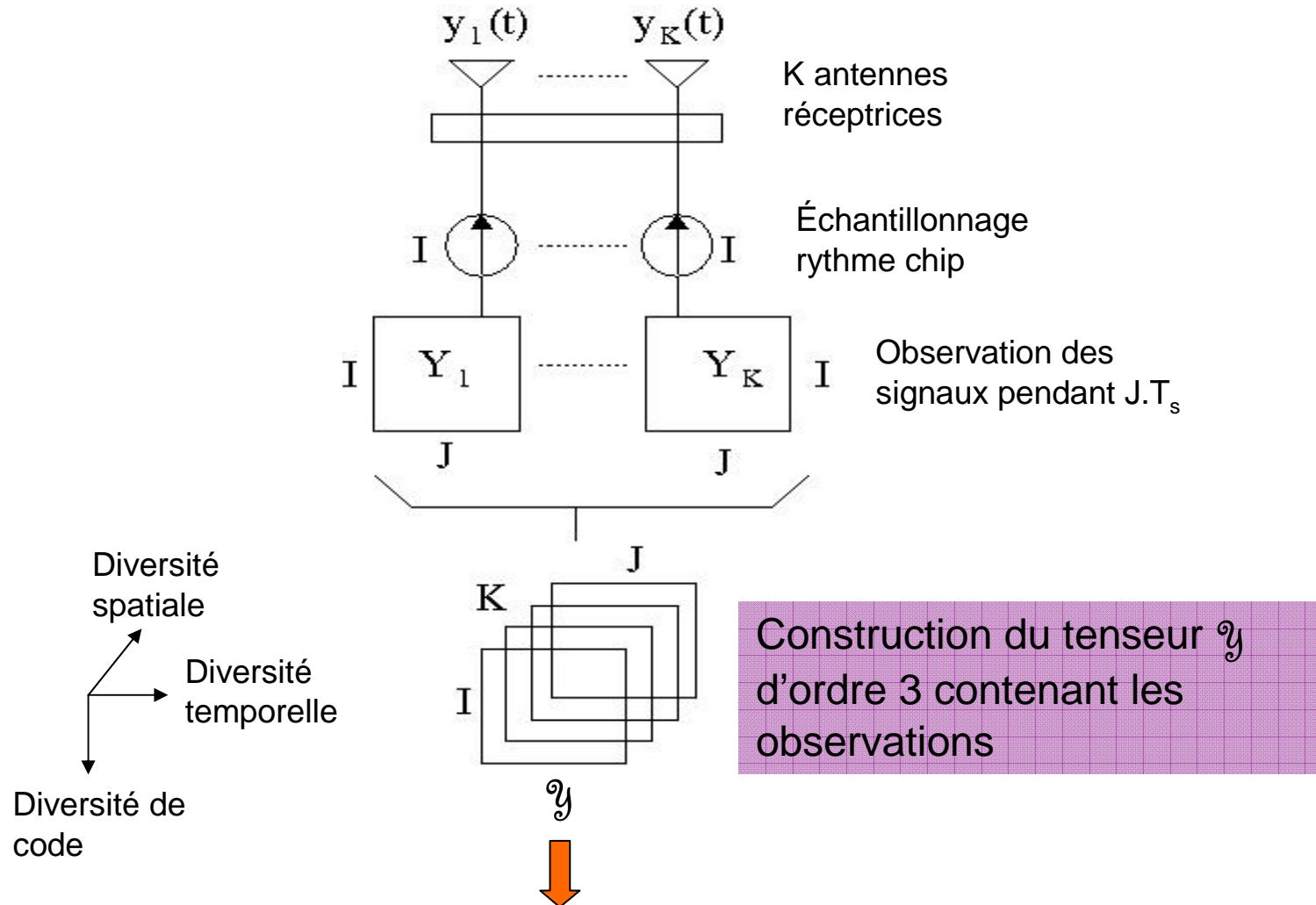


L'approche aveugle: Pourquoi?

Plusieurs motivations:

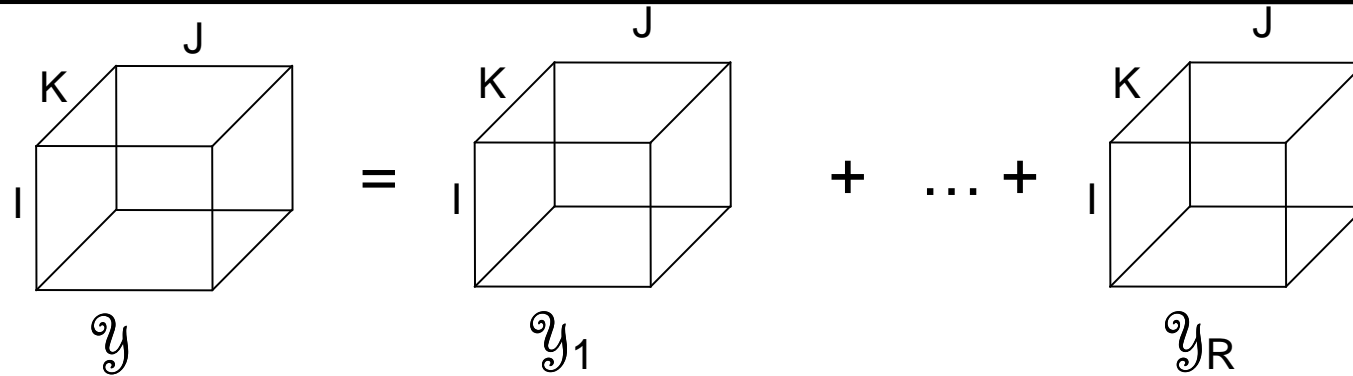
- ➔ Augmenter le débit 'utile' (séquence d'apprentissage=jusqu'à 50% du débit total en UMTS)
- ➔ Interception de communications pour écoute discrète
- ➔ Localisation de source
- ➔ Si la séquence d'apprentissage est indisponible ou partiellement reçue
- ➔ Si le canal est très sélectif en fréquence, la séquence d'apprentissage en subit les effets
- ➔ Méthodes aveugles robustes → méthodes semi-aveugles robustes

L'approche aveugle: Comment?

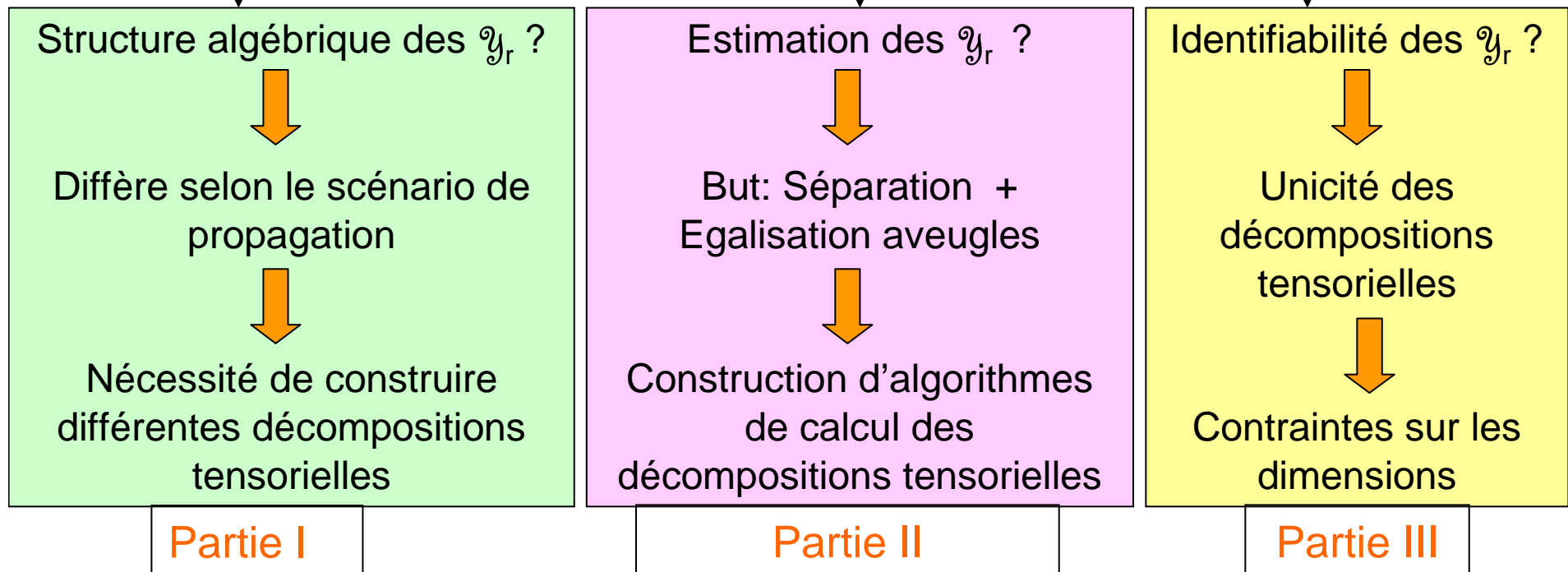


Traitement Numérique: Egalisation et Séparation aveugle par décomposition de \mathcal{Y}

L'approche aveugle: Comment? (suite)



Décomposition de y : somme des contributions des R utilisateurs



Introduction

I. Les décompositions tensorielles

1. Trajets directs:

→ Décomposition PARAFAC standard

2. Trajets multiples (réflexions champ lointain des antennes) + IES:

→ Décomposition en Composantes Blocs de rang-(L,L,1) : DCB(L,L,1)

3. Trajets multiples (modèle spéculaire) + IES:

→ Décomposition en Composantes Blocs de rang-(L,P,..) : DCB(L,P,..)

II. Algorithmes de calcul des décompositions tensorielles

III. Unicité des décompositions tensorielles

Conclusion et Perspectives

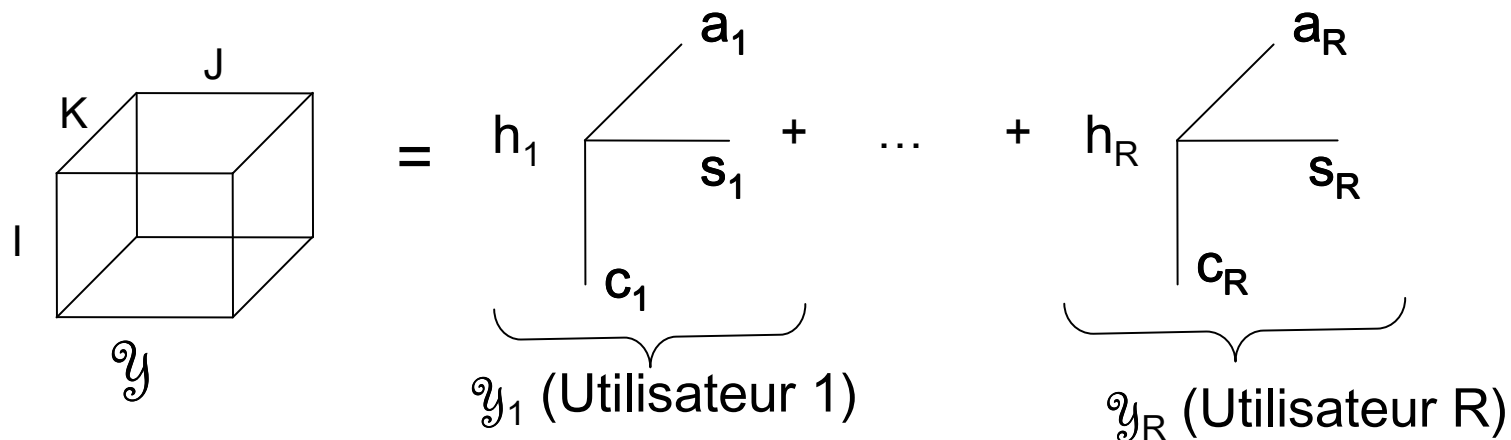
Décomposition PARAFAC

Si **trajets directs uniquement** (mélange instantané), \mathcal{Y} suit une **décomposition PARAFAC** [Sidiropoulos, Giannakis & Bro, 2000].

Modèle analytique:

$$y_{ijk} = \sum_{r=1}^R h_r c_{ir} s_{jr} a_{kr}$$

Modèle algébrique:



\mathbf{c}_r contient les I 'chips' du code d'étalement de l'utilisateur r

\mathbf{a}_r contient la réponse des K antennes à l'angle d'arrivée du trajet direct

\mathbf{s}_r contient les J symboles successifs émis par l'utilisateur r

h_r coefficient d'évanouissement (canal instantané)

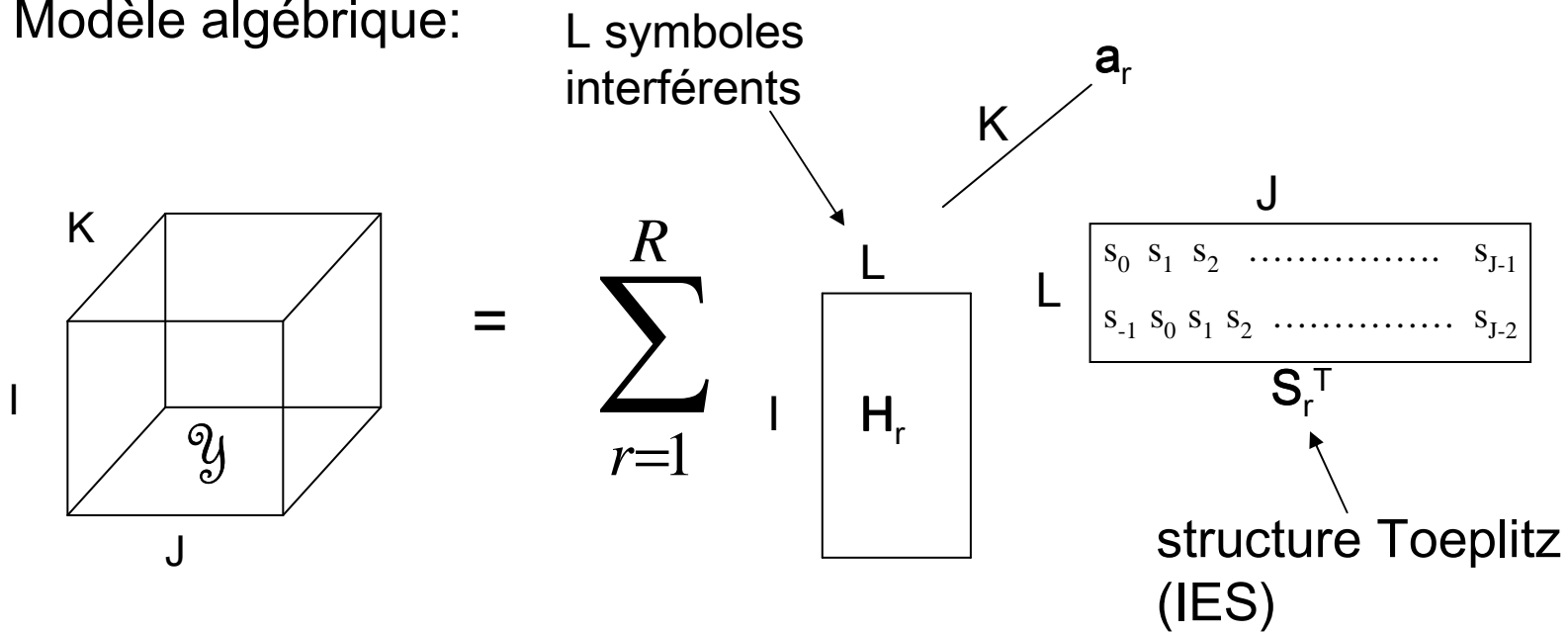
DCB-(L,L,1)

Si trajets multiples dans le champ lointain + IES , \mathcal{y} suit une DCB-(L,L,1)
 [De Lathauwer & De Baynast, 2003], [Nion & De Lathauwer, SPAWC 2007].

Modèle analytique:

$$y_{ijk} = \sum_{r=1}^R a_{kr} \sum_{l=1}^L h_r (i + (l-1)I) s_{j-l+1}^{(r)}$$

Modèle algébrique:



DCB-(L,P,..)

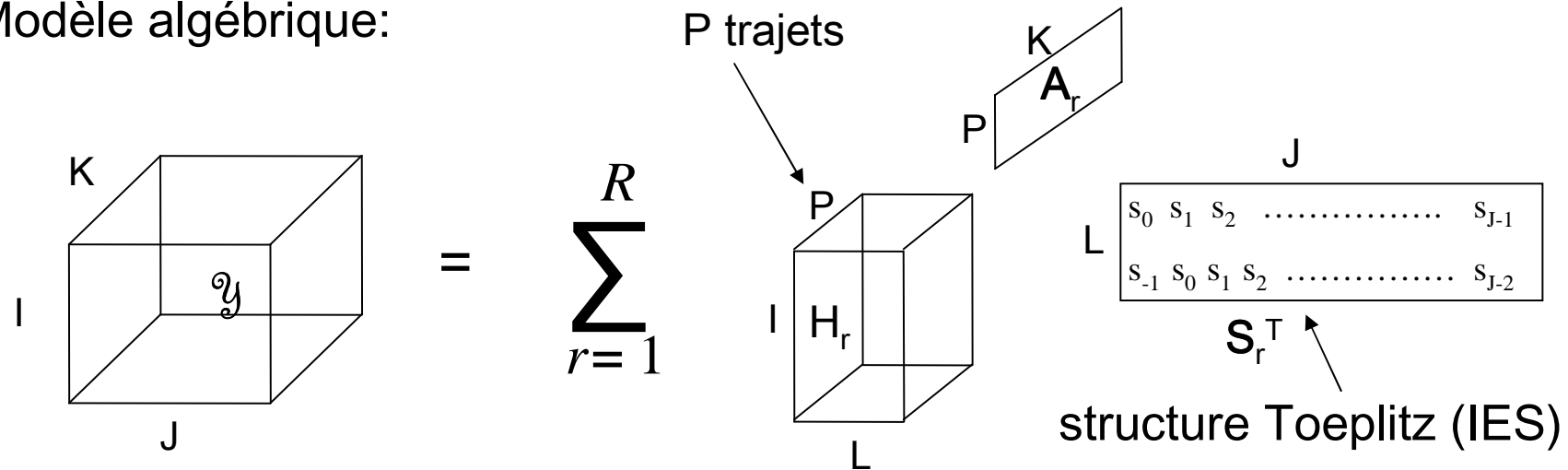
Si **trajets multiples non uniquement dans le champ lointain + IES**, y suit une **DCB-(L,P,..)** [Nion & De Lathauwer, ICASSP 2005].

Modèle analytique:

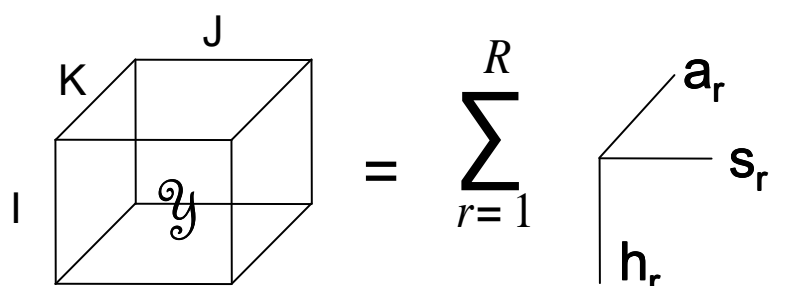
$$y_{ijk} = \sum_{r=1}^R \sum_{p=1}^P a_k(\theta_{rp}) \sum_{l=1}^L h_{rp}(i + (l-1)I) s_{j-l+1}^{(r)}$$

1 trajet = 1 retard, 1 angle d'arrivée et 1 coeff d'évanouissement

Modèle algébrique:

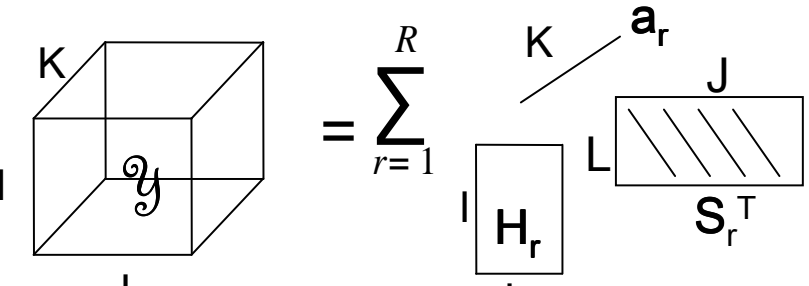


Inconnues pour chaque décomposition



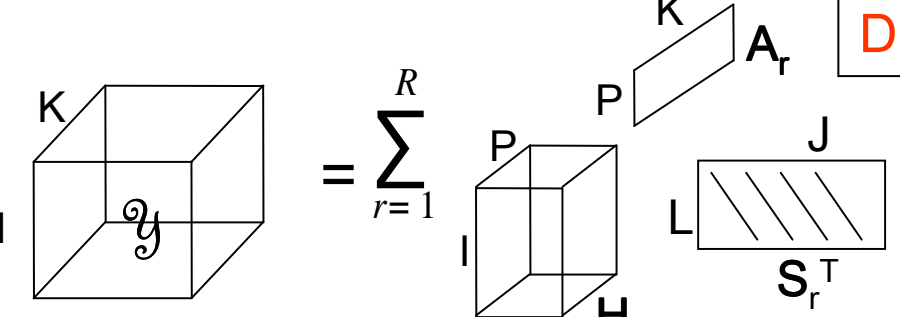
PARAFAC

$$\left\{ \begin{array}{l} \mathbf{H} \in \mathbb{C}^{I \times R} \\ \mathbf{S} \in \mathbb{C}^{J \times R} \\ \mathbf{A} \in \mathbb{C}^{K \times R} \end{array} \right.$$



DCB-(L,L,1)

$$\left\{ \begin{array}{l} \mathbf{H} \in \mathbb{C}^{I \times RL} \\ \mathbf{S} \in \mathbb{C}^{J \times RL} \text{ Block-Toeplitz} \\ \mathbf{A} \in \mathbb{C}^{K \times R} \end{array} \right.$$



DCB-(L,P,..)

$$\left\{ \begin{array}{l} \mathbf{H} \in \mathbb{C}^{I \times RPL} \\ \mathbf{S} \in \mathbb{C}^{J \times RL} \text{ Block-Toeplitz} \\ \mathbf{A} \in \mathbb{C}^{K \times RP} \end{array} \right.$$

Indéterminations des décompositions

PARAFAC

$$\sum_{r=1}^R \begin{matrix} \text{--- } \mathbf{a}_r \\ \text{--- } \mathbf{s}_r \\ \text{--- } \mathbf{h}_r \end{matrix} = \sum_{r=1}^R \frac{1}{\alpha_r \beta_r \gamma_r} \begin{matrix} \text{--- } \gamma_r \mathbf{a}_r \\ \text{--- } \beta_r \mathbf{s}_r \\ \text{--- } \alpha_r \mathbf{h}_r \end{matrix}$$

Si $\alpha_r \beta_r \gamma_r = 1, \forall r \in [1 \dots R]$

alors $\gamma_r \mathbf{a}_r, \beta_r \mathbf{s}_r$ et $\alpha_r \mathbf{h}_r$

sont aussi solutions.

DCB-(L,L,1)

$$\sum_{r=1}^R \begin{matrix} \text{--- } \mathbf{a}_r \\ \text{--- } \mathbf{H}_r \\ \text{--- } \mathbf{S}_r^T \end{matrix} = \sum_{r=1}^R \begin{matrix} \text{--- } \gamma_r \mathbf{a}_r \\ \text{--- } \mathbf{F}_r \mathbf{F}_r^{-1} \mathbf{H}_r \\ \text{--- } \mathbf{S}_r^T \end{matrix}$$

DCB-(L,P,..)

$$\sum_{r=1}^R \begin{matrix} \text{--- } \mathbf{A}_r \\ \text{--- } \mathbf{H}_r \\ \text{--- } \mathbf{S}_r^T \end{matrix} = \sum_{r=1}^R \begin{matrix} \text{--- } \mathbf{G}_r \mathbf{G}_r^{-1} \mathbf{A}_r \\ \text{--- } \mathbf{H}_r \\ \text{--- } \mathbf{F}_r \mathbf{F}_r^{-1} \mathbf{S}_r^T \end{matrix}$$

Conclusion de la Partie I:

Les décompositions tensorielles

- Selon le scénario de propagation, le tenseur des observations suit un modèle algébrique particulier.
- Généralisation des travaux de *[Sidiropoulos et al., 2000]*
PARAFAC \rightarrow DCB-(L,L,1) \rightarrow DCB-(L,P,.).
- Nécessité de construire des algorithmes de calcul des décompositions tensorielles réalisant conjointement la séparation et l'égalisation aveugles.
- Séparation aveugle: estimation de A , S , et H .
Egalisation aveugle: nous choisissons de préserver la structure Toeplitz des S_r .

Introduction

I. Les décompositions tensorielles

II. Algorithmes de calcul des décompositions tensorielles

1. Moindres Carrés Alternés (ALS pour “Alternating Least Squares”)

2. Ajout d’une étape d’interpolation linéaire dans l’ALS

3. L’algorithme de Levenberg-Marquardt (LM)

4. Compression des données

5. Calcul par EVD - Initialisation

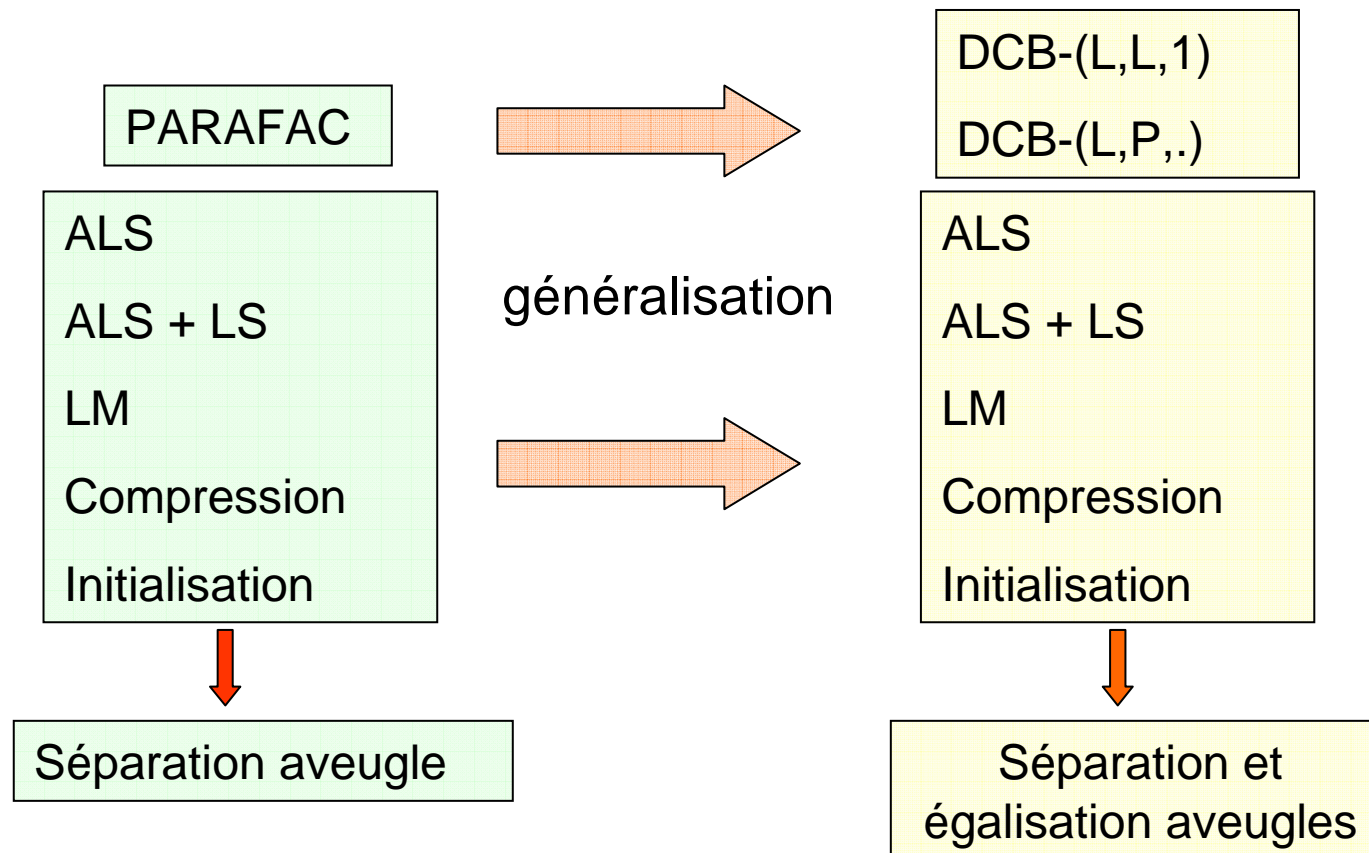
IV. Unicité des décompositions tensorielles

Conclusion et Perspectives

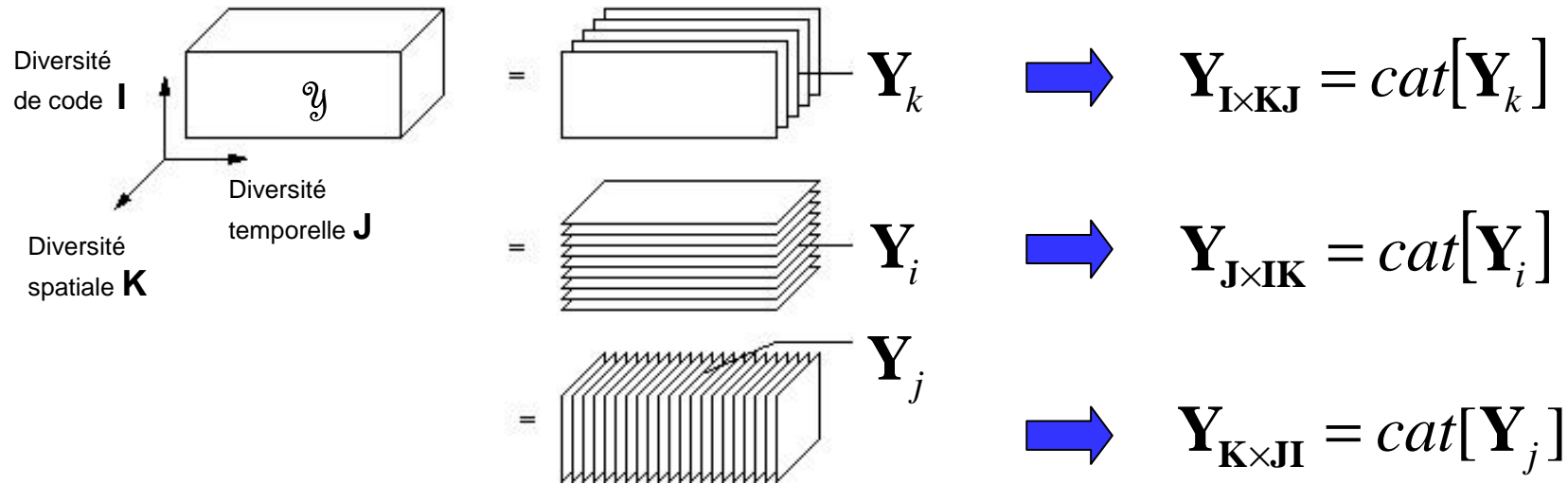
Algorithmes développés

- Décomposition de \mathcal{Y} \longleftrightarrow Estimation des composantes **A**, **S** et **H**
- Minimisation de la norme frobenius des résidus

$$\Phi = \left\| \mathbf{Y} - \text{Tens}(\hat{\mathbf{H}}, \hat{\mathbf{S}}, \hat{\mathbf{A}}) \right\|_F^2 \quad \text{Tens} = \text{PARAFAC ou DCB-(L,L,1) ou DCB-(L,P,.)}$$



Ecritures matricielles du tenseur des observations



Pour PARAFAC, DCB-(L,L,1) et DCB-(L,P,.), on peut écrire

$$\left\{ \begin{array}{l} \mathbf{Y}_{J \times IK} = \mathbf{S} \cdot \mathbf{Z}_1(\mathbf{A}, \mathbf{H}) \\ \mathbf{Y}_{I \times KJ} = \mathbf{H} \cdot \mathbf{Z}_2(\mathbf{S}, \mathbf{A}) \\ \mathbf{Y}_{K \times JI} = \mathbf{A} \cdot \mathbf{Z}_3(\mathbf{H}, \mathbf{S}) \end{array} \right. \Rightarrow \left\{ \begin{array}{l} \Phi = \left\| \mathbf{Y}_{J \times IK} - \hat{\mathbf{S}} \cdot \mathbf{Z}_1(\hat{\mathbf{A}}, \hat{\mathbf{H}}) \right\|_F^2 \\ \Phi = \left\| \mathbf{Y}_{I \times KJ} - \hat{\mathbf{H}} \cdot \mathbf{Z}_2(\hat{\mathbf{S}}, \hat{\mathbf{A}}) \right\|_F^2 \\ \Phi = \left\| \mathbf{Y}_{K \times JI} - \hat{\mathbf{A}} \cdot \mathbf{Z}_3(\hat{\mathbf{H}}, \hat{\mathbf{S}}) \right\|_F^2 \end{array} \right.$$

\mathbf{Z}_1 , \mathbf{Z}_2 et \mathbf{Z}_3 sont construites à partir de 2 matrices inconnues et leur structure dépend de la décomposition tensorielle considérée.

Algorithme 1: ALS « Alternating Least Squares »

- Principe: Mise à jour alternée des trois matrices $\mathbf{A}=[\mathbf{A}_1, \dots, \mathbf{A}_R]$, $\mathbf{S}=[\mathbf{S}_1, \dots, \mathbf{S}_R]$ et $\mathbf{H}=[\mathbf{H}_1, \dots, \mathbf{H}_R]$ par la résolution de 3 systèmes d'équations linéaires au sens des moindres carrés.
- Pour chaque système: minimisation de la même fonction de coût par rapport à une seule matrice.

Initialisation : $\hat{\mathbf{A}}^{(0)}, \hat{\mathbf{H}}^{(0)}, k = 1$

→ *while* $|\Phi^{(k-1)} - \Phi^{(k)}| > \varepsilon$ (e.g. $\varepsilon = 10^{-6}$)

$$\hat{\mathbf{S}}^{(k)} = \mathbf{Y}_{\mathbf{J} \times \mathbf{I} \times \mathbf{K}} \cdot \left[\mathbf{Z}_1(\hat{\mathbf{A}}^{(k-1)}, \hat{\mathbf{H}}^{(k-1)}) \right]^\dagger \quad (1)$$

$$\hat{\mathbf{H}}^{(k)} = \mathbf{Y}_{\mathbf{I} \times \mathbf{K} \times \mathbf{J}} \cdot \left[\mathbf{Z}_2(\hat{\mathbf{S}}^{(k)}, \hat{\mathbf{A}}^{(k-1)}) \right]^\dagger \quad (2)$$

$$\hat{\mathbf{A}}^{(k)} = \mathbf{Y}_{\mathbf{K} \times \mathbf{J} \times \mathbf{I}} \cdot \left[\mathbf{Z}_3(\hat{\mathbf{H}}^{(k)}, \hat{\mathbf{S}}^{(k)}) \right]^\dagger \quad (3)$$

$k \leftarrow k + 1$

Problème: Avec cette mise à jour, la structure Block-Toeplitz de \mathbf{S} est perdue à cause de l'indétermination → l'égalisation n'est pas réalisée

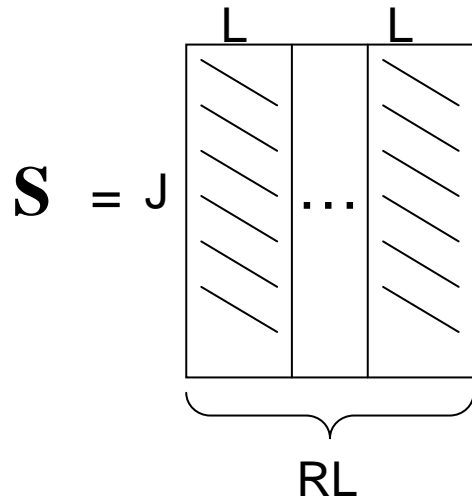
ALS: préservation de la structure Toeplitz

$$\mathbf{Y}_{J \times I \times K} = \mathbf{S} \cdot \mathbf{Z}_1(\mathbf{A}, \mathbf{H})$$

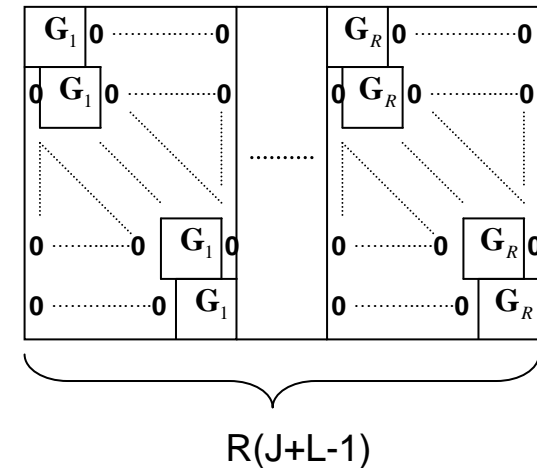


$$Y_{JIK} = \tilde{\mathbf{Z}}_1(\mathbf{A}, \mathbf{H}) \cdot \mathbf{s}$$

vectorisation



$$\tilde{\mathbf{Z}}_1(\mathbf{A}, \mathbf{H}) = \mathbf{JIK}$$



générée par

Equation (1) de l'ALS
remplacée par

$$\mathbf{s} = [\mathbf{s}_1 \mathbf{s}_2 \dots \mathbf{s}_R] \in \mathbb{C}^{R(J+L-1) \times 1}$$

MAJ de \mathbf{s}

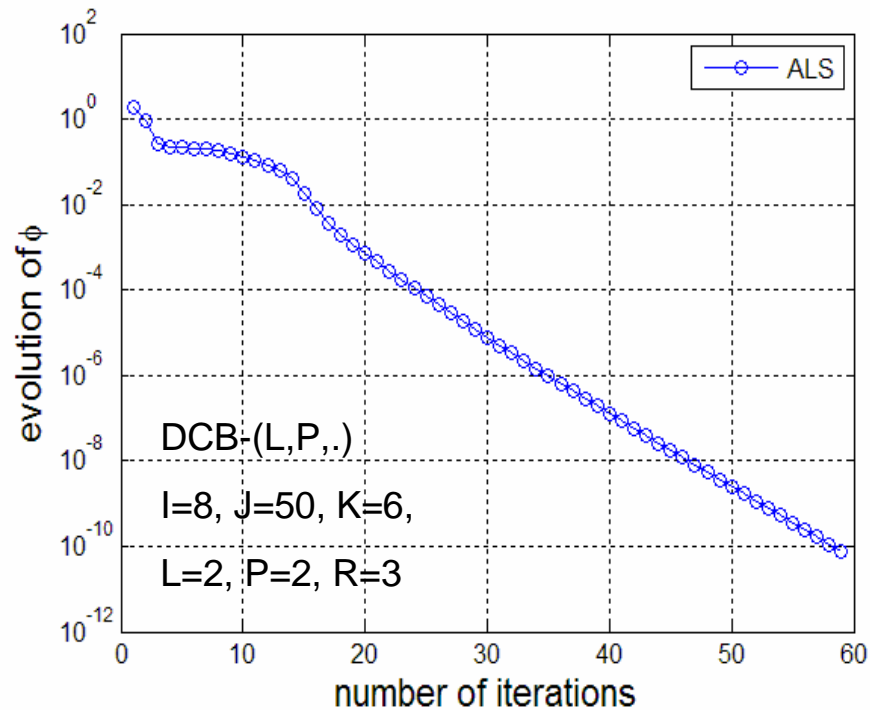
$$\hat{\mathbf{s}}^{(k)} = \left[\tilde{\mathbf{Z}}_1(\hat{\mathbf{A}}^{(k-1)}, \hat{\mathbf{H}}^{(k-1)}) \right]^\dagger \cdot Y_{JIK}$$

L'indétermination sur \mathbf{S} se résume alors à un facteur d'échelle complexe

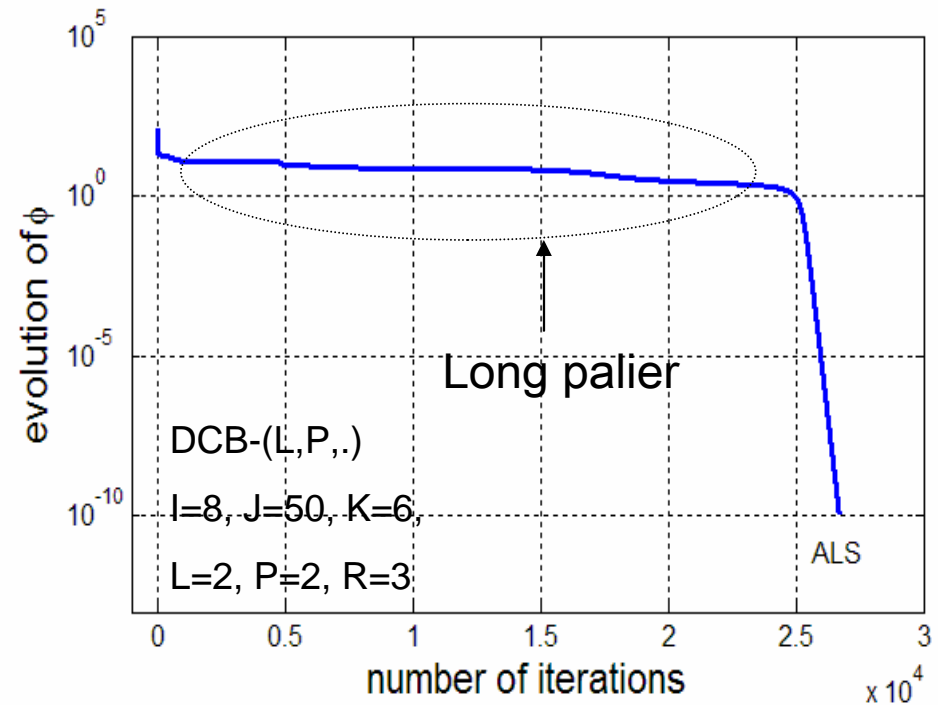
$$\mathbf{Y}_{J \times I \times K} = (\mu \mathbf{S}) \cdot (\mu^{-1} \mathbf{Z}_1(\mathbf{A}, \mathbf{H}))$$

Convergence de l'ALS

Problème «facile»



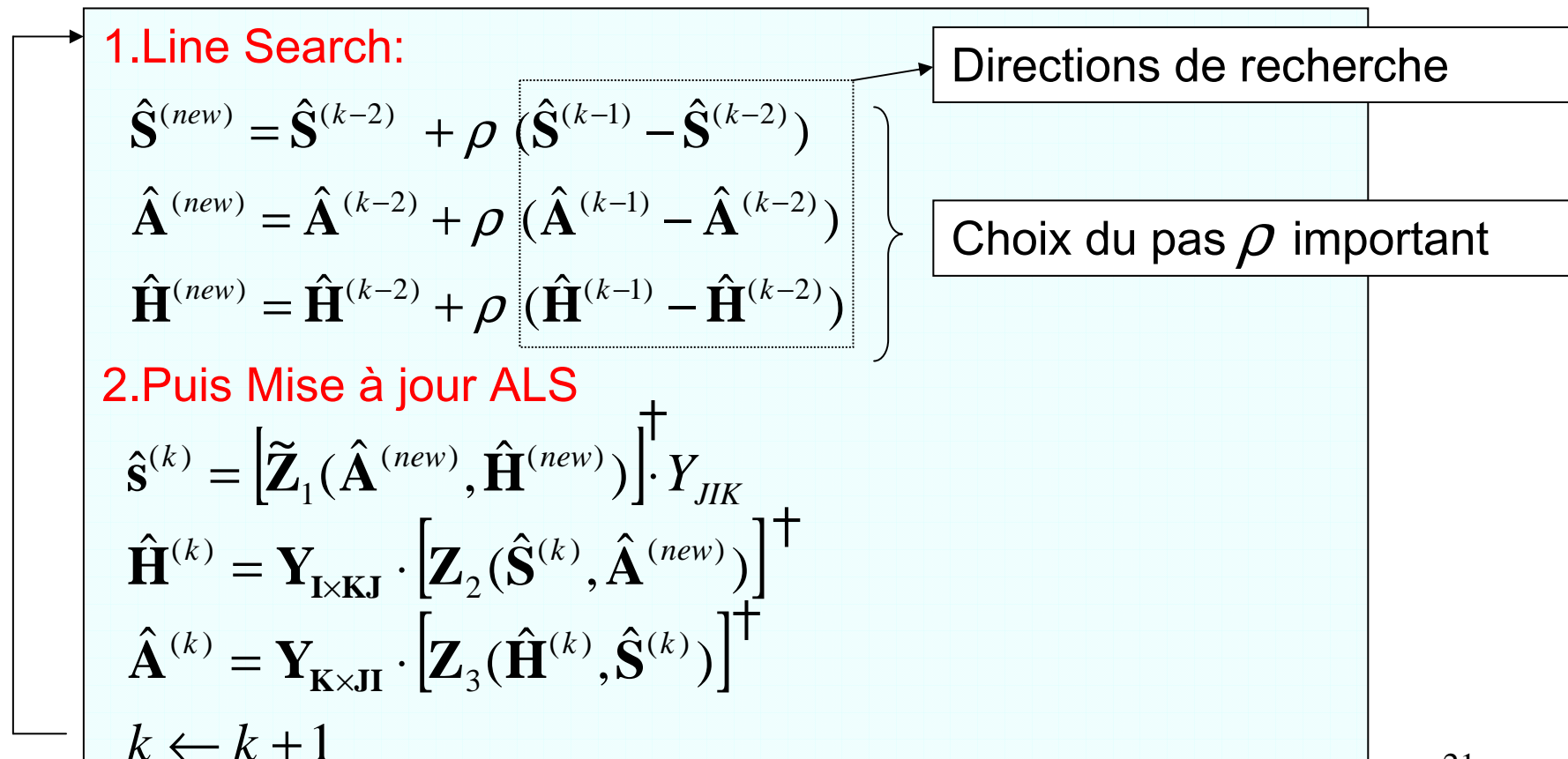
Problème «difficile»



- Convergence au mieux linéaire.
- Apparition de paliers → Grand nombre d'itérations nécessaire
- Nécessité de trouver des algorithmes plus performants pour réduire la longueur des paliers

Algorithme 2: Insertion d'une étape d'interpolation linéaire (LS = "Line Search") dans l'ALS

A chaque itération, interpolation linéaire des 3 composantes A, H et S à partir de leurs valeurs aux 2 itérations précédentes.



Choix du pas de l'interpolation linéaire

[Harshman, 1970] « LSH » Choisir $\rho = 1.25$

[Bro, 1997] « LSB » Choisir $\rho = k^{1/3}$

[Rajih, Comon, 2005] « Enhanced Line Search (ELS) »

Pour des tenseurs réels $\Phi(\mathbf{A}^{(new)}, \mathbf{S}^{(new)}, \mathbf{H}^{(new)}) = \Phi(\rho) = \text{polynôme de degré 6.}$

[Nion, De Lathauwer, SPAWC 2006]

« Enhanced Line Search with Complex Step (ELSCS) »

Pour des tenseurs complexes, recherche de $\rho = m.e^{i\theta}$ optimal.

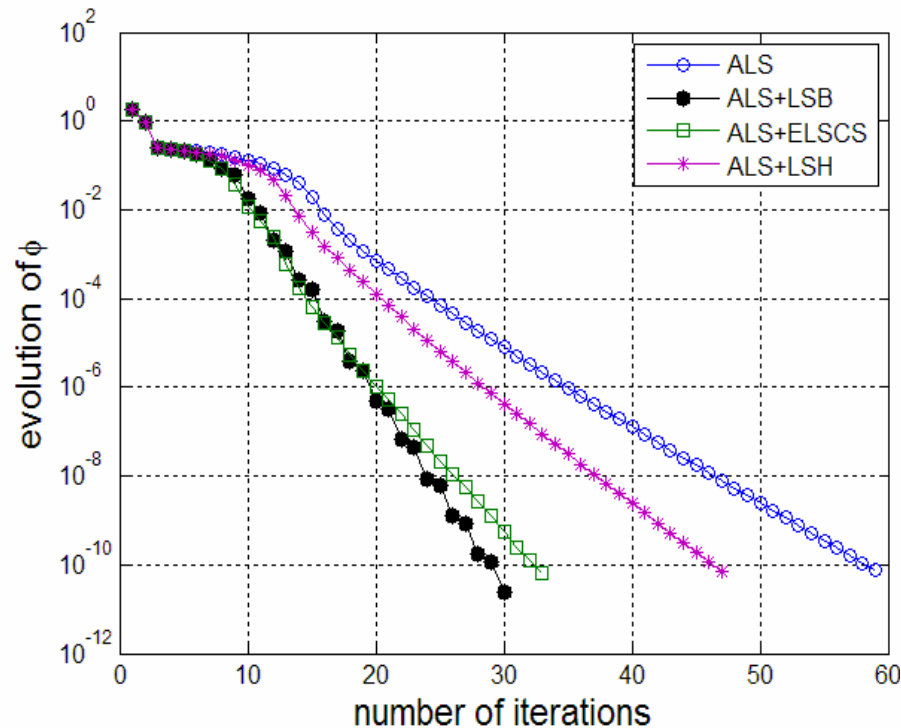
On a $\Phi(\mathbf{A}^{(new)}, \mathbf{S}^{(new)}, \mathbf{H}^{(new)}) = \Phi(m, \theta)$

MAJ alternée de m et de θ :

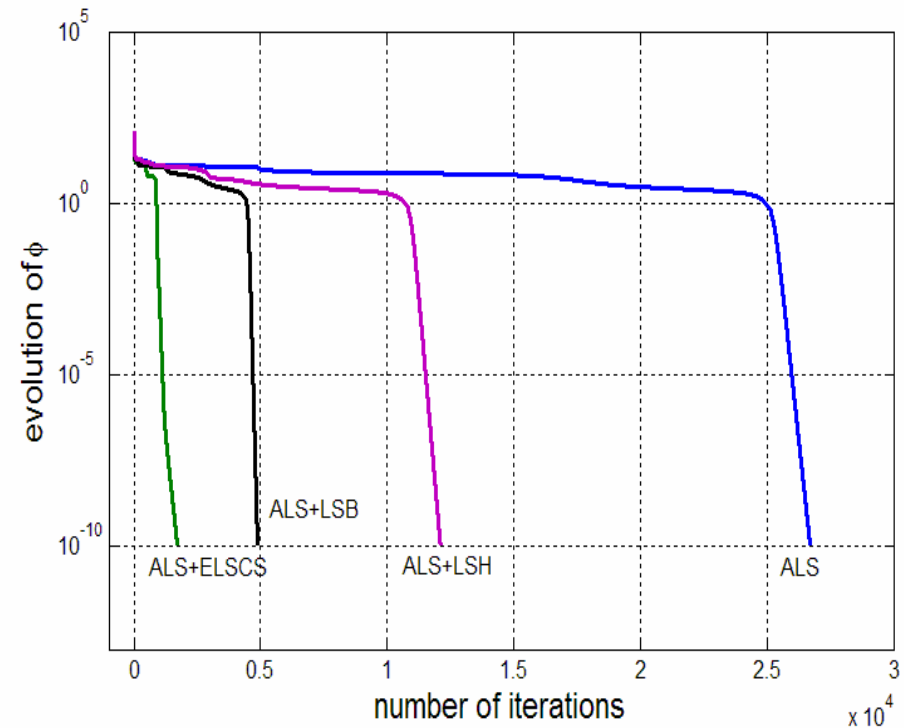
- MAJ de m : pour θ fixé, $\frac{\partial \Phi(m, \theta)}{\partial m} = \text{polynôme de degré 5 en } m$
- ← MAJ de θ : pour m fixé, $\frac{\partial \Phi(m, \theta)}{\partial \theta} = \text{polynôme de degré 6 en } t = \tan\left(\frac{\theta}{2}\right)$

Convergence de l'algorithme 2: ALS + interpolation linéaire

Problème «facile»



Problème «difficile»



- ELSCS: réduction considérable de la taille des paliers
- Coût de l'itération ELSCS faible (racines de polynômes) par rapport au coût de l'itération ALS

Algorithme 3: LM « Levenberg-Marquardt »

➤ Méthode d'optimisation alternative à l'ALS

➤ Concaténation de toutes les inconnues $\text{vec}(\mathbf{A})$, $\text{vec}(\mathbf{H})$ et \mathbf{s} dans un long vecteur \mathbf{p}

➤ Mise à jour de \mathbf{p} :
$$\mathbf{p}^{(k+1)} = \mathbf{p}^{(k)} + \Delta\mathbf{p}^{(k)} \quad (1)$$

➤ Gauss-Newton:
$$(\mathbf{J}^H \mathbf{J}) \Delta\mathbf{p}^{(k)} = -\mathbf{g} \quad (2)$$

➤ Levenberg-Marquardt:
$$(\mathbf{J}^H \mathbf{J} + \lambda \mathbf{I}) \Delta\mathbf{p}^{(k)} = -\mathbf{g} \quad (3)$$

➤ La matrice $(\mathbf{J}^H \mathbf{J} + \lambda \mathbf{I})$ est définie positive donc on peut résoudre (3) efficacement par une décomposition de Cholesky suivie de 2 éliminations de Gauss.

➤ Selon le conditionnement de $\mathbf{J}^H \mathbf{J}$, on ajuste la valeur de λ à chaque itération.

▪ Si λ grand $\Delta\mathbf{p}^{(k)} \approx -\frac{1}{\lambda} \mathbf{g}$ donc de type descente de gradient

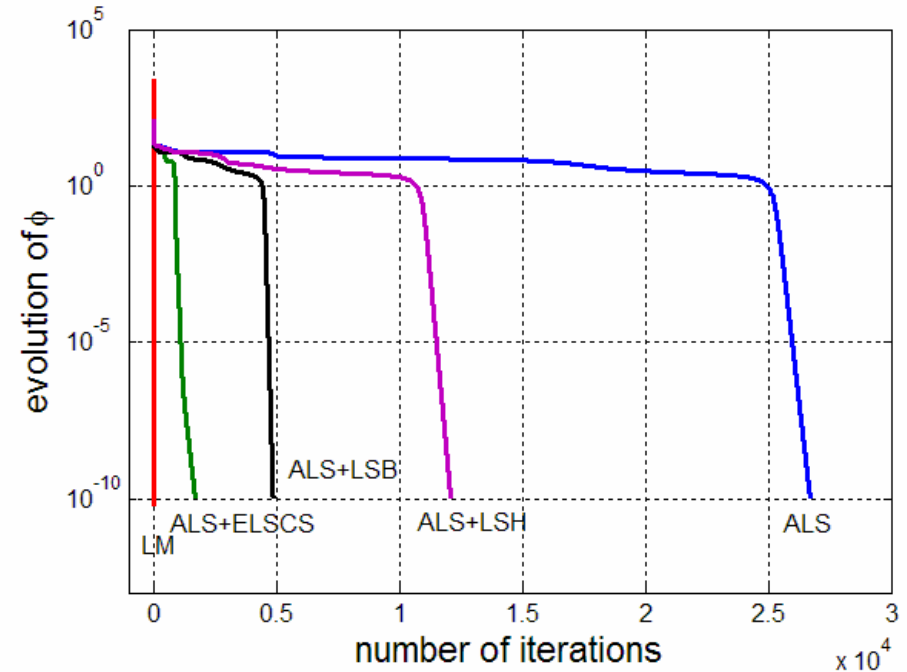
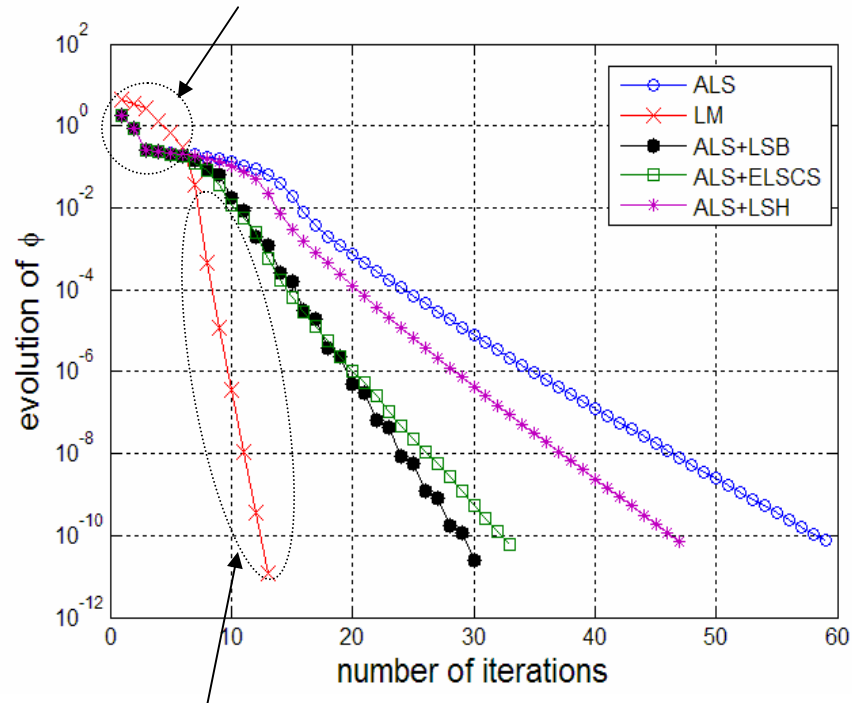
▪ Si λ petit $(\mathbf{J}^H \mathbf{J}) \Delta\mathbf{p}^{(k)} \approx -\mathbf{g}$ donc de type Gauss-Newton

Convergence des algorithmes ALS, ALS+ELSCS et LM

Problème «facile»

Problème «difficile»

Descente de Gradient



Gauss Newton

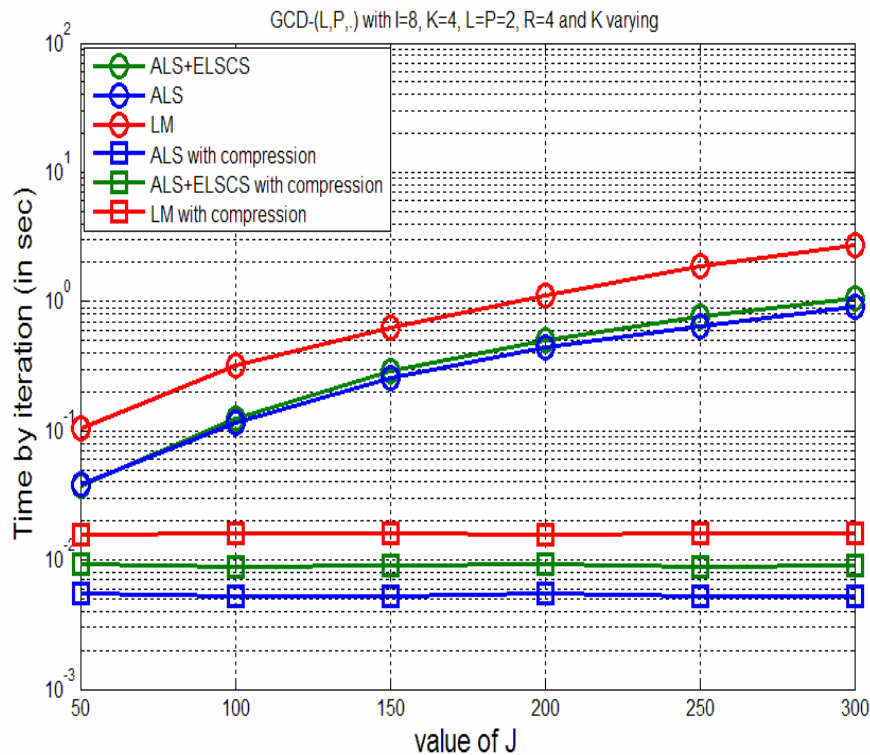
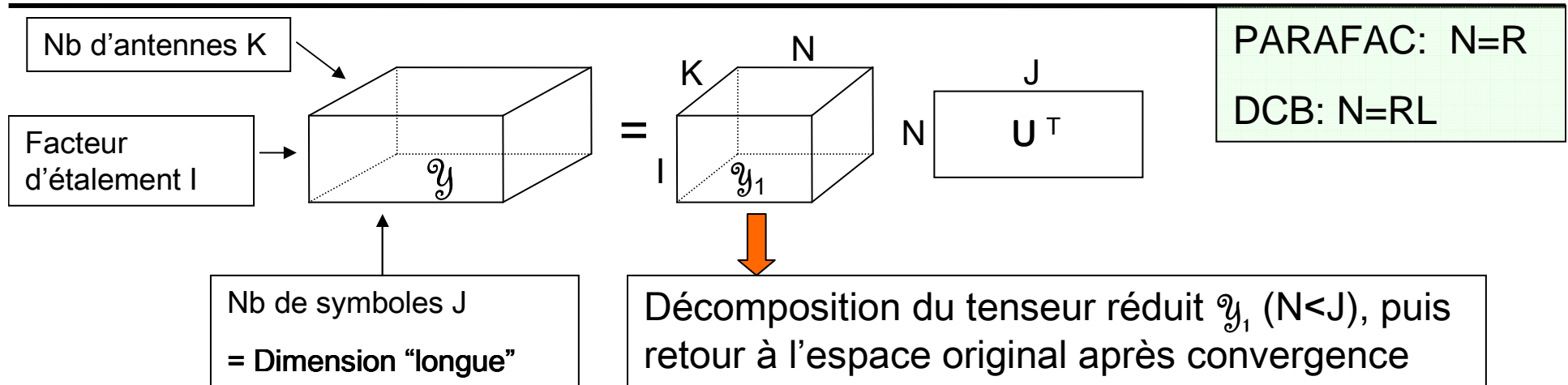
(convergence quadratique)

LM: nécessite beaucoup moins d'itérations qu'ALS

ELSCS: nécessite moins d'itérations qu'ALS

A quel prix?

Compression du tenseur des observations



Compression → réduction considérable du coût de calcul

Initialisation des algorithmes

- Problème: paliers plus ou moins longs selon l'initialisation.
- Première solution: choisir n initialisations aléatoires et garder la meilleure mais coût de calcul multiplié par n .
- Solution proposée: initialiser en exploitant le tenseur des observations.

PARAFAC et DCB-(L,L,1)

Les tranches \mathbf{Y}_k ($I \times J$) de \mathcal{Y} vérifient:

$$\begin{cases} \mathbf{Y}_1 = \mathbf{H} \cdot \Lambda_1 \cdot \mathbf{S}^T \\ \mathbf{Y}_2 = \mathbf{H} \cdot \Lambda_2 \cdot \mathbf{S}^T \\ \vdots \\ \mathbf{Y}_K = \mathbf{H} \cdot \Lambda_K \cdot \mathbf{S}^T \end{cases}$$

où les Λ_i sont diagonales

$$\mathbf{Y}_{k_1} \cdot (\mathbf{Y}_{k_2})^\dagger = \mathbf{H} \cdot (\Lambda_{k_1} \cdot \Lambda_{k_2}^{-1}) \cdot \mathbf{H}^\dagger$$



Estimation de $\hat{\mathbf{H}}^{(0)}$ puis de $\hat{\mathbf{S}}^{(0)}$ et $\hat{\mathbf{A}}^{(0)}$

DCB-(L,P,..)

Les tranches \mathbf{Y}_i ($K \times J$) de \mathcal{Y} vérifient:

$$\begin{cases} \mathbf{Y}_1 = \mathbf{A} \cdot \Lambda_1 \cdot \mathbf{S}^T \\ \mathbf{Y}_2 = \mathbf{A} \cdot \Lambda_2 \cdot \mathbf{S}^T \\ \vdots \\ \mathbf{Y}_I = \mathbf{A} \cdot \Lambda_I \cdot \mathbf{S}^T \end{cases}$$

où les $\Lambda_i \in \mathbb{C}^{RL \times RL}$ sont block-diagonales

$$\mathbf{Y}_{i_1} \cdot (\mathbf{Y}_{i_2})^\dagger = \mathbf{A} \cdot (\Lambda_{i_1} \cdot \Lambda_{i_2}^{-1}) \cdot \mathbf{A}^\dagger$$

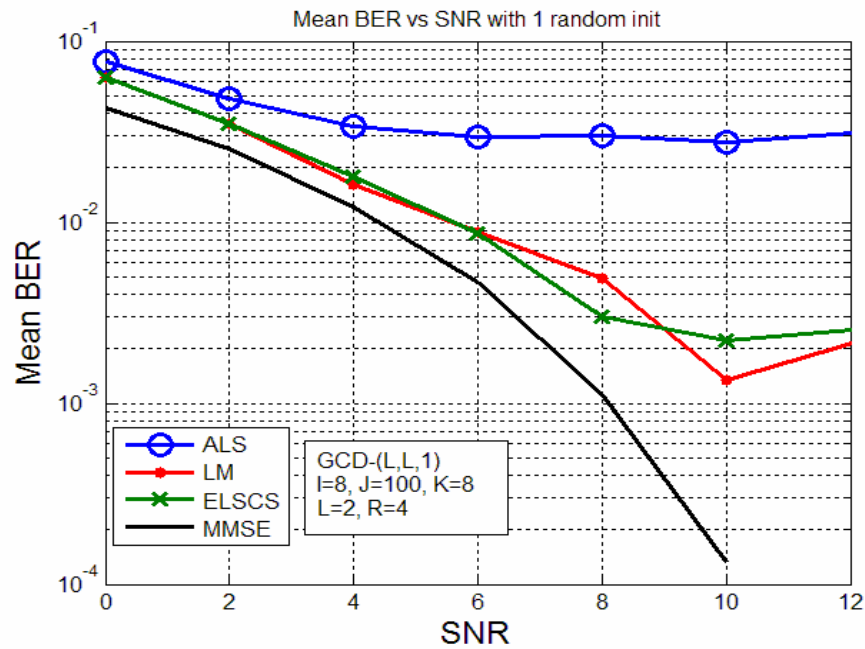


Estimation de $\hat{\mathbf{A}}^{(0)}$ puis de $\hat{\mathbf{S}}^{(0)}$ et $\hat{\mathbf{H}}^{(0)}$

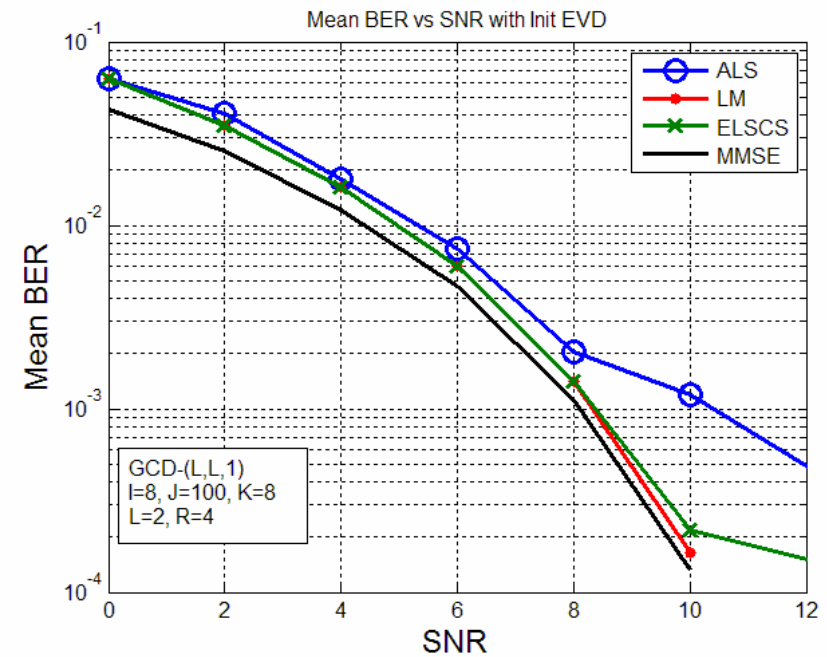
Impact de l'initialisation sur les performances

Simulations avec GCD-(L,L,1), $I=8$, $J=100$, $K=8$, $L=2$, $R=4$

Une seule init. aléatoire

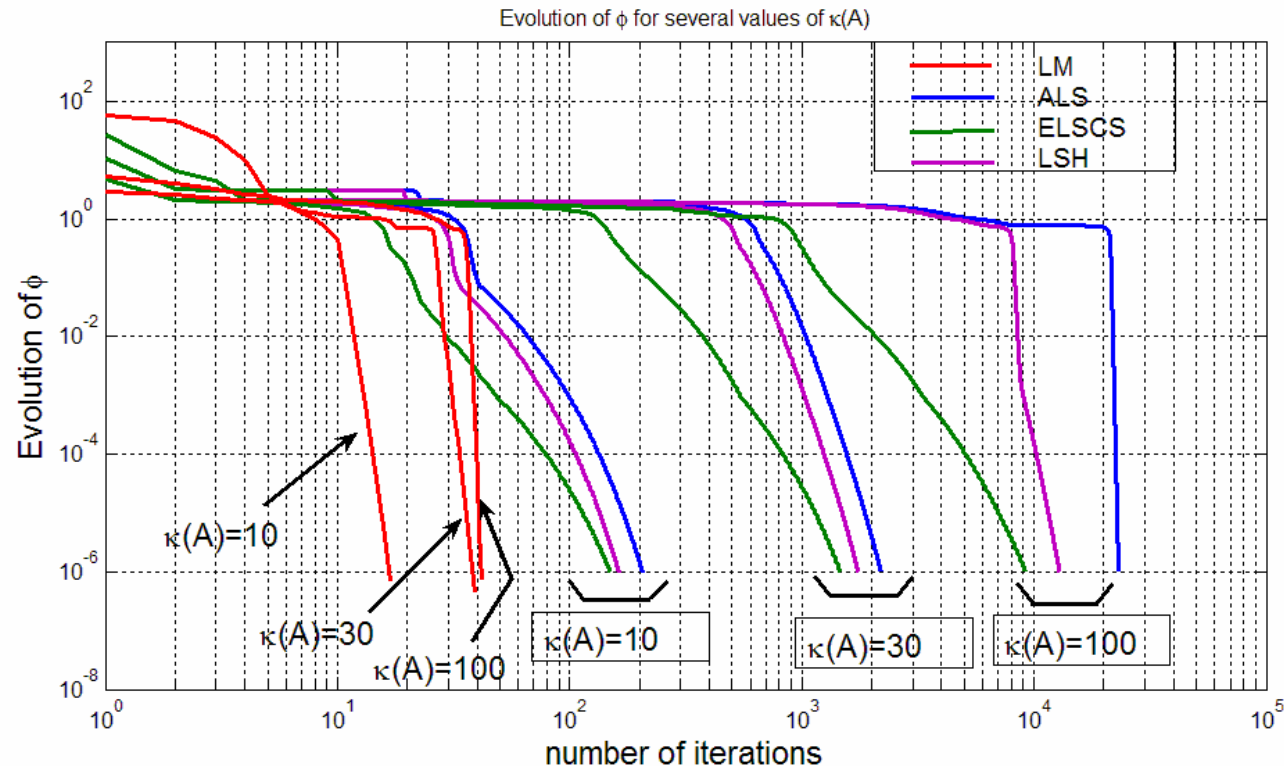


Init par technique proposée



Impact du conditionnement

Données non-bruitées, on fait varier le conditionnement de \mathbf{A}



ALS: paliers de plus en plus longs quand $\kappa(\mathbf{A})$ augmente

ELSCS: l'interpolation linéaire permet de réduire ces paliers

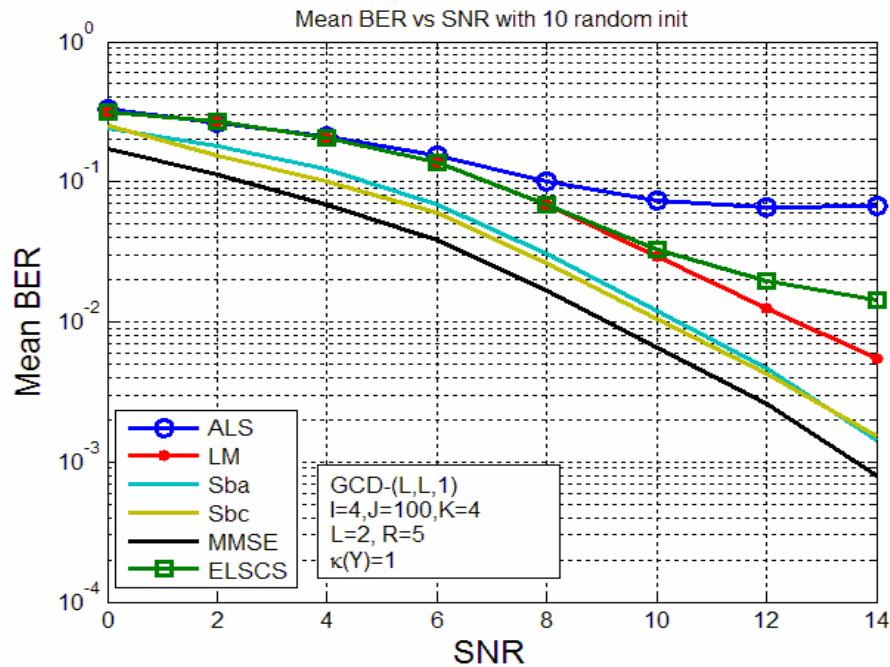
LM: très peu sensible au conditionnement de \mathbf{A}

Impact de l'effet Near-Far

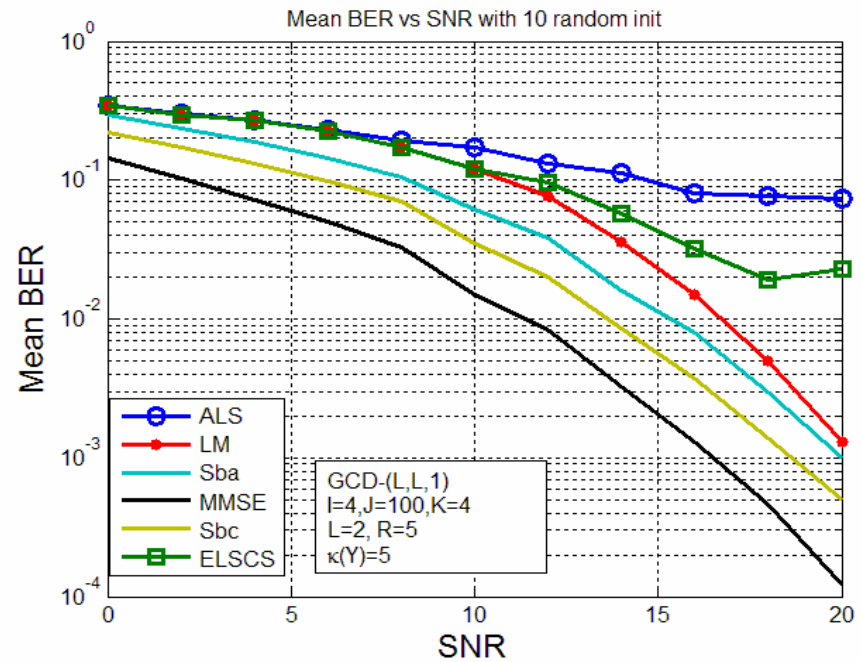
$$\mathcal{Y} = \sum_{r=1}^R \alpha_r \frac{\mathcal{Y}_r}{\|\mathcal{Y}_r\|_F} + \mathcal{B} \quad \kappa(\mathcal{Y}) = \frac{\max(\alpha_r)}{\min(\alpha_r)}$$

DCB-(L,L,1) avec l=4, J=100, K=4, L=2, R=5 et 10 initialisations différentes.

Problème difficile car: $R > l$ et $R > K$



$\kappa(\mathcal{Y}) = 1$



$\kappa(\mathcal{Y}) = 5$

Conclusion de la Partie II:

Algorithmes de calcul des décompositions tensorielles

Etat de l'art: Algorithmes existants pour PARAFAC (ALS, ALS+ELS, LM, ...)

Contributions et Résultats:

- ➔ - Généralisation de ces algorithmes au calcul des DCB.
- Préservation de la structure Toeplitz (égalisation).
- ELS généralisé au cas complexe: ELSCS.

- ➔ Compression: réduction du temps de calcul sans perte de performances.

- ➔ Initialisation: exploitation du tenseur des observations pour construire une initialisation performante.

- ➔ Performances:
 - LM et ELSCS convergent (beaucoup) plus vite qu'ALS.
 - LM beaucoup moins sensible au conditionnement et à l'effet Near-Far.

Introduction

- I. Les récepteurs aveugles tensoriels
- II. Algorithmes de calcul des décompositions tensorielles
- III. Unicité des décompositions tensorielles**
 1. PARAFAC
 2. DCB-(L,L,1)
 3. DCB-(L,P,.)

Conclusion et Perspectives

Borne d'unicité de la décomposition PARAFAC

Borne générique suffisante 1

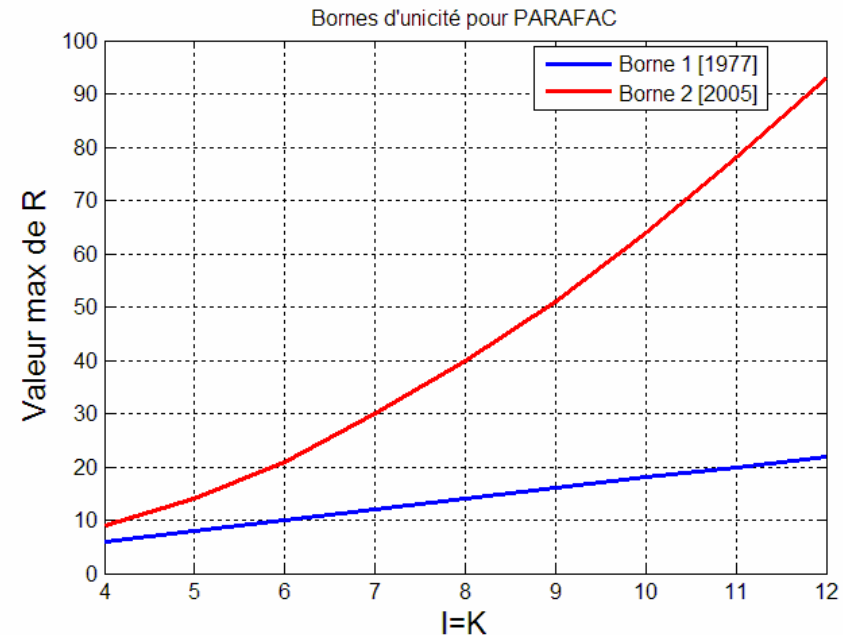
[Kruskal 1977]:

$$\min(I,R) + \min(J,R) + \min(K,R) \geq 2(R+1) \quad (1)$$

Borne générique suffisante 2

[Castaing & De Lathauwer 2005]:

$$J \geq R \text{ et } \frac{I(I-1)}{2} \frac{K(K-1)}{2} \geq \frac{R(R-1)}{2} \quad (2)$$



Questions: La borne (2), beaucoup moins contraignante, a été trouvée en reformulant la décomposition PARAFAC comme un problème de diagonalisation simultanée (SD) d'un ensemble de matrices.

→ Peut-on reformuler les DCB comme un problème de SD?

→ Si oui, en résulte-t-il une nouvelle borne d'unicité pour les DCB?

Réponses: Oui et Oui, nous l'avons montré pour la DCB-(L,L,1)

Bornes d'unicité de la DCB-(L,L,1)

Borne suffisante 1
[De Lathauwer 2006]

$$LR \leq IJ \text{ et } \min\left(\left\lfloor \frac{I}{L} \right\rfloor, R\right) + \min\left(\left\lfloor \frac{J}{L} \right\rfloor, R\right) + \min(K, R) \geq 2(R+1) \quad (1)$$

Borne suffisante 2
[Nion & De Lathauwer
SIMAX, 2007]:

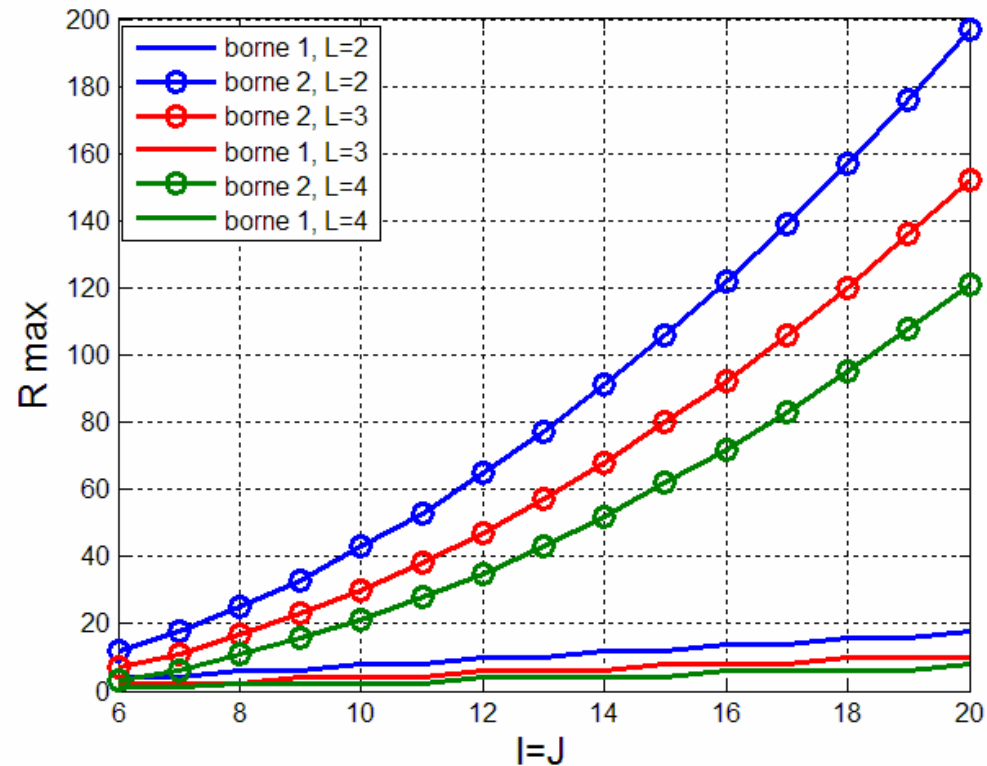
$$R \leq \min(IJ, K) \text{ et } C_I^{L+1} \cdot C_J^{L+1} \geq C_{R+L}^{L+1} - R \quad (2)$$

$$\text{où } C_n^k = \frac{n!}{k!(n-k)!}$$

Démarche pour trouver la borne (2):

- Reformulation de la DCB-(L,L,1) en un problème de SD.
- Cette reformulation est basée sur l'indépendance linéaire d'un ensemble de vecteurs.
- La borne (2) est en réalité la condition générique d'indépendance de ces vecteurs.
- Cette borne a été prouvée pour L=2.
- La démonstration est en cours pour L>2.
- L'hypothèse $R \leq \min(IJ, K)$ implique que la dimension « longue » est K

Bornes d'unicité de la DCB-(L,L,1) (suite)



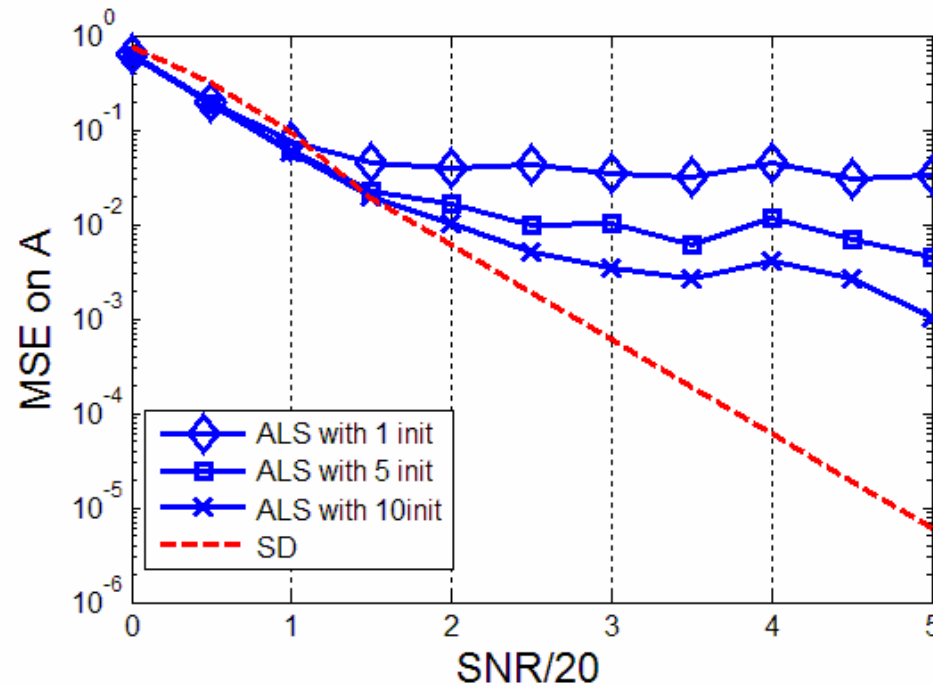
Nouvelle borne beaucoup moins contraignante que la précédente car on peut extraire beaucoup plus de contributions.

Performance de l'algorithme de SD

$I=6, J=6, K=8, L=2, R=7$
satisfaite



borne (1) non satisfaite, borne (2)



- Même après 10 initialisations aléatoires, l'algorithme ALS n'atteint pas les performances de l'algorithme SD.
- La diagonalisation simultanée en elle-même est très rapide.

- **Mais** les manipulations algébriques nécessaires pour arriver au système de matrices à diagonaliser sont excessivement coûteuses dès que $L > 2$.
- **De plus**, la structure Toeplitz est perdue.

Conclusion de la Partie III:

Unicité des décompositions tensorielles

- Pour chaque décomposition:
condition d'unicité = borne sur le nombre d'utilisateurs admissibles.
- PARAFAC: reformulation de la décomposition en terme d'un système de matrices à diagonaliser conjointement mène à une borne beaucoup moins contraignante que la borne de Kruskal.
- Ce résultat a été généralisé à la DCB-(L,L,1), dans le cas où la dimension « longue » est portée par K, i.e. par les termes de rang 1. [SIMAX, 2007]
- DCB-(L,P,.): une borne suffisante existe [De Lathauwer, 2006]
Perspective: assouplir cette borne.

Conclusion Générale

Modélisation:

- Récepteur PARAFAC 'classique': **ok si trajets directs** (séparation aveugle d'un mélange instantané)
 - Récepteur DCB: **trajets multiples + IES** (séparation + égalisation aveugles)
-

Approche:

- Déterministe, exploitation de la multilinearité du signal reçu, i.e. de la structure algébrique du tenseur des observations. 1 diversité = 1 dimension du tenseur.
-

Algorithmes:

- ALS sensible au conditionnement, à l'initialisation et à l'effet near-far
→ ALS+ELSCS et surtout LM préférables.
 - Techniques de compression et d'initialisation pour améliorer les performances
-

Performances:

- Récepteurs aveugles DCB proches du MMSE non-aveugle.
-

Unicité:

- Une nouvelle borne d'unicité a été trouvée pour DCB-(L,L,1).

Perspectives

Aspects à étudier :

- ➔ Nouvelle borne d'unicité pour DCB-(L,L,1) dans le cas où la dimension « longue » est portée par I ou J, i.e. par l'une des deux matrices de rang-L.
- ➔ Nouvelle borne d'unicité pour la DCB-(L,P,..).
- ➔ Exploitation de la connaissance des codes CDMA (applications civiles) dans les DCB.

Post-Doctorat : (avec Pr. Sidiropoulos, Grèce, financement DGA)

- ➔ Méthodes de poursuites de sous-espaces vectoriels pour les décompositions tensorielles.

Long-Terme :

- ➔ Appliquer les décompositions tensorielles en composantes blocs à des problèmes multivariés pour lesquels la capacité de modélisation de la décompositions PARAFAC n'est pas suffisante.

Publications

Articles de revue:

- [1] De Lathauwer, Nion, « *Decompositions of a Higher-Order Tensor in Block Terms- Part III: Alternating Least Squares Algorithms* », **SIMAX**, 2006, accepted.
- [2] Nion, De Lathauwer, « *An Enhanced Line Search Scheme for Complex-Valued Tensor Decompositions. Application in DS-CDMA* », Signal Processing, **ELSEVIER**, 2006, accepted.
- [3] Nion, De Lathauwer, « *A Block Component Model Based Blind DS-CDMA Receiver* », **IEEE Transactions on Signal Processing**, 2007, second stage of revision.
- [4] Nion, De Lathauwer, « *A Simultaneous Matrix Diagonalization Technique for the Decomposition of a Third-Order Tensor in rank-(L,L,1) Terms* », **SIMAX**, 2007, to be submitted.

Conférences internationales:

- [1] Nion, De Lathauwer, « *Generalized PARAFAC decompositions for blind multi-user access in wireless communications* », **WTDA**, Marseille, France, 2005.
- [2] Nion, De Lathauwer, « *A Block Factor Analysis Based Receiver for Blind Multi-User Access in Wireless Communications* », **ICASSP**, **IEEE**, Toulouse, France, 2006.
- [3] Nion, De Lathauwer, « *Line Search Computation of the Block Factor Model for Blind Multi-User Access in Wireless Communications* », **SPAWC**, **IEEE**, Cannes, France, 2006.
- [4] Nion, De Lathauwer, « *Levenberg-Marquardt Computation of the Block Factor Model for Blind Multi-User Access in Wireless Communications* », **EUSIPCO**, Florence, Italie, 2006.
- [5] Nion, De Lathauwer, « *A Tensor-Based Blind DS-CDMA Receiver using Simultaneous Matrix Diagonalization* », **SPAWC**, **IEEE**, Helsinki, Finlande, 2007.
- [6] Nion, De Lathauwer, « *Tensor-Based Blind CDMA Receivers* », **ASILOMAR**, **IEEE**, Pacific Grove, USA, 2007.


Conférences nationales:

- [1] Nion, De Lathauwer, « *Séparation et Egalisation aveugles de signaux CDMA par la décomposition en blocs d'un tenseur au moyen de l'algorithme de Levenberg-Marquardt* », **GRETSI**, Troyes, France, 2007.

Slides supplémentaires

Les tenseurs: un nombre croissant d'applications

- Vecteur = Tenseur d'ordre 1, Matrice = Tenseur d'ordre 2
- Tenseur d'ordre N = entité dont les éléments sont désignés par N indices

- De plus en plus d'applications impliquent la manipulation de tenseurs:
 - Phonétique
 - Psychométrie
 - Chimiométrie
 - Traitement d'images
 - Programmation scientifique
 - Séparation de sources et Analyse en Composantes Indépendantes
 - Télécommunications  Contexte de l'étude
 - etc...

Partie II: Algorithmes de calcul des décompositions tensorielles

Choix du pas de l'interpolation linéaire

[Harshman, 1970] « LSH » Choisir $\rho = 1.25$

[Bro, 1997] « LSB » Choisir $\rho = k^{1/3}$

Si $\Phi(\mathbf{A}^{(new)}, \mathbf{S}^{(new)}, \mathbf{H}^{(new)}) < \Phi(\mathbf{A}^{(k-1)}, \mathbf{S}^{(k-1)}, \mathbf{H}^{(k-1)})$ alors valider étape

Sinon $\mathbf{A}^{(new)} = \mathbf{A}^{(k-1)}, \mathbf{S}^{(new)} = \mathbf{S}^{(k-1)}, \mathbf{H}^{(new)} = \mathbf{H}^{(k-1)}$

[Rajih, Comon, 2005] « Enhanced Line Search (ELS) »

Pour des tenseurs réels $\Phi(\mathbf{A}^{(new)}, \mathbf{S}^{(new)}, \mathbf{H}^{(new)}) = \Phi(\rho) = \sum_{p=0}^6 \alpha_p \rho^p$

Le pas optimal $\rho \in \mathbb{R}$ est la racine d'un polynôme de degré 6.

[Nion, De Lathauwer, 2006] « Enhanced Line Search with Complex Step (ELSCS) »

Pour des tenseurs complexes, recherche de $\rho = m.e^{i\theta}$ optimal. On a $\Phi(\mathbf{A}^{(new)}, \mathbf{S}^{(new)}, \mathbf{H}^{(new)}) = \Phi(m, \theta)$

→ while $|\Phi^{(n)}(m, \theta) - \Phi^{(n-1)}(m, \theta)| > \varepsilon$

MAJ de m : racine de $\frac{\partial \Phi(m, \theta)}{\partial m} = \sum_{p=0}^5 \alpha_p m^p$ qui minimise $\Phi(m, \theta)$

MAJ de θ : racine de $\frac{\partial \Phi(m, \theta)}{\partial \theta} = \frac{1}{(1+t^2)^3} \sum_{p=0}^6 \beta_p t^p$, où $t = \tan(\frac{\theta}{2})$, qui minimise $\Phi(m, \theta)$

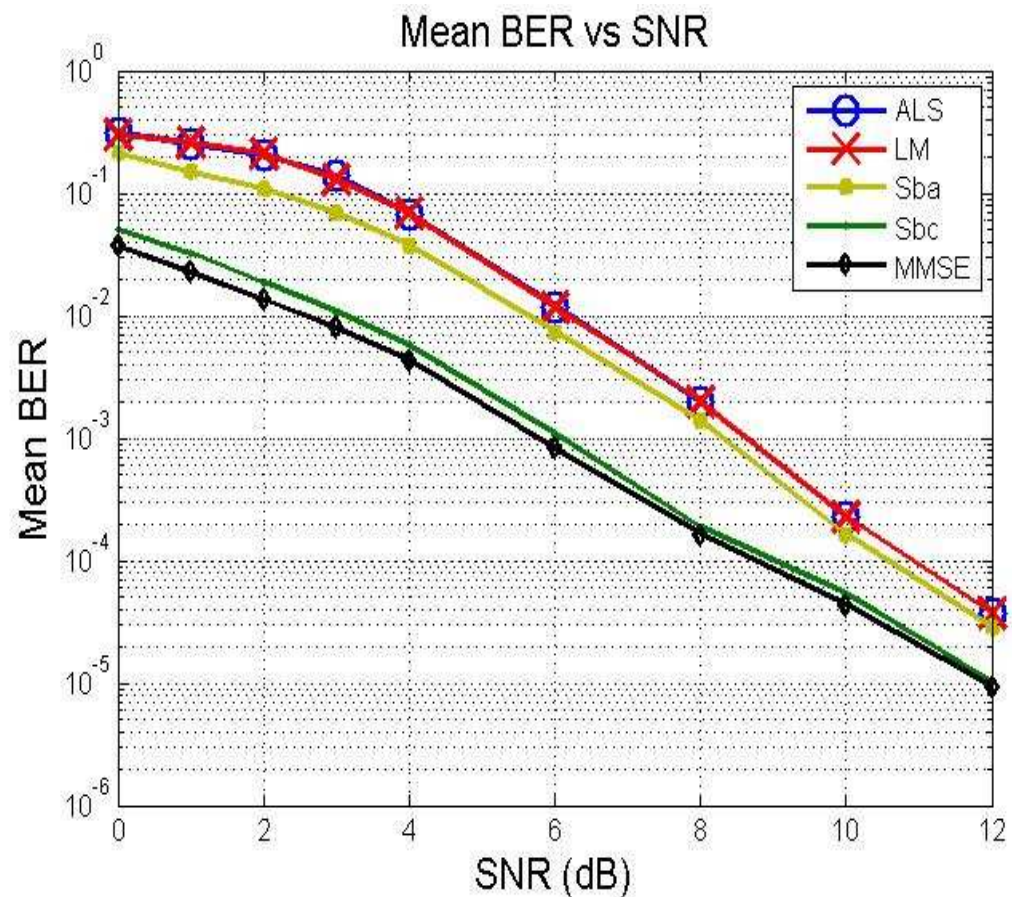
$n \leftarrow n + 1$

Méthode appelée ELSCS "Enhanced Line Search with Complex Step".

Test 1: pas d'effet near-far

➔ Tous les utilisateurs contribuent à puissance égale au signal global reçu

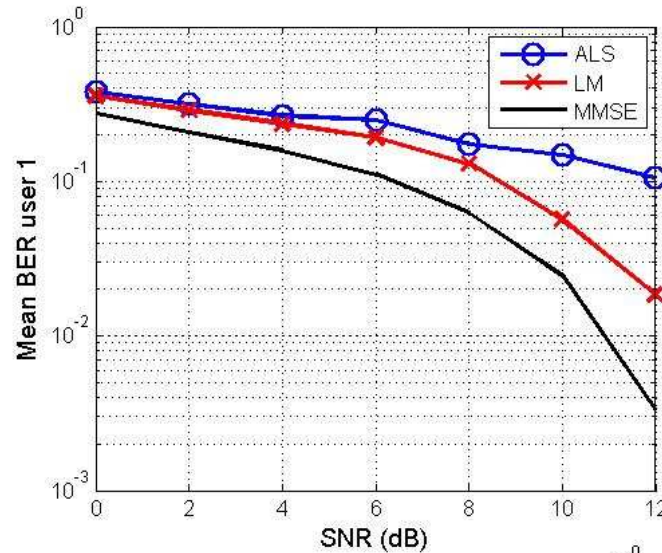
Paramètres: [I J K L P R]=[6 50 6 2 2 3], QPSK



Test 2: effet near-far

➔ Certains utilisateurs « éblouissent les autres »

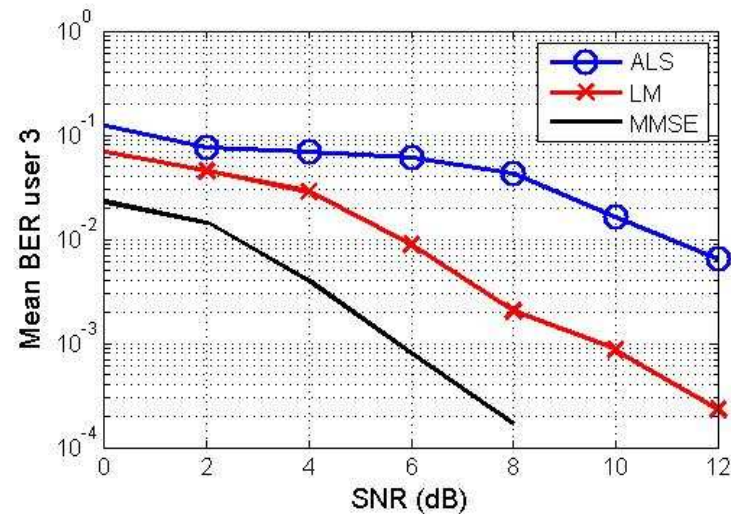
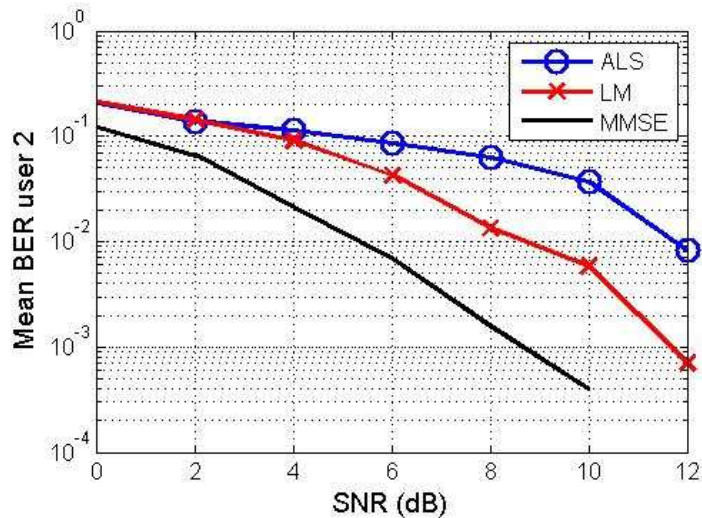
Paramètres: [I J K L P R]=[6 30 6 2 2 3], QPSK



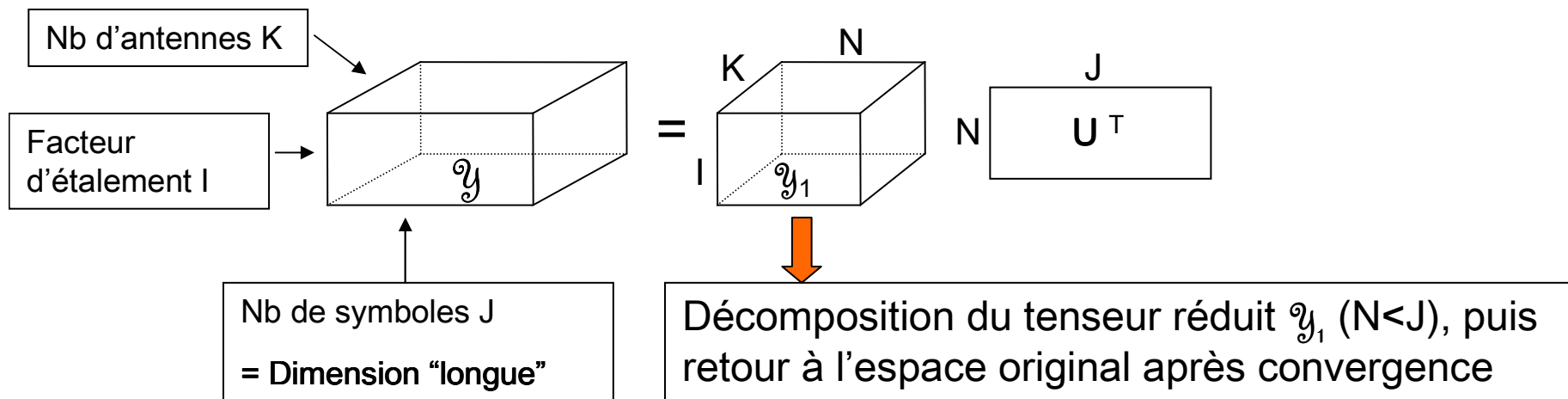
User 1:
 $P_1=1$

User 2:
 $P_2=5$

User 3:
 $P_3=10$



Compression du tenseur des observations



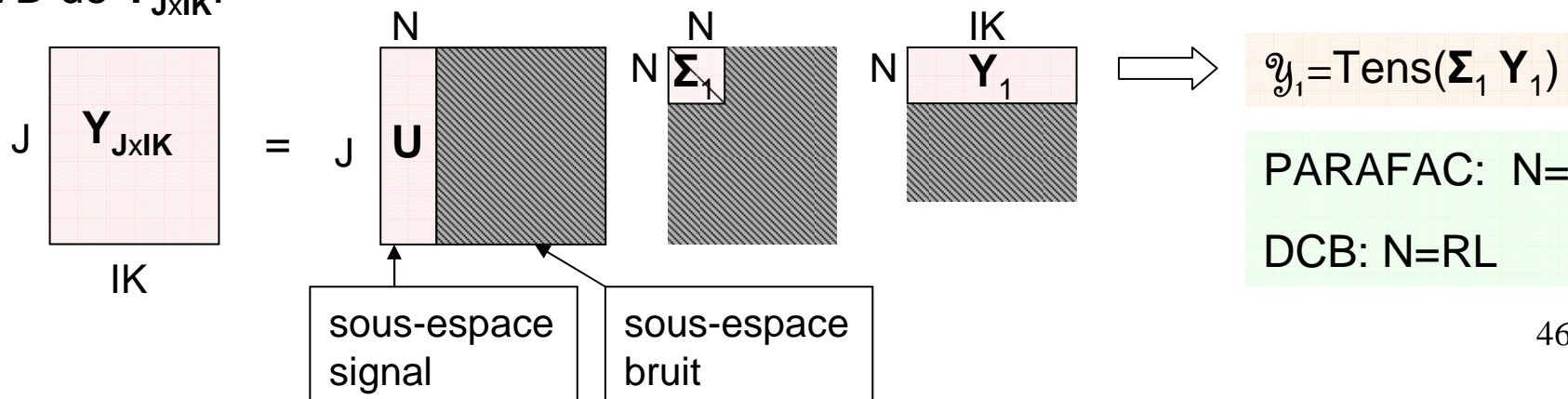
$$\mathbf{Y}_{J \times IK} = \mathbf{S} \cdot \mathbf{Z}_1$$

PARAFAC: $\mathbf{S} \in \mathbb{C}^{J \times R}$. Si $\min(J, IK) \geq R$ alors $\text{rank}(\mathbf{Y}_{J \times IK}) = R$.

DCB - (L, L, 1) et - (L, P, .): $\mathbf{S} \in \mathbb{C}^{J \times RL}$.

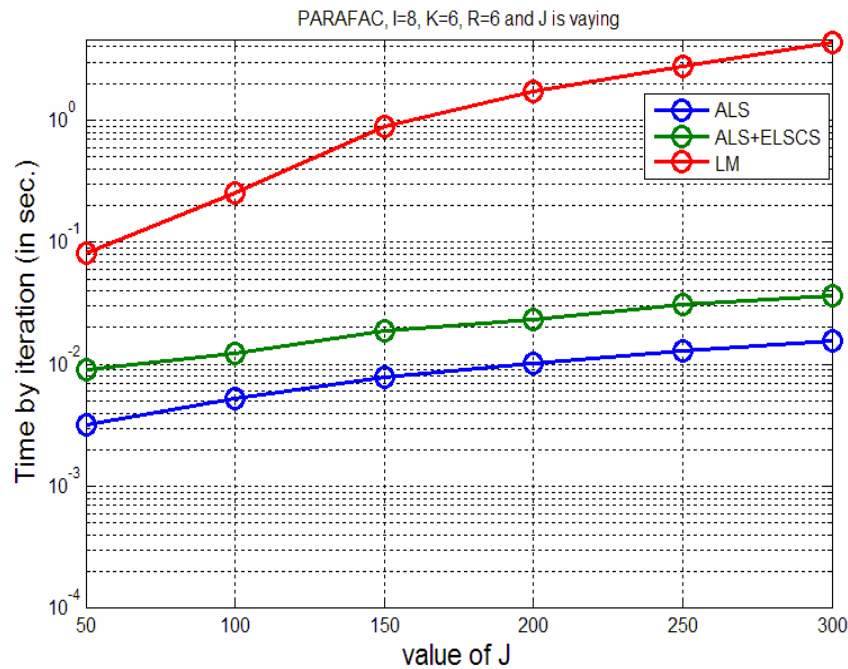
Si $\min(J, IK) \geq RL$ alors $\text{rank}(\mathbf{Y}_{J \times IK}) = RL$.

SVD de $\mathbf{Y}_{J \times IK}$:

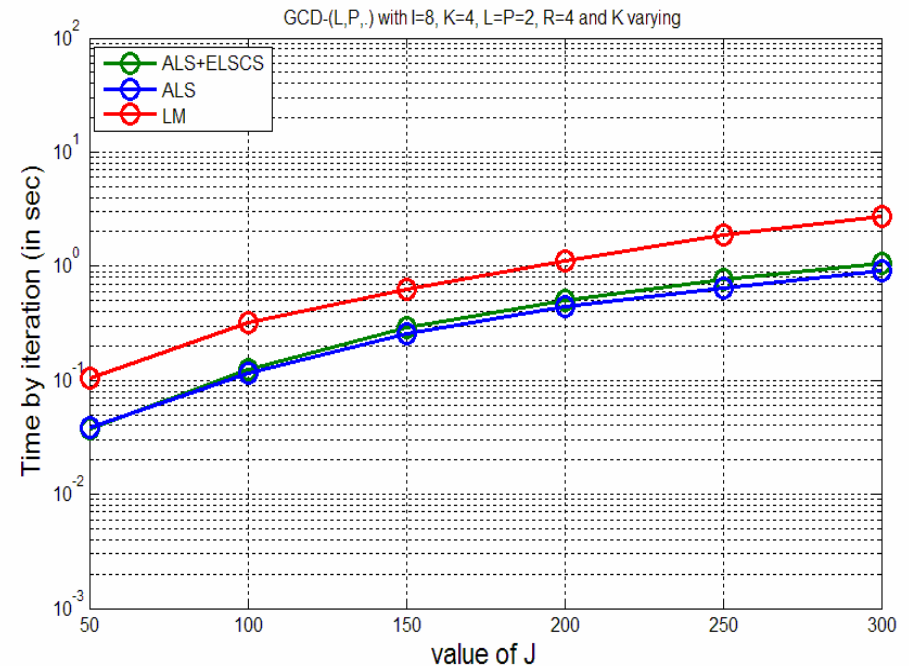


Coût de calcul des algorithmes ALS, ALS+ELSCS et LM

Evolution du temps de calcul par itération lorsque l'une des dimensions augmente (ici augmentation du nombre de symboles J).



PARAFAC

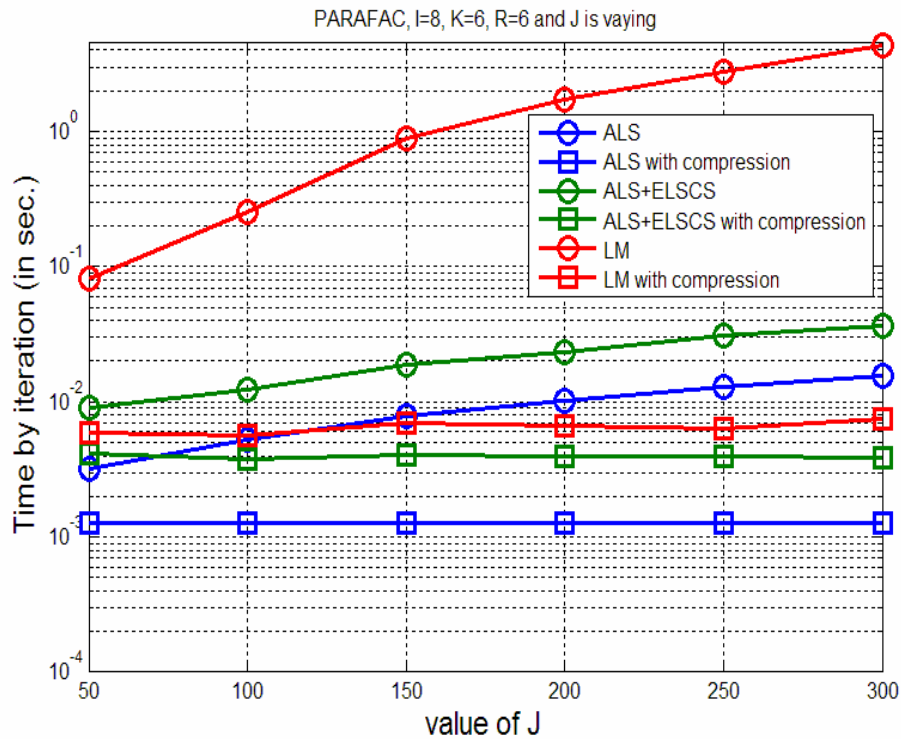


DCB-(L,P,.)

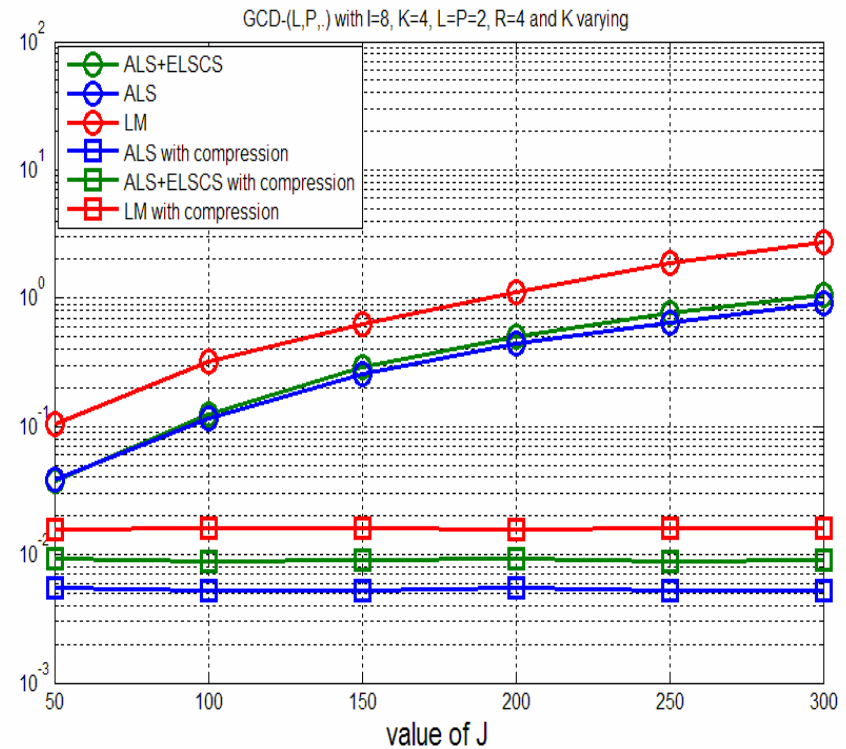
LM: plus coûteux qu'ALS
 ELSCS: à peine plus coûteux qu'ALS

Le coût devient prohibitif pour de grandes dimensions.

Coût de calcul après compression



PARAFAC

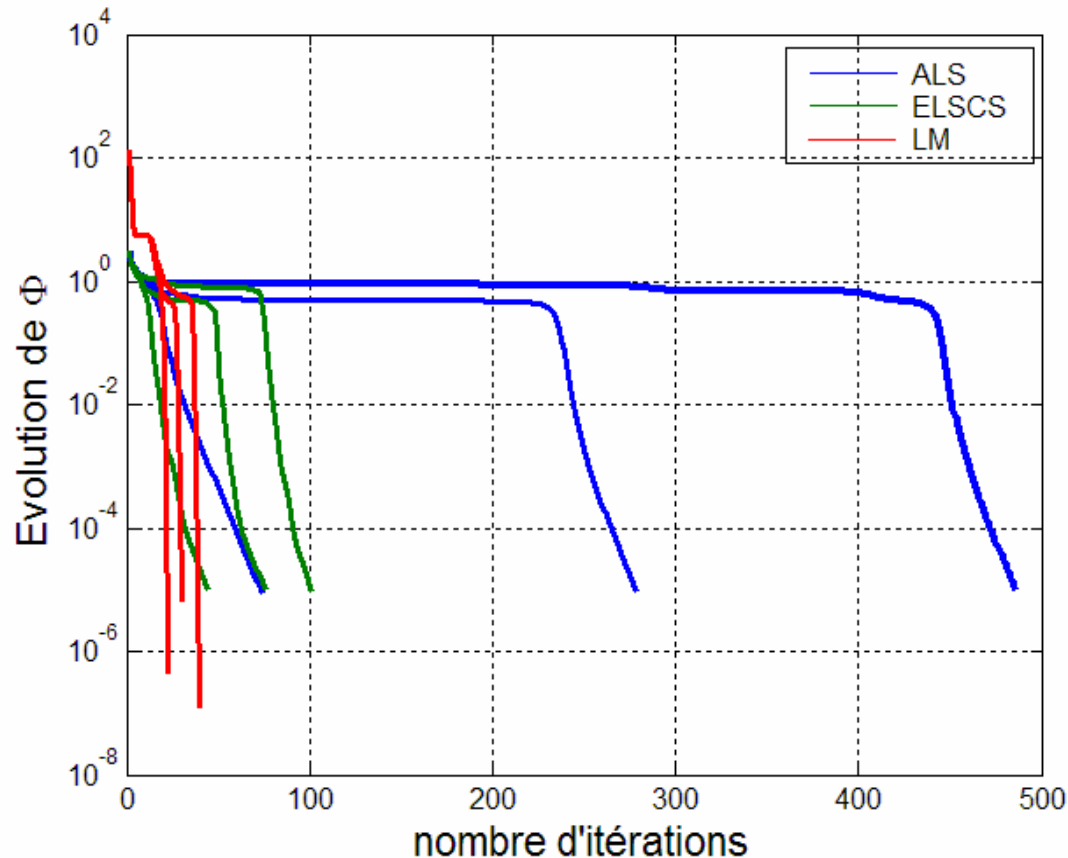


DCB-(L,P,.)

Compression → réduction considérable du coût de calcul

Pas de perte de performance car on tronque la SVD en préservant la dimension originale du sous-espace signal.

Importance de l'initialisation



- Modèle non bruité
- 3 initialisations aléatoires différentes.

- Techniques itératives: importance de l'initialisation $\hat{\mathbf{H}}^{(0)}, \hat{\mathbf{S}}^{(0)}, \hat{\mathbf{A}}^{(0)}$.
- Première solution: choisir n initialisations aléatoires et garder la meilleure.
Problème: coût de calcul multiplié par n .
- Solution proposée: initialiser en exploitant le tenseur des observations.

Initialisation des algorithmes

PARAFAC et DCB-(L,L,1)

Les tranches \mathbf{Y}_k ($I \times J$) de \mathcal{Y} vérifient

$$\begin{cases} \mathbf{Y}_1 = \mathbf{H} \cdot \mathbf{D}_1 \cdot \mathbf{S}^T \\ \mathbf{Y}_2 = \mathbf{H} \cdot \mathbf{D}_2 \cdot \mathbf{S}^T \\ \vdots \\ \mathbf{Y}_K = \mathbf{H} \cdot \mathbf{D}_K \cdot \mathbf{S}^T \end{cases}$$

	PARAFAC	DCB-(L,L,1)
\mathbf{H}	$(I \times R)$	$(I \times RL)$
\mathbf{S}	$(J \times R)$	$(J \times RL)$
$\mathbf{D}_k =$	$diag([\mathbf{A}]_{k,:})$	$diag([\mathbf{A}]_{k,:}) \otimes \mathbf{I}_L$
Hypothèse	$R \leq \min(I, J)$	$RL \leq \min(I, J)$

Génériquement, on a $\mathbf{Y}_{k_1} \cdot (\mathbf{Y}_{k_2})^\dagger = \mathbf{H} \cdot (\mathbf{D}_{k_1} \cdot \mathbf{D}_{k_2}^{-1}) \cdot \mathbf{H}^\dagger$

$\hat{\mathbf{H}}^{(0)}$ estimé comme les R (ou RL) premiers vecteurs propres de $\mathbf{Y}_{k_1} \cdot (\mathbf{Y}_{k_2})^\dagger$

$\hat{\mathbf{S}}^{(0)} = \mathbf{Y}_k^T (\hat{\mathbf{H}}^T)^\dagger$ puis on déduit $\hat{\mathbf{A}}^{(0)}$

Technique basée sur l'EVD pour l'initialisation (suite)

DCB-(L,P,..)

On suppose $L = P$ et $RL \leq \min(J, K)$

Contraignant en pratique: $L = \text{nb. d'IES}$, $P = \text{nb. de trajets}$, $K = \text{nb. d'antennes}$

Les tranches \mathbf{Y}_i ($K \times J$) de \mathcal{Y} vérifient

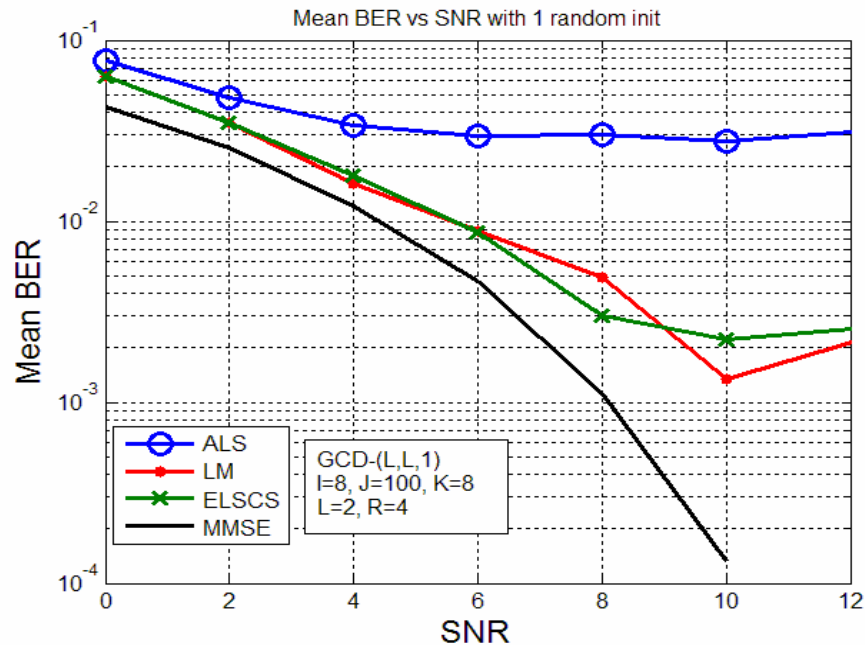
$$\begin{cases} \mathbf{Y}_1 = \mathbf{A} \cdot \Lambda_1 \cdot \mathbf{S}^T \\ \mathbf{Y}_2 = \mathbf{A} \cdot \Lambda_2 \cdot \mathbf{S}^T \\ \vdots \\ \mathbf{Y}_I = \mathbf{A} \cdot \Lambda_I \cdot \mathbf{S}^T \end{cases}$$

où $\mathbf{A} \in \mathbb{C}^{K \times RP}$, $\mathbf{S} \in \mathbb{C}^{J \times RL}$ et où les $\Lambda_i \in \mathbb{C}^{RL \times RL}$ sont block-diagonales

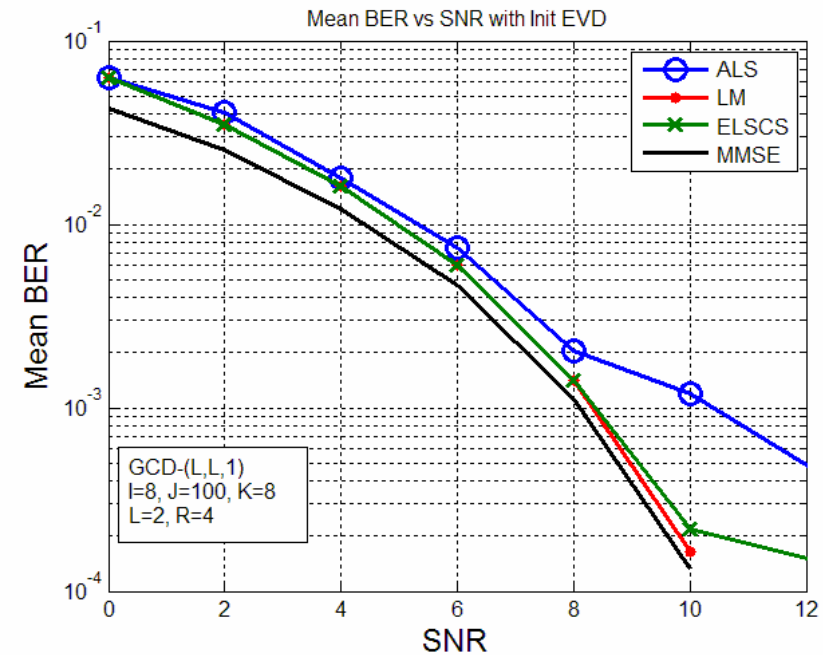
Comme pour PARAFAC et DCB-(L,P,..), on peut estimer $\hat{\mathbf{H}}^{(0)}$, $\hat{\mathbf{S}}^{(0)}$, $\hat{\mathbf{A}}^{(0)}$

Impact de l'initialisation sur les performances

Une seule init. aléatoire



Init par EVD

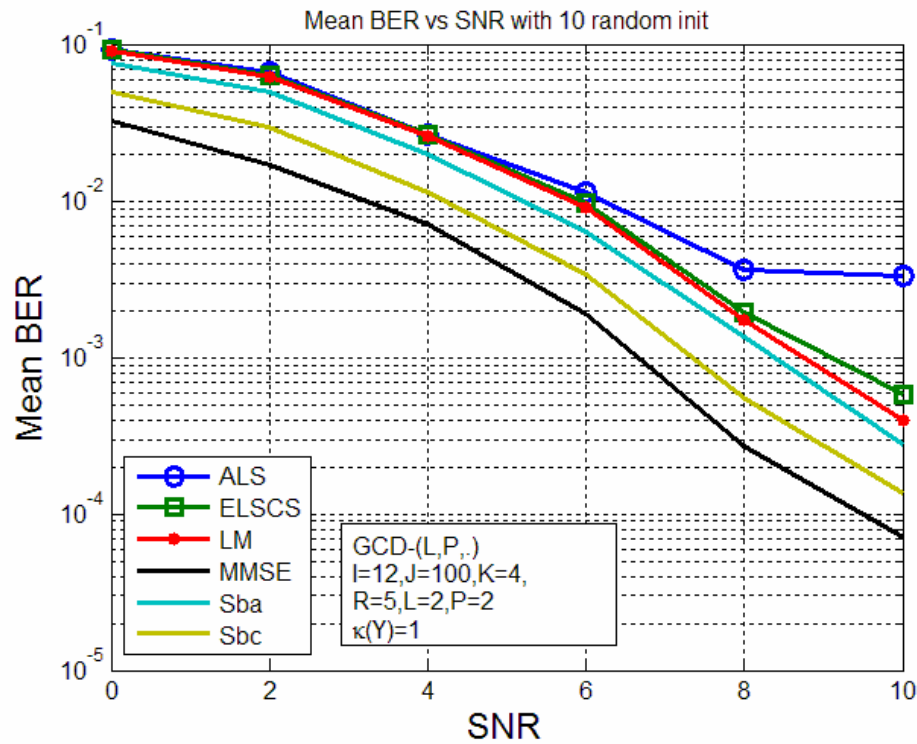


Conclusion Calcul par EVD

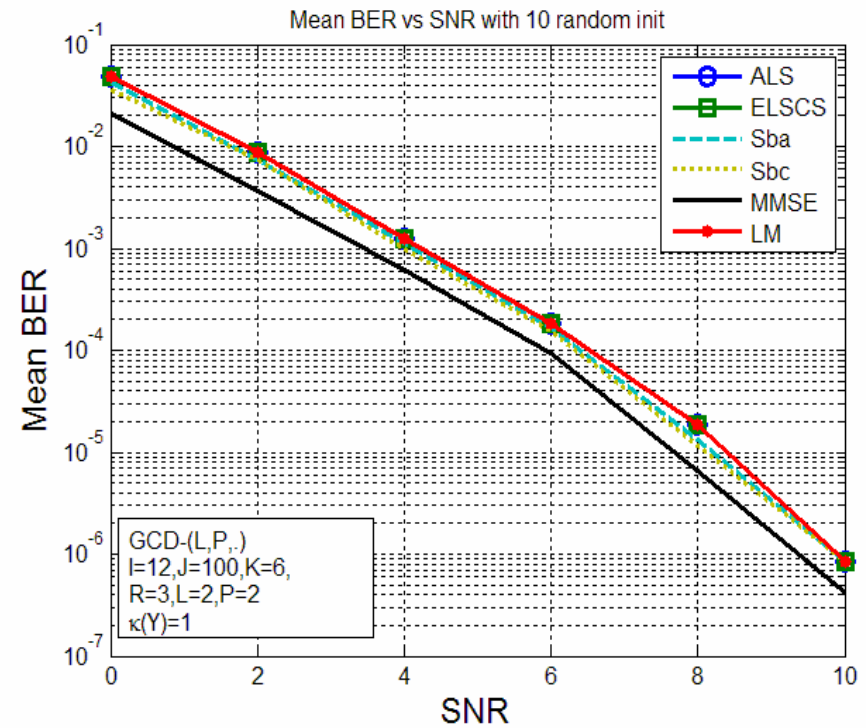
- Contraintes sur les dimensions
- Absence de bruit: cette technique permet de trouver les composantes $\mathbf{H}, \mathbf{S}, \mathbf{A}$ (aux indéterminations du modèle près) sans recourir à une méthode itérative.
- Présence de bruit: cette technique fournit une initialisation efficace aux algorithmes itératifs (ALS, ELSCS, LM, etc).

Impact du nombre d'antennes

DCB-(L,P,..) avec $l=12, J=100, L=2, P=2$ et 10 initialisations différentes.



K=4 antennes et R=5 utilisateurs



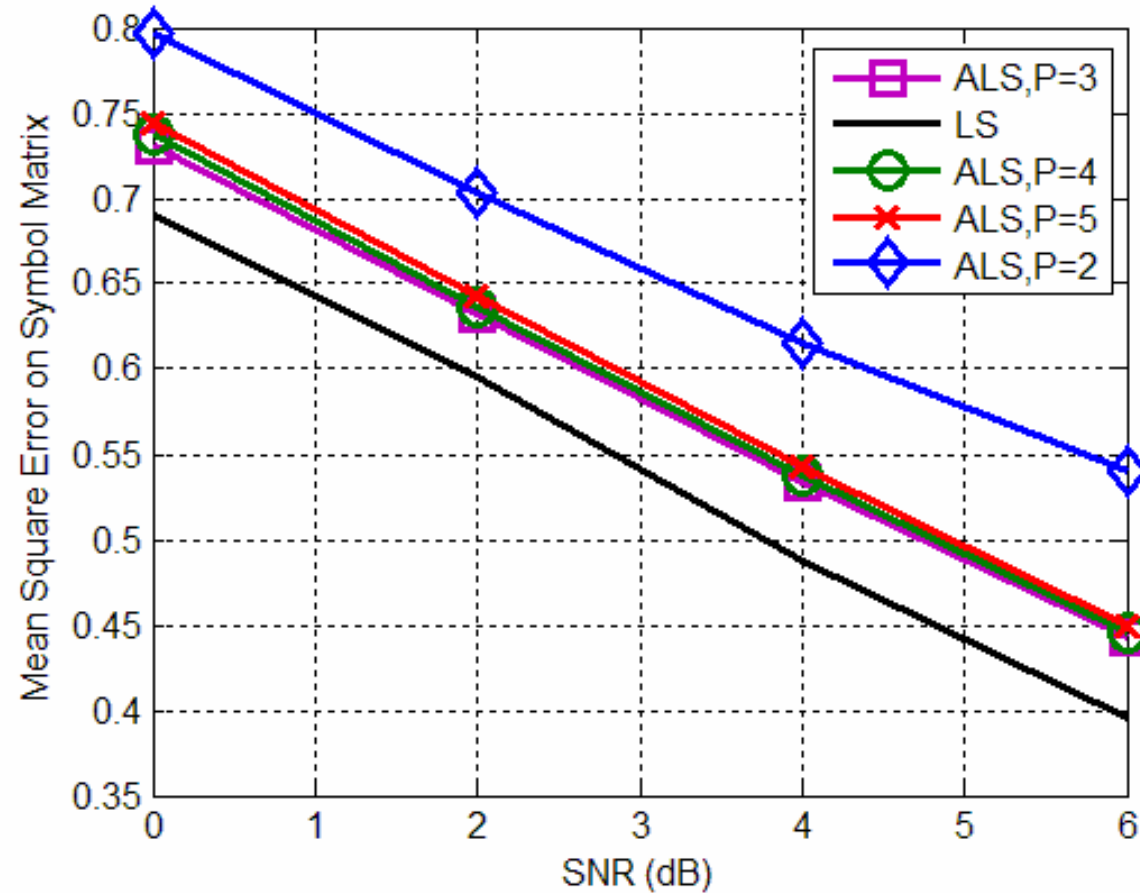
K=6 antennes et R=3 utilisateurs

Sur-estimation du nombre de trajets P

Tenseur des observations construit avec $P=3$ trajets par utilisateurs

Calcul de sa décomposition avec une sur-estimation ($P=4$ et $P=5$) ou une sous-estimation ($P=2$) de ce facteur.

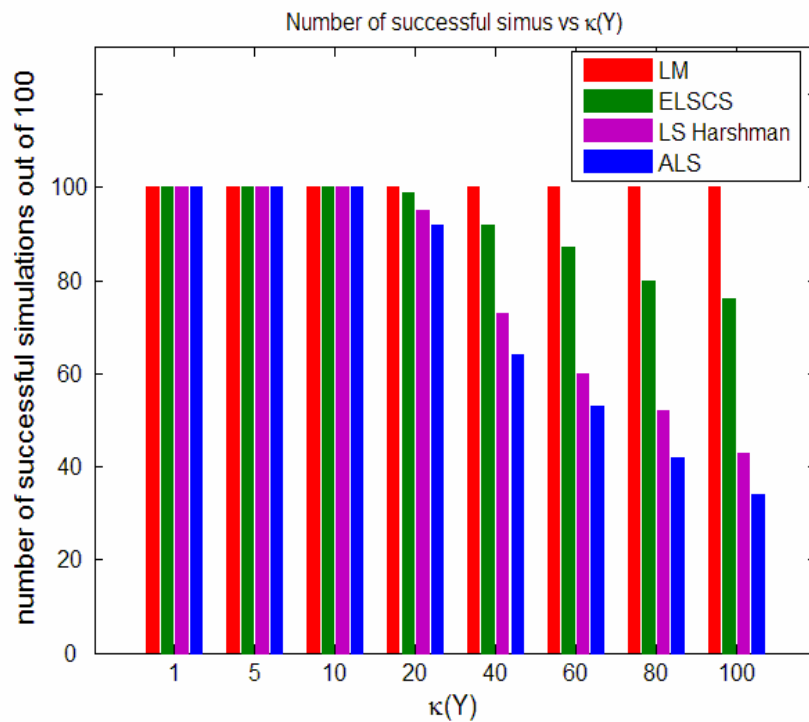
Erreur quadratique moyenne sur la matrice de symboles vs. SNR



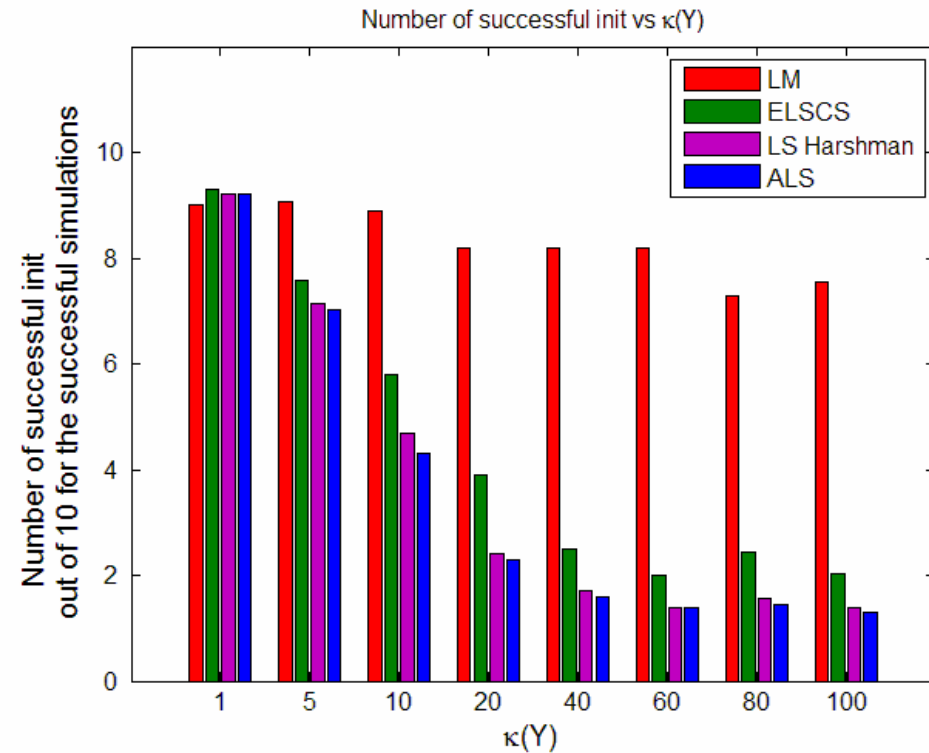
Effet Near-Far sur des données non-bruitées

Pourcentage de simulations réussies

(au moins une des 10 initialisations aléatoires converge vers le min. global)

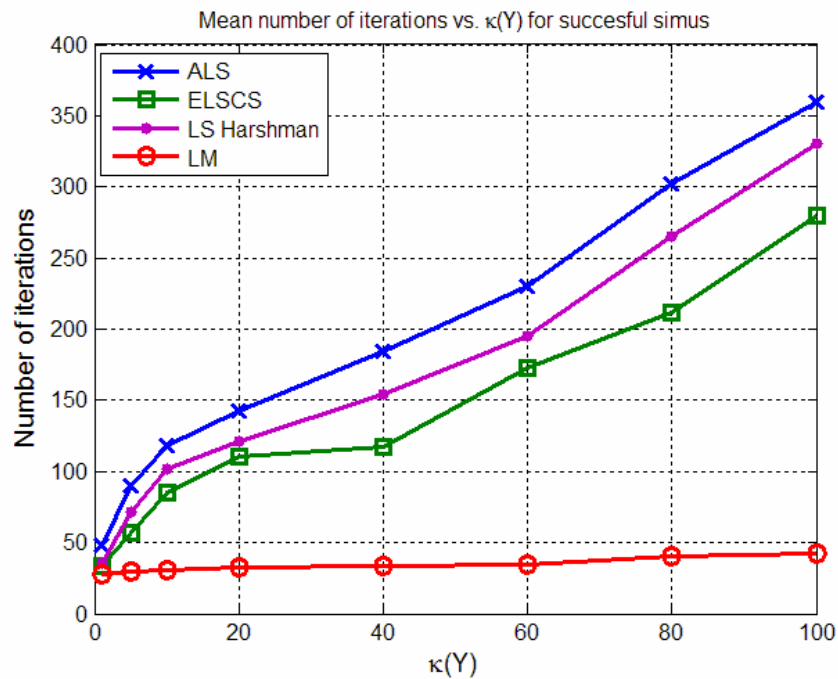


Parmi les simulations réussies, combien d'initialisations sur 10 ont mené au min. global?

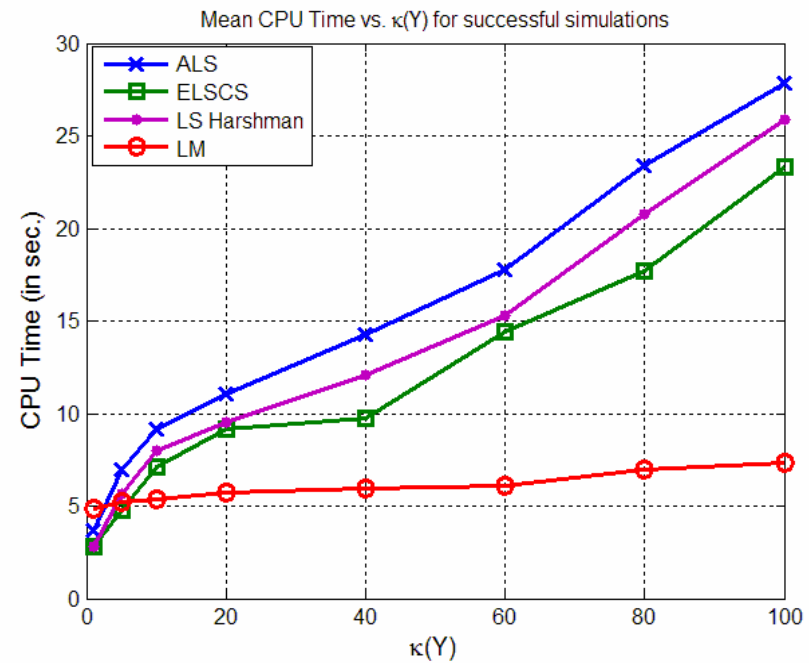


Effet Near-Far sur des données non-bruitées (suite)

Nombre d'itérations moyen parmi les simulations réussies



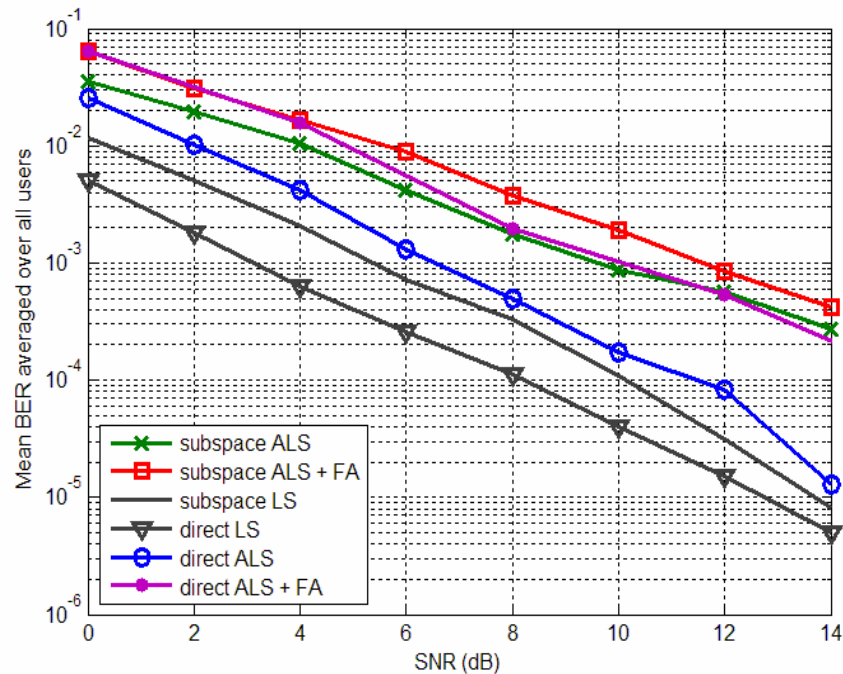
Temps moyen parmi les simulations réussies



Préservation de la structure Toeplitz

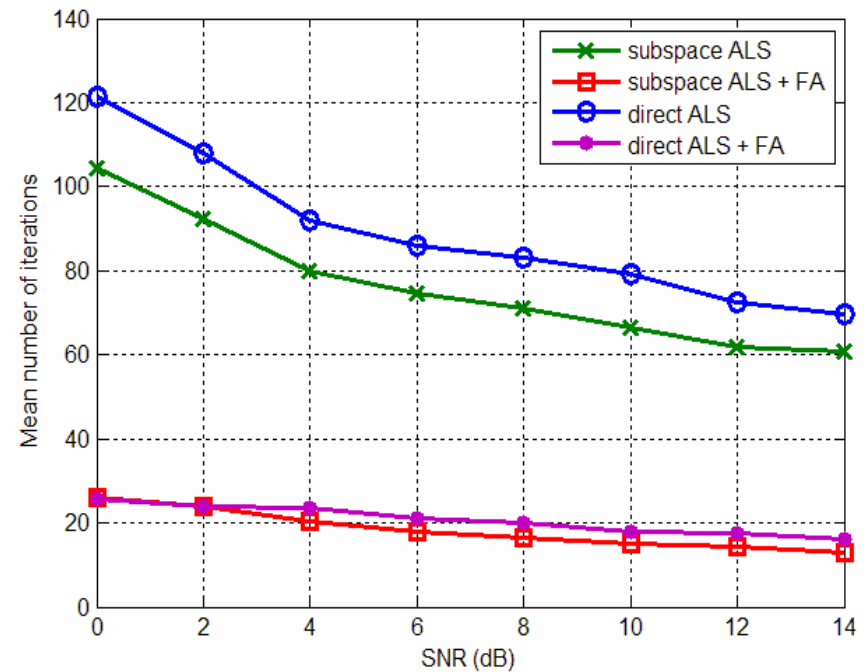
Impact de la préservation de la structure Toeplitz et de la projection sur une alphabet limité à chaque itération.

BER vs. SNR



Impact de la préservation de la structure Toeplitz et de la projection sur une alphabet limité à chaque itération.

Nb. d'itérations vs. SNR



Reformulation de la DCB-(L,L,1) en un problème de SD

$$y = \sum_{r=1}^R (\mathbf{H}_r \cdot \mathbf{S}_r^T) \circ \mathbf{a}_r = \sum_{r=1}^R \mathbf{X}_r \circ \mathbf{a}_r$$

Hypothèse:

$$R \leq \min(IJ, K)$$

2 expressions pour $\mathbf{Y}_{JI \times K}$

$$\begin{cases} \mathbf{Y} = (\text{vec}(\mathbf{X}_1) \cdots \text{vec}(\mathbf{X}_R)) \cdot \mathbf{A}^T = \tilde{\mathbf{X}} \cdot \mathbf{A}^T \\ \mathbf{Y} = \mathbf{U} \cdot \mathbf{\Sigma} \cdot \mathbf{V}^H = \mathbf{E} \cdot \mathbf{V}^H \end{cases}$$



$\exists \mathbf{W} \in \mathbb{C}^{R \times R}$

telle que

$$\begin{cases} \tilde{\mathbf{X}} = \mathbf{E} \cdot \mathbf{W} \\ \mathbf{A}^T = \mathbf{W}^{-1} \cdot \mathbf{V}^H \end{cases}$$

Objectif: trouver

\mathbf{W}

Reformulation de la DCB-(L,L,1) en un problème de SD (suite)

$$\tilde{\mathbf{X}} = \mathbf{E} \cdot \mathbf{W}$$

Pour $r = 1 \dots R$

$$\begin{matrix} | & & | \\ \mathbf{X}_r & = & \mathbf{W}_{1r} | & \mathbf{E}_1 & + \dots + & \mathbf{W}_{Rr} | & \mathbf{E}_R \\ | & & | & & & & | \\ \mathbf{J} & & \mathbf{J} & & & & \mathbf{J} \end{matrix}$$

Les coeff. de \mathbf{W} sont ceux de combinaisons linéaires des \mathbf{E}_r qui mènent aux matrices \mathbf{X}_r de rang L

Outil: mapping ψ pour la détection de rang-L. Soit $\mathbf{X}_r \in \mathbb{C}^{I \times J}$, alors $\psi(\mathbf{X}_r, \mathbf{X}_r, \dots, \mathbf{X}_r) = 0$ si et seulement si \mathbf{X}_r est au plus de rang L

\mathbf{W} peut ensuite être estimée par la diagonalisation simultanée du système de matrices

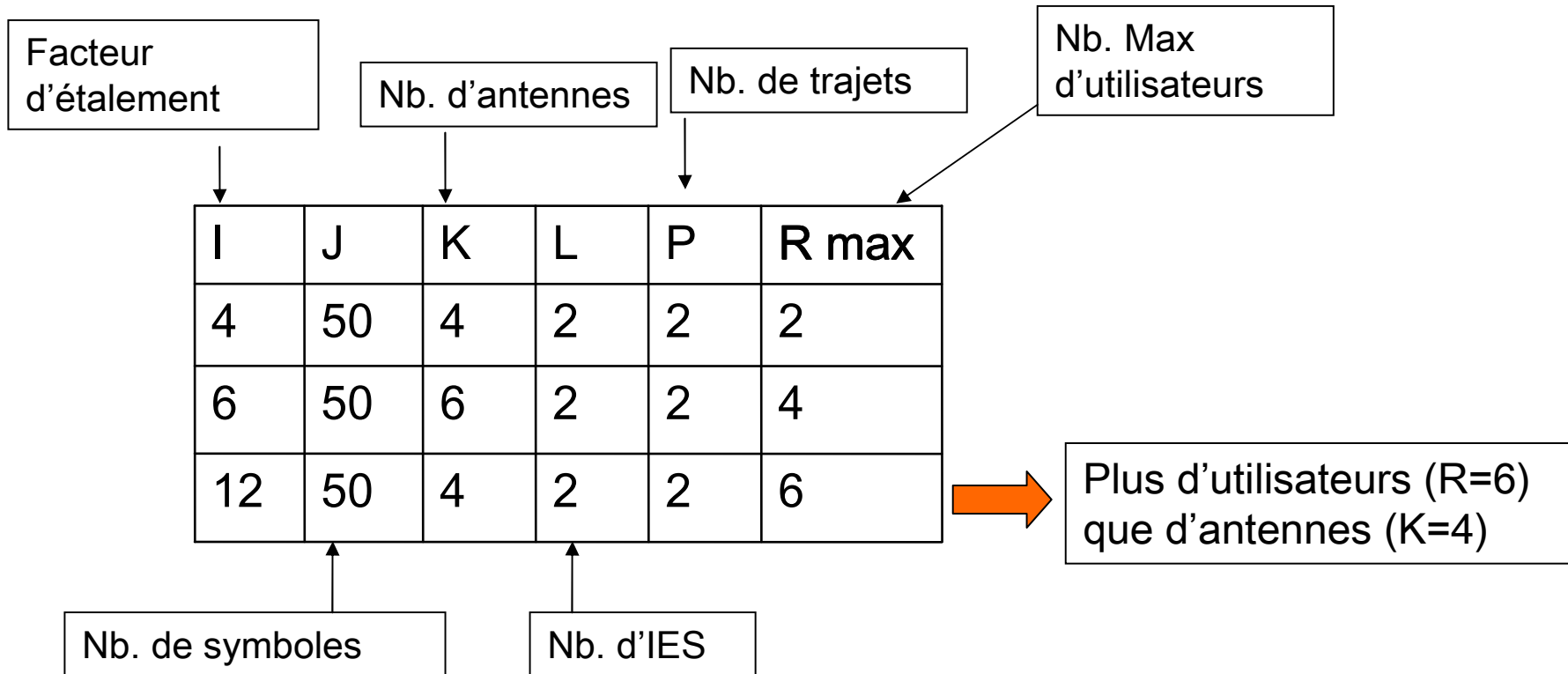
$$\begin{cases} \mathbf{M}_1 = \mathbf{W} \cdot \Lambda_1 \cdot \mathbf{W}^T \\ \mathbf{M}_2 = \mathbf{W} \cdot \Lambda_2 \cdot \mathbf{W}^T \\ \vdots \\ \mathbf{M}_R = \mathbf{W} \cdot \Lambda_I \cdot \mathbf{W}^T \end{cases}$$

Bornes d'unicité de la DCB-(L,P,..)

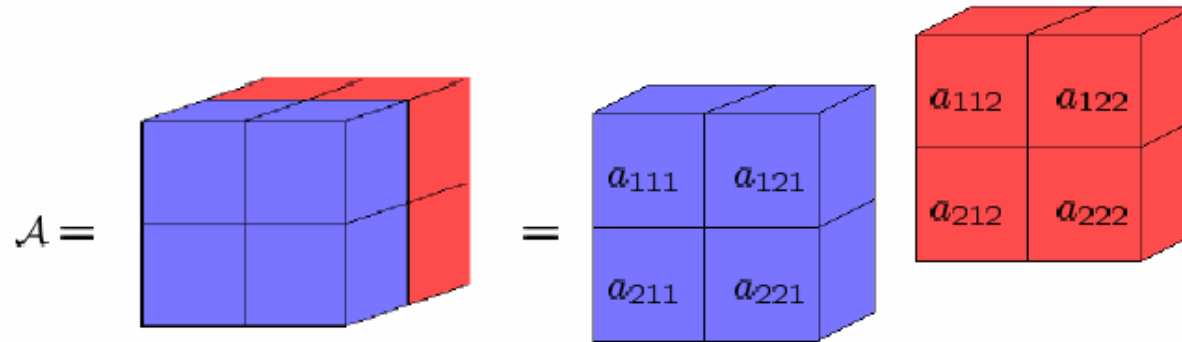
Borne suffisante 1

[De Lathauwer 2006]

$$\min \left(\left\lfloor \frac{I}{\max(L, P)} \right\rfloor, R \right) + \min \left(\left\lfloor \frac{J}{L} \right\rfloor, R \right) + \min \left(\left\lfloor \frac{K}{P} \right\rfloor, R \right) \geq 2R + 2$$



Écriture matricielle d'un Tenseur d'ordre 3 (1)



Découpage du tenseur selon 3 modes:

➡ Mode1: découpage 'gauche-droite'

$$A_{(1)} = \left[\begin{array}{cc|cc} a_{111} & a_{112} & a_{121} & a_{122} \\ a_{211} & a_{212} & a_{221} & a_{222} \end{array} \right]$$

Écriture matricielle d'un Tenseur d'ordre 3 (2)

→ Mode2: découpage 'avant-arrière'

$$A_{(2)} = \left[\begin{array}{cc|cc} a_{111} & a_{211} & a_{112} & a_{212} \\ a_{121} & a_{221} & a_{122} & a_{222} \end{array} \right]$$

→ Mode3: découpage 'haut-bas'

$$A_{(3)} = \left[\begin{array}{cc|cc} a_{111} & a_{121} & a_{211} & a_{221} \\ a_{112} & a_{122} & a_{212} & a_{222} \end{array} \right]$$

Quelques outils d'algèbre multilinéaire (1)

Soient A et B $\in \mathfrak{R}^{2 \times 2}$ A = $\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$ et B = $\begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix}$

Le produit de kronecker: $A \otimes B = \begin{pmatrix} a_{11}B & a_{12}B \\ a_{21}B & a_{22}B \end{pmatrix} \in \mathfrak{R}^{4 \times 4}$

Le produit de Kathri-Rao: $A \odot B = \begin{pmatrix} a_{11} \begin{pmatrix} b_{11} \\ b_{21} \end{pmatrix} & a_{12} \begin{pmatrix} b_{12} \\ b_{22} \end{pmatrix} \\ a_{21} \begin{pmatrix} b_{11} \\ b_{21} \end{pmatrix} & a_{22} \begin{pmatrix} b_{12} \\ b_{22} \end{pmatrix} \end{pmatrix} \in \mathfrak{R}^{4 \times 2}$
 (= produit de kronecker des colonnes)

Quelques outils d'algèbre multilinéaire (2)

Le rang de Kruskal (k-rank): si $k(A)=p$ alors n'importe quelle matrice formée de p colonnes de A est de rang plein, ceci n'étant pas vérifiée pour $p+1$.

exemple : $A = \begin{pmatrix} 0 & 1 & 2 \\ 1 & 0 & 0 \end{pmatrix}$ $\text{rank}(A)=2$ mais $k(A)=1$

Le k'-rank, généralise le k-rank: si $A \in \mathfrak{R}^{I*LR}$ est partitionnée $A=[A_1 \dots A_R]$ alors $k'(A)=p$ est le nombre max tel que n'importe quelle matrice formée de p sous-blocs de A présente des colonnes linéairement indépendantes.

exemple: $A = \left(\begin{array}{c|c|c|c} 1 & 0 & 0 & 0 & 2 & 0 \\ 0 & 1 & 0 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 & 3 & 0 \\ 0 & 0 & 0 & 1 & 0 & 3 \end{array} \right)$ $A_3=2A_1+3A_2$ donc $k'(A) \neq 3$
 $\text{rank}([A_1 \ A_2])=\text{rank}([A_1 \ A_3])=\text{rank}([A_2 \ A_3])=4$ donc
 $k'(A)=2$