

---

# Extension de Matching Pursuit au modèle des moindres carrés totaux

---

**Sébastien MARIA,  
IRISA/Université de Rennes 1**



---

# Introduction

Pourquoi de tels algorithmes?

- A l'origine: le problème de la sélection de variables en statistiques (depuis 1950).
- Nécessité de représenter une observation par un nombre restreint de signaux élémentaires: notion de parcimonie.
- Plus formellement:

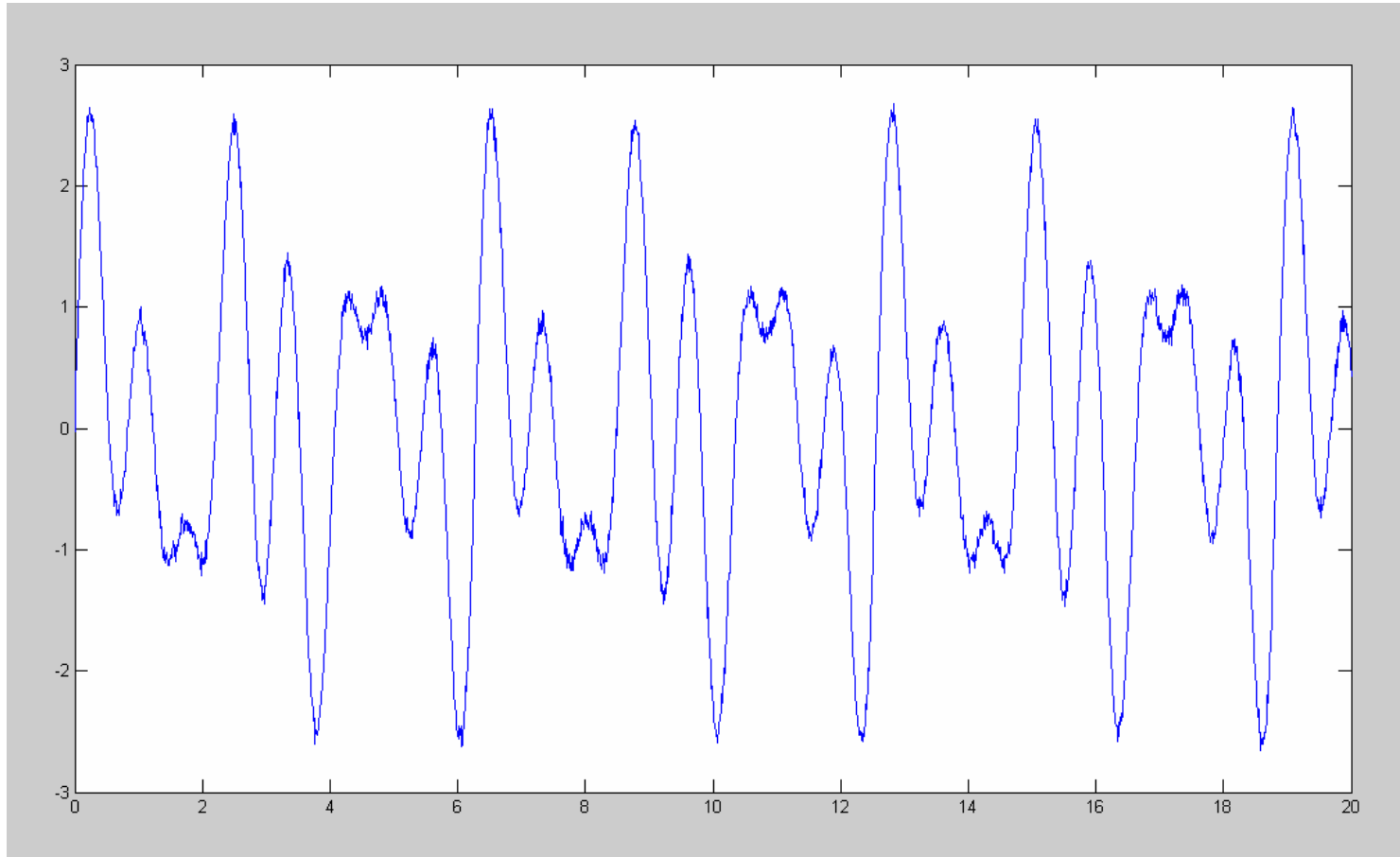
$$y = Ax + e = \sum_{i=1}^n a_i x_i + e$$

---

---

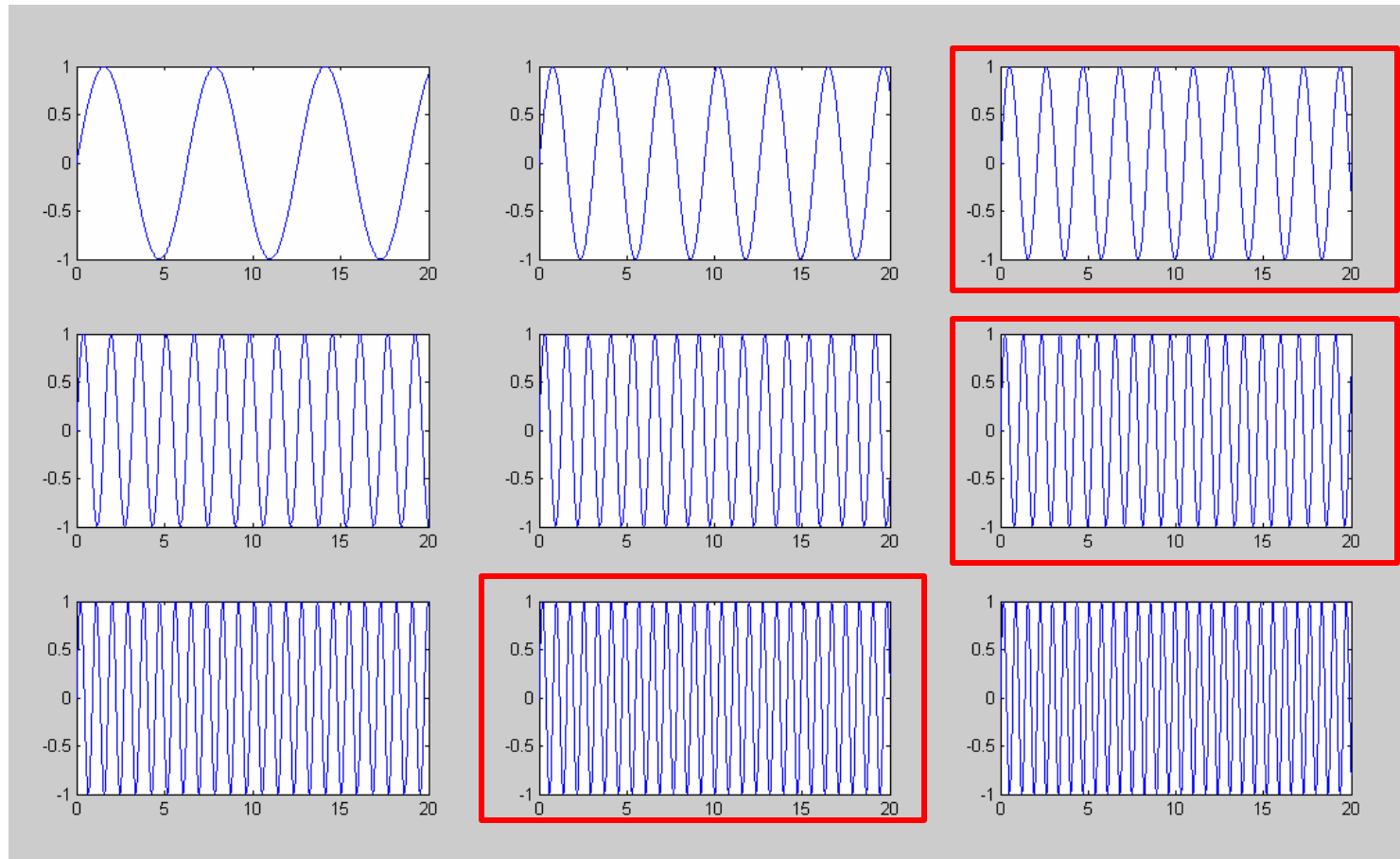
# Introduction

Exemple d'application: somme de sinusöides



# Introduction

Résulte d'une somme de sinusoides pures:



---

# Plan de l'exposé

1. L'algorithme Matching Pursuit
  2. Vers une version plus efficace: l'algorithme OMP
  3. Et avec robustesse: l'extension au modèle TLS
  4. Comparaison des différents algorithmes: quelques résultats expérimentaux
-

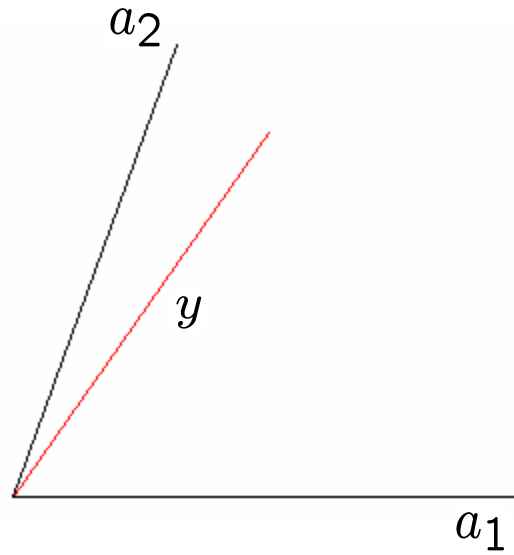
---

# Plan de l'exposé

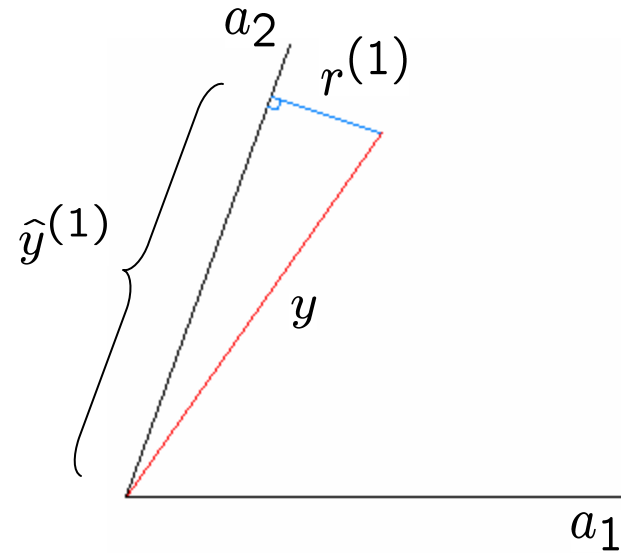
1. L'algorithme Matching Pursuit
  2. Vers une version plus efficace: l'algorithme OMP
  3. Et avec robustesse: l'extension au modèle TLS
  4. Comparaison des différents algorithmes: quelques résultats expérimentaux
-

---

# Représentation graphique dans le cas simple 2D



# Représentation graphique dans le cas simple 2D

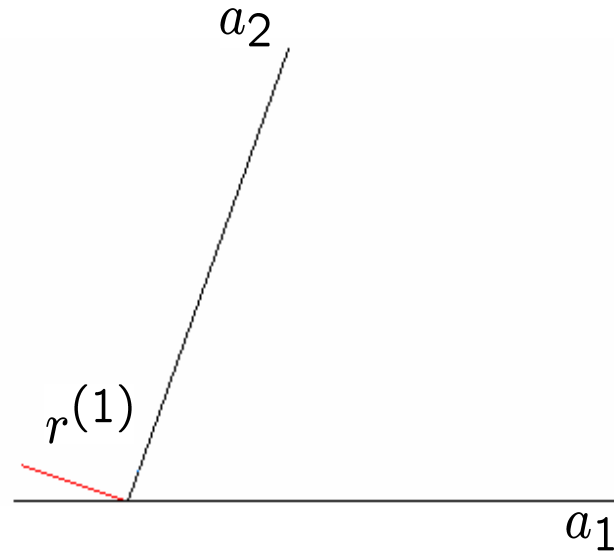


$$\hat{y}^{(1)} = \langle y, a_2 \rangle a_2$$

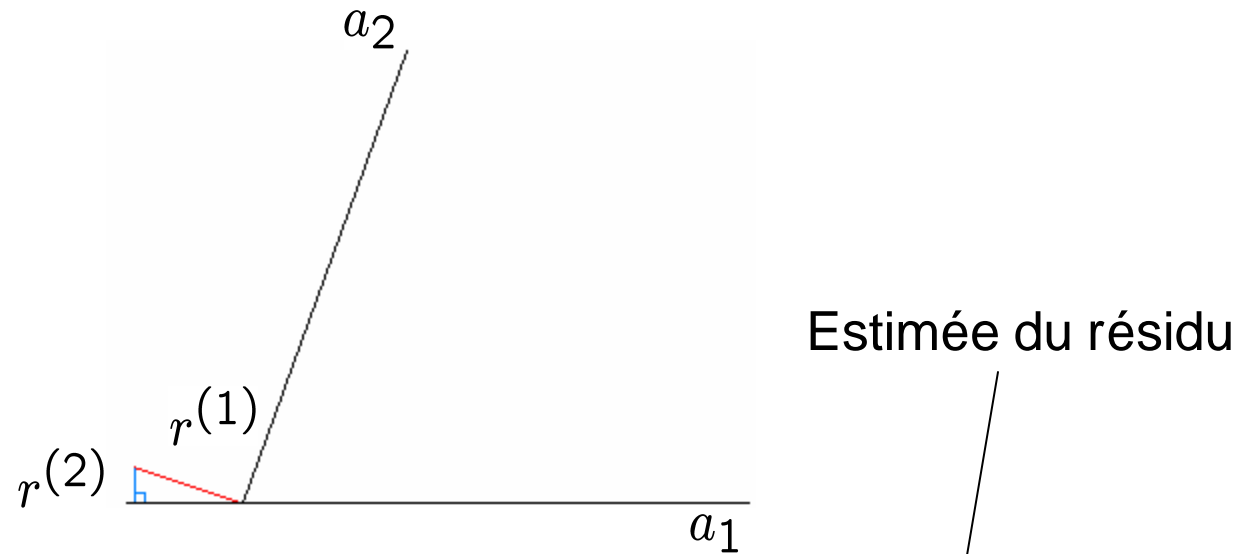
$$r^{(1)} = y - \hat{y}^{(1)}$$



# Représentation graphique dans le cas simple 2D



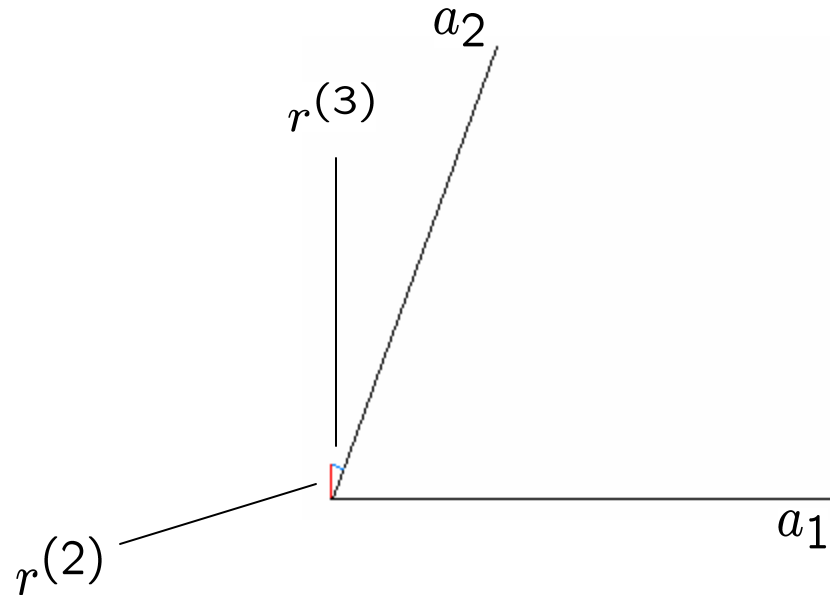
# Représentation graphique dans le cas simple 2D



$$\hat{y}^{(2)} = \hat{y}^{(1)} + \langle a_1, r^{(1)} \rangle a_1$$

$$r^{(2)} = y - \hat{y}^{(2)}$$

# Représentation graphique dans le cas simple 2D



$$\hat{y}^{(3)} = \hat{y}^{(2)} + \langle a_2, r^{(2)} \rangle a_2$$

$$r^{(3)} = y - \hat{y}^{(3)}$$

---

# Descriptif de l'algorithme MP (1/2)

## 1. Initialisation:

$$\hat{y}^{(0)} = 0, \quad r^{(0)} = y, \quad k = 1.$$

## 2. A l'itération $k$ :

- Calcul de toutes les corrélations

$$\{\langle a_i, r^{(k-1)} \rangle = a_i^T \cdot r^{(k-1)}\}_{i=1..n}$$

- Choix de la plus forte de ces corrélations:

$$i_k = \arg \max_i |\langle a_i, r^{(k-1)} \rangle|$$

---

---

# Descriptif de l'algorithme MP (2/2)

- Mise à jour de l'estimée et de son résidu:

$$\hat{y}^{(k)} = \hat{y}^{(k-1)} + \langle a_{i_k}, r^{(k-1)} \rangle a_{i_k}$$

$$r^{(k)} = y - \hat{y}^{(k)}$$

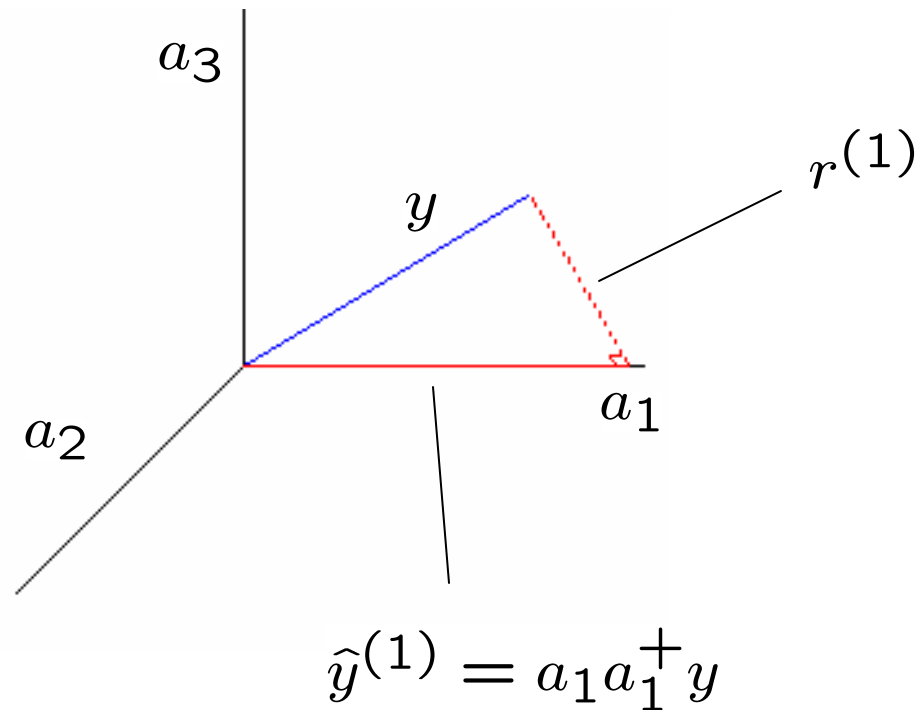
- Incrémentation  $k \Rightarrow k + 1$  et retour à l'étape 2 jusqu'à qu'un critère d'arrêt soit atteint
-

---

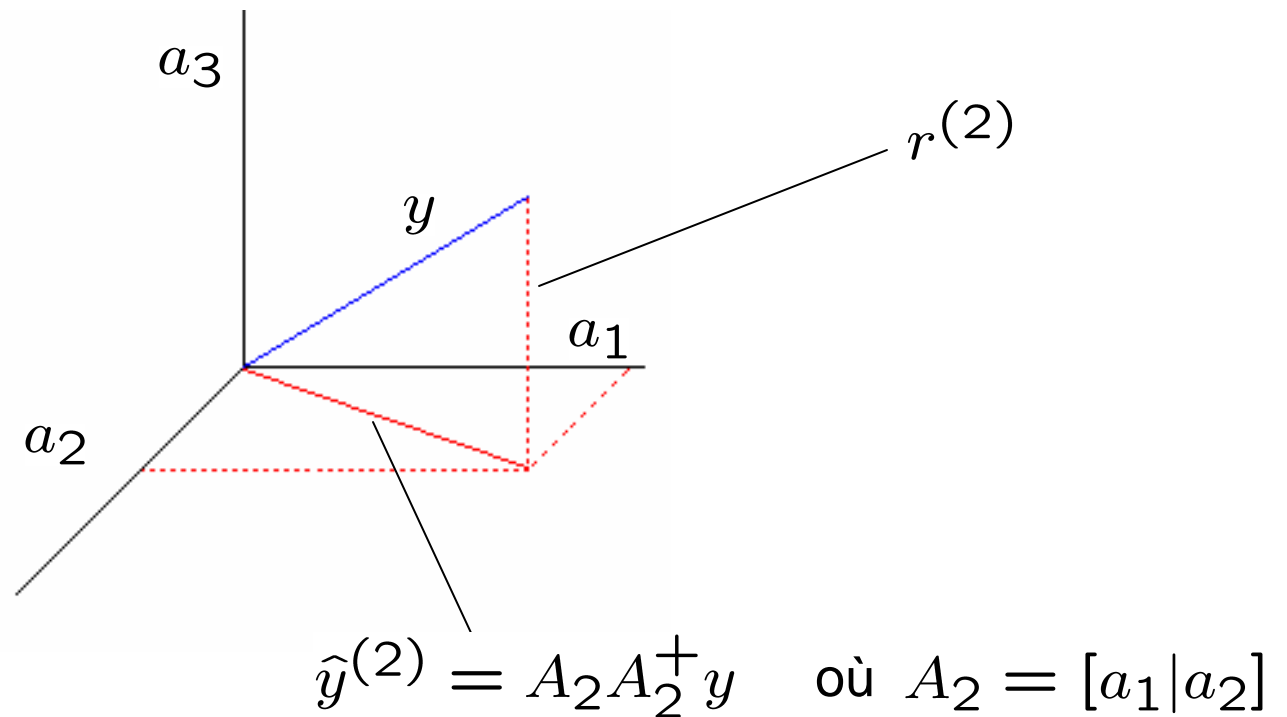
# Plan de l'exposé

1. L'algorithme Matching Pursuit
  2. Vers une version plus efficace: l'algorithme OMP
  3. Et avec robustesse: l'extension au modèle TLS
  4. Comparaison des différents algorithmes: quelques résultats expérimentaux
-

# Représentation graphique dans le cas 3D



# Représentation graphique dans le cas 3D





---

# Descriptif de l'algorithme OMP (1/2)

## 1. Initialisation:

$$\hat{y}^{(0)} = 0, \quad r^{(0)} = y, \quad A_0 = [], \quad k = 1.$$

## 2. A l'itération $k$ :

- Calcul de toutes les corrélations

$$i_k = \arg \max_i | \langle a_i, r^{(k-1)} \rangle |$$

- Choix de la plus forte de ces corrélations:

$$\{ \langle a_i, r^{(k-1)} \rangle = a_i^T \cdot r^{(k-1)} \}_{i=1..n}$$

---

---

# Descriptif de l'algorithme OMP (2/2)

- Mise à jour de l'estimée:

$$a^{(k)} = a_{i_k} \quad \text{puis} \quad A_k = [A_{k-1} \mid a^{(k)}]$$

$$\hat{x}^{(k)} = A_k^+ \hat{y}^{(k)} \Rightarrow \hat{y}^{(k)} = A_k \hat{x}^{(k)} = A_k A_k^+ y$$

$$r^{(k)} = y - \hat{y}^{(k)}$$

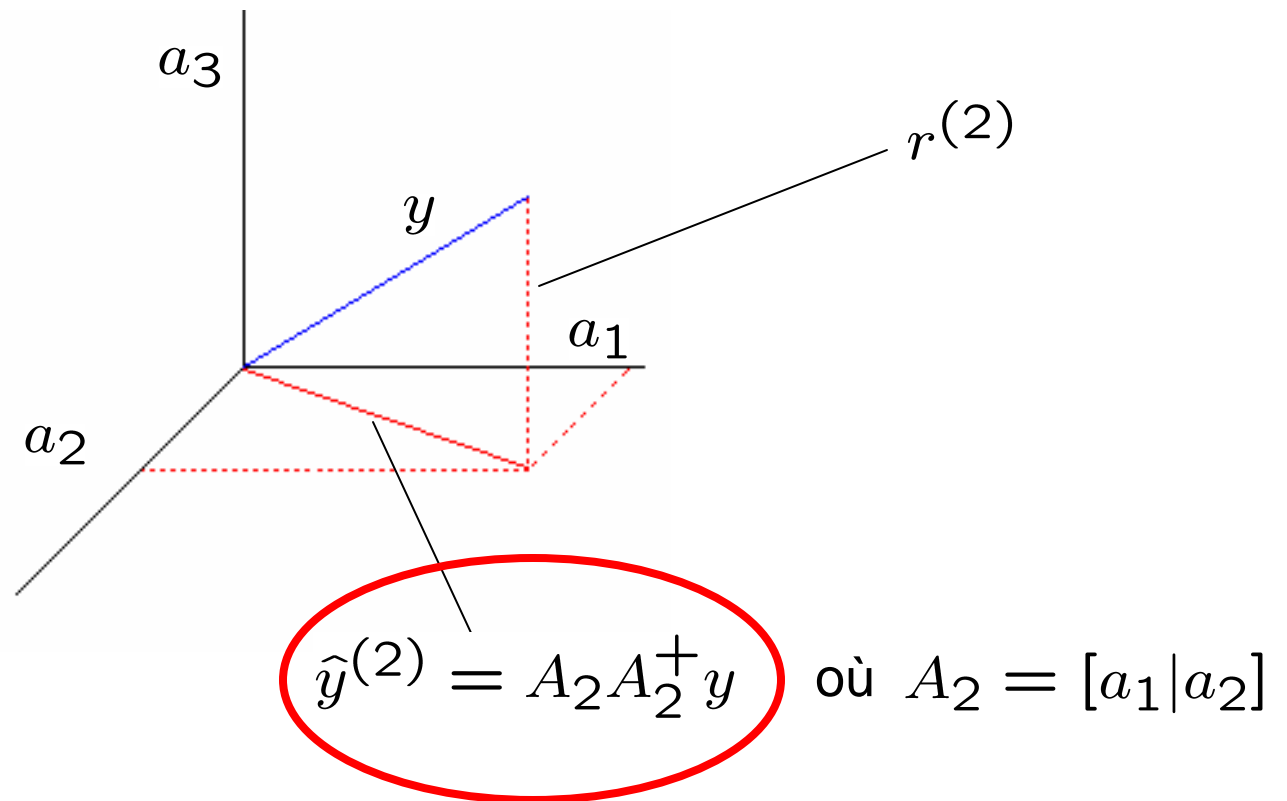
- Incrémentation  $k \leftarrow k + 1$  et retour à l'étape 2 jusqu'à qu'un critère d'arrêt soit atteint
-

---

# Plan de l'exposé

1. L'algorithme Matching Pursuit
  2. Vers une version plus efficace: l'algorithme OMP
  3. Et avec robustesse: l'extension au modèle TLS
  4. Comparaison des différents algorithmes: quelques résultats expérimentaux
-

# Représentation graphique dans le cas 3D



# Le modèle LS

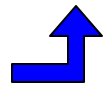
Le modèle des moindres carrés

$$y - e = Ax$$

où  $A \in \mathbb{R}^{n \times p}$  et  $e \sim \mathcal{N}(0, \sigma^2 I)$

$$\begin{bmatrix} y \end{bmatrix} - \begin{bmatrix} e \end{bmatrix} = \begin{bmatrix} A \end{bmatrix} \begin{bmatrix} x \end{bmatrix}$$

Bruit Gaussien



---

# Le modèle LS

Le critère à résoudre:

$$\min_x \|y - Ax\|_2^2$$

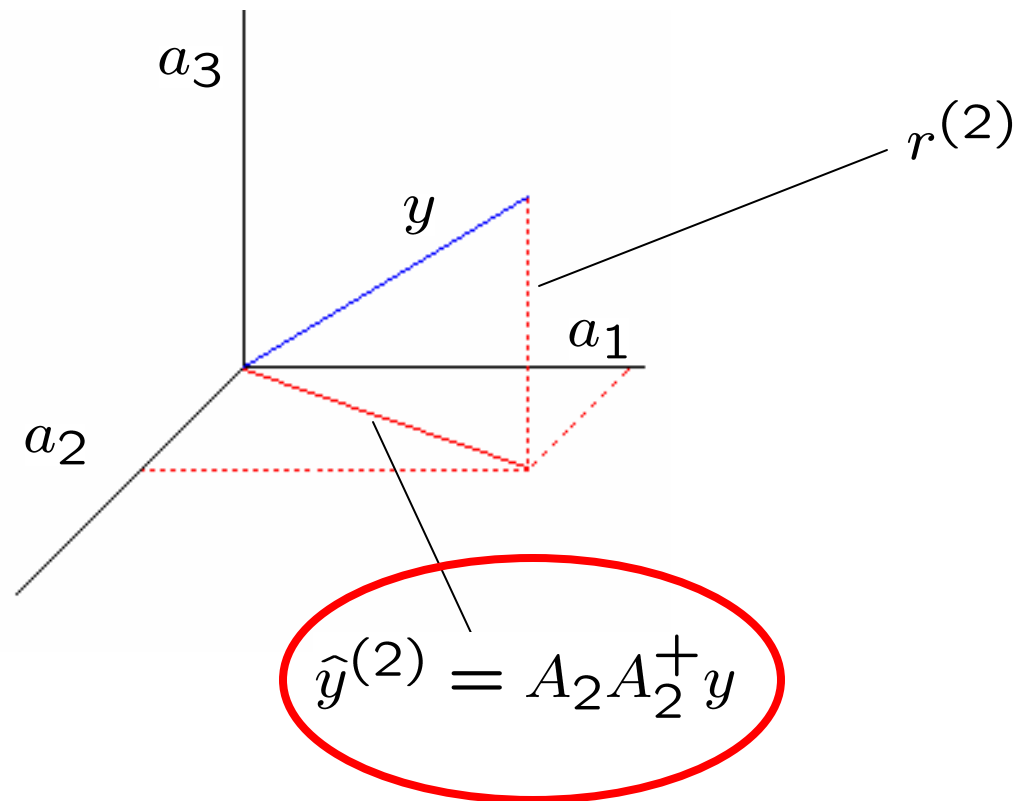
La solution des moindres carrés:

$$\hat{x} = A^+ y \quad \Rightarrow \quad \hat{y} = AA^+ y$$

Rappelez-vous

---

# Représentation graphique dans le cas 3D



---

**Et pourquoi ne pas utiliser un modèle plus robuste que LS?**

**C'est le modèle des moindres carrés totaux TLS (Total Least Squares)**

---

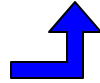
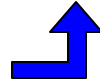


# Le modèle TLS

Le modèle de régression linéaire devient alors:

$$\begin{cases} y = Ax + e \\ Z = A + N \end{cases} \quad y \text{ et } Z \text{ sont les observations.}$$

$$\begin{bmatrix} y \end{bmatrix} - \begin{bmatrix} e \end{bmatrix} = \left( \begin{bmatrix} Z \end{bmatrix} - \begin{bmatrix} N \end{bmatrix} \right) \begin{bmatrix} x \end{bmatrix}$$

Bruit Gaussien       Bruit Gaussien 

---

# Le modèle TLS

## OBJECTIFS:

- Estimer le vecteur  $x$ .
  - Estimer le bruit perturbateur  $N$  afin de reconstruire la matrice  $A$ .
-

---

# Le modèle TLS

L'estimée du maximum de vraisemblance de  $x$  est obtenue en résolvant:

$$\min_{x, A} \|y - Ax\|_2^2 + \|Z - A\|_F^2$$

La solution est obtenue en utilisant la matrice augmentée

$$\hat{C} = [Z \quad : \quad y]$$

---

# Analyse de la solution

- La solution est obtenue à partir de la décomposition en valeur singulière:  $\hat{C} = \hat{U}\hat{S}\hat{V}^T$

avec  $\hat{U}, \hat{V}$  matrices unitaires et  $\hat{S} =$

$$\begin{bmatrix} \hat{s}_1 & 0 & \dots & 0 \\ 0 & \hat{s}_2 & \dots & 0 \\ \vdots & \dots & \dots & 0 \\ 0 & \dots & 0 & \hat{s}_{min} \\ \\ 0 & \dots & \dots & 0 \\ \vdots & & & \vdots \\ 0 & \dots & \dots & 0 \end{bmatrix}$$

- On en déduit les estimées:

$$\hat{x} = -\frac{\hat{v}_{min}(1:n)}{\hat{v}_{min}(n+1)}, \quad \hat{N}_k = \hat{s}_{min}^{(k)} [\hat{u}_{min}^{(k)}] [\hat{v}_{min}^{(k)}(1:n)]^T$$

---

# Descriptif de l'algorithme TLS - OMP

## 1. Initialisation:

$$\hat{y}^{(0)} = 0, \quad r^{(0)} = y, \quad Z_0 = [], \quad k = 1.$$

## 2. A l'itération $k$ :

- Calcul de toutes les corrélations

$$i_k = \arg \max_i | \langle z_i, r^{(k-1)} \rangle |$$

- Choix de la plus forte de ces corrélations:

$$\{ \langle z_i, r^{(k-1)} \rangle = z_i^T \cdot r^{(k-1)} \}_{i=1..n}$$

---

# Descriptif de l'algorithme TLS - OMP

□ Mise à jour de l'estimée:

- $z^{(k)} = z_{i_k}$  et  $Z_k = [Z_{k-1} \mid z^{(k)}]$
- Calcul de la SVD:  $\hat{C}_k = \hat{U}_k \hat{S}_k \hat{V}_k^T$
- Estimation de  $\hat{N}_k$  puis de  $\hat{A}_k = Z_k - \hat{N}_k$
- $\hat{x}^{(k)} = -\frac{v_{\min}^{(k)}(1:n)}{v_{\min}^{(k)}(n+1)} \Rightarrow \hat{y}^{(k)} = \hat{A}_k \hat{x}^{(k)}$
- $r^{(k)} = y - \hat{y}^{(k)}$

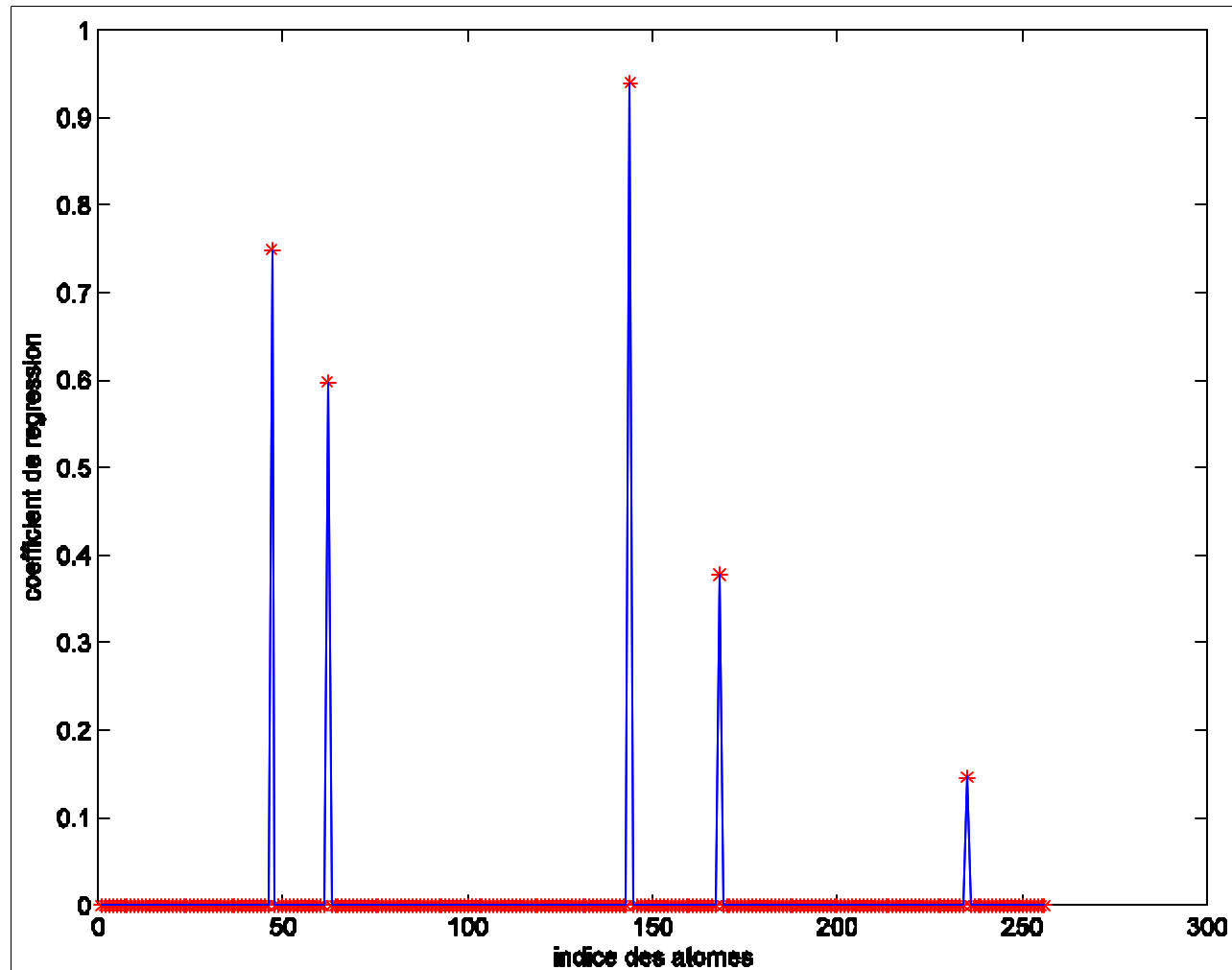
□ Incrémentation  $k \leftarrow k + 1$  et retour à l'étape 2 jusqu'à qu'un critère d'arrêt soit atteint

---

# Plan de l'exposé

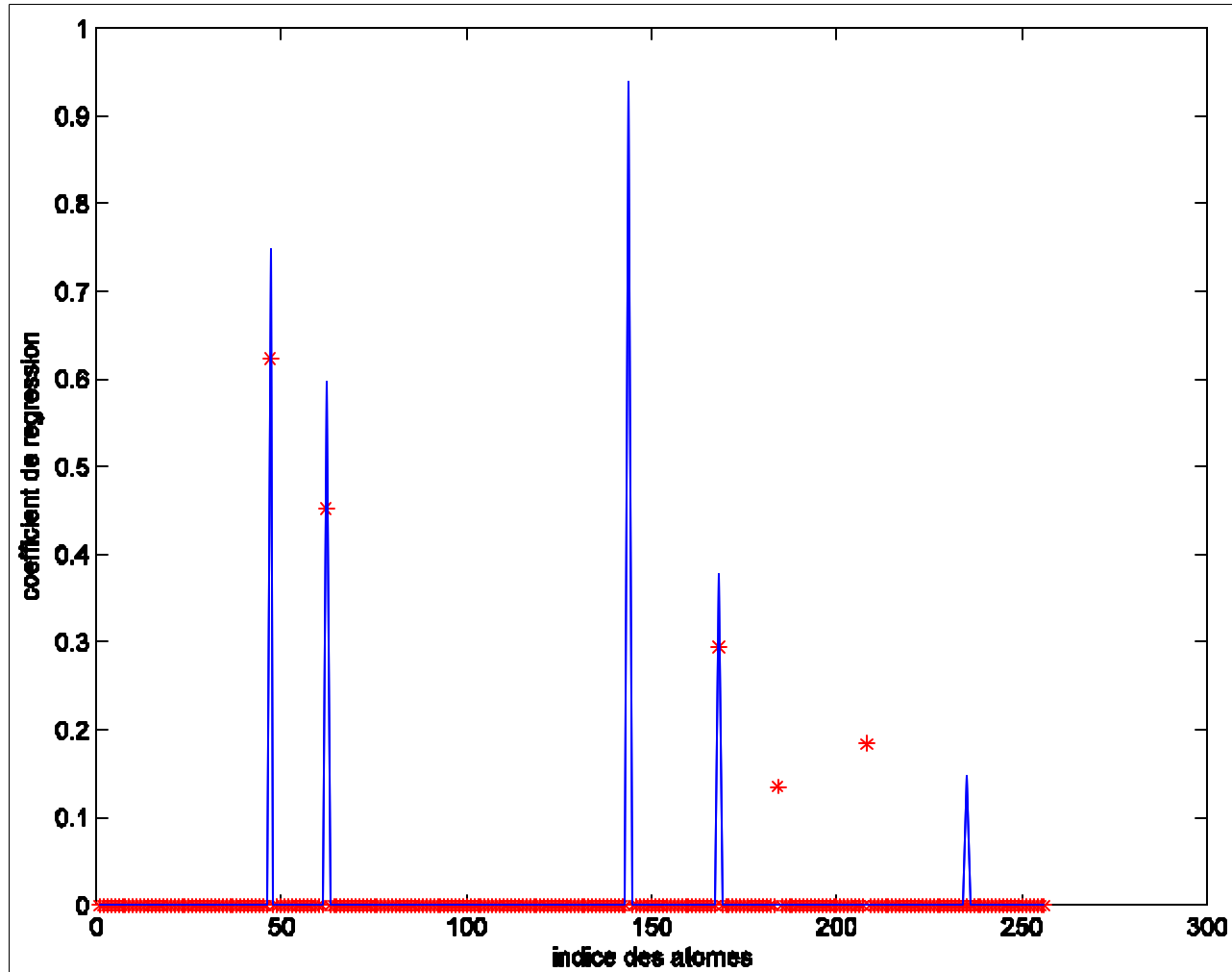
1. L'algorithme Matching Pursuit
  2. Vers une version plus efficace: l'algorithme OMP
  3. Et avec robustesse: l'extension au modèle TLS
  4. Comparaison des différents algorithmes: quelques résultats expérimentaux
-

# Performances dans le cas sans bruit

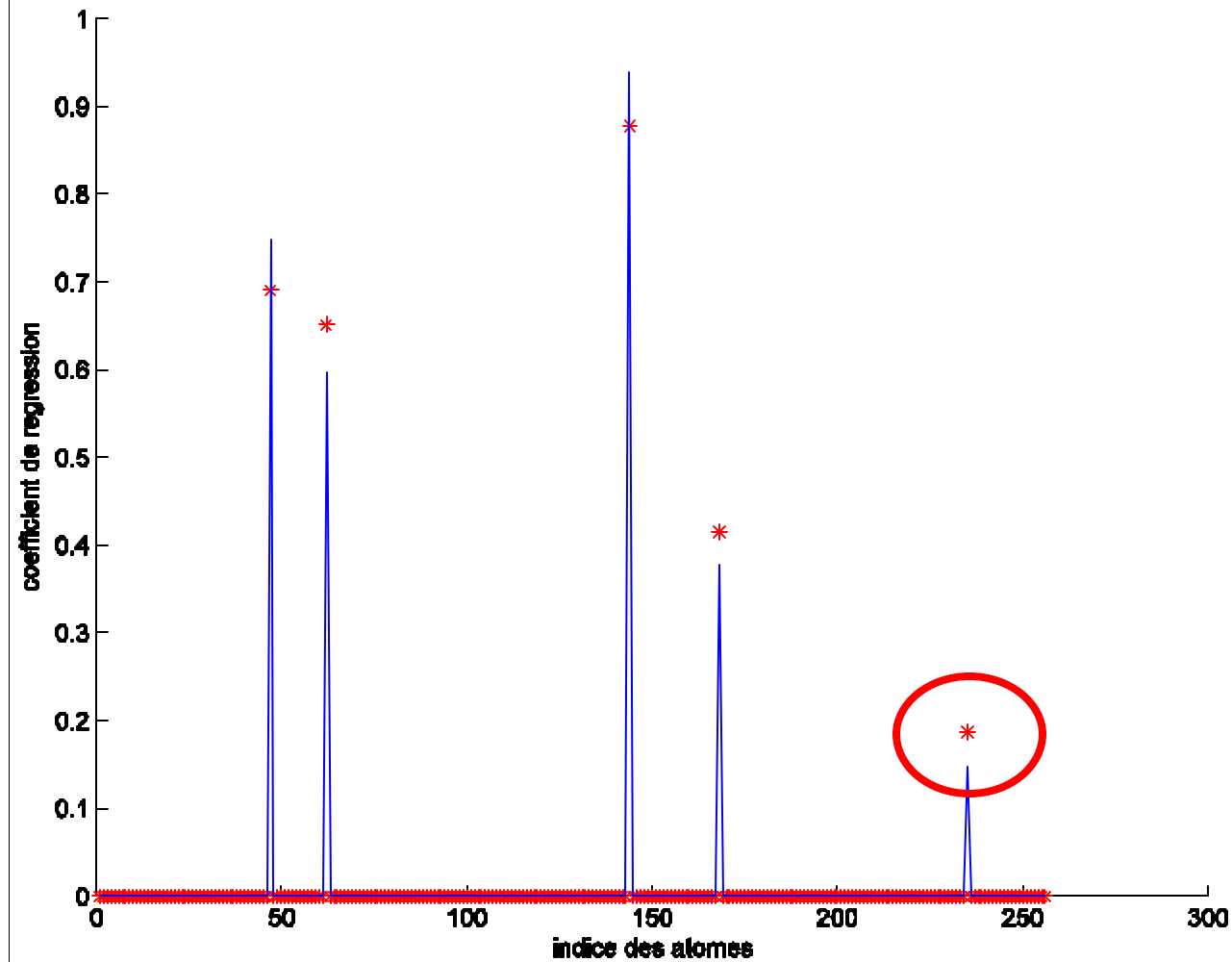




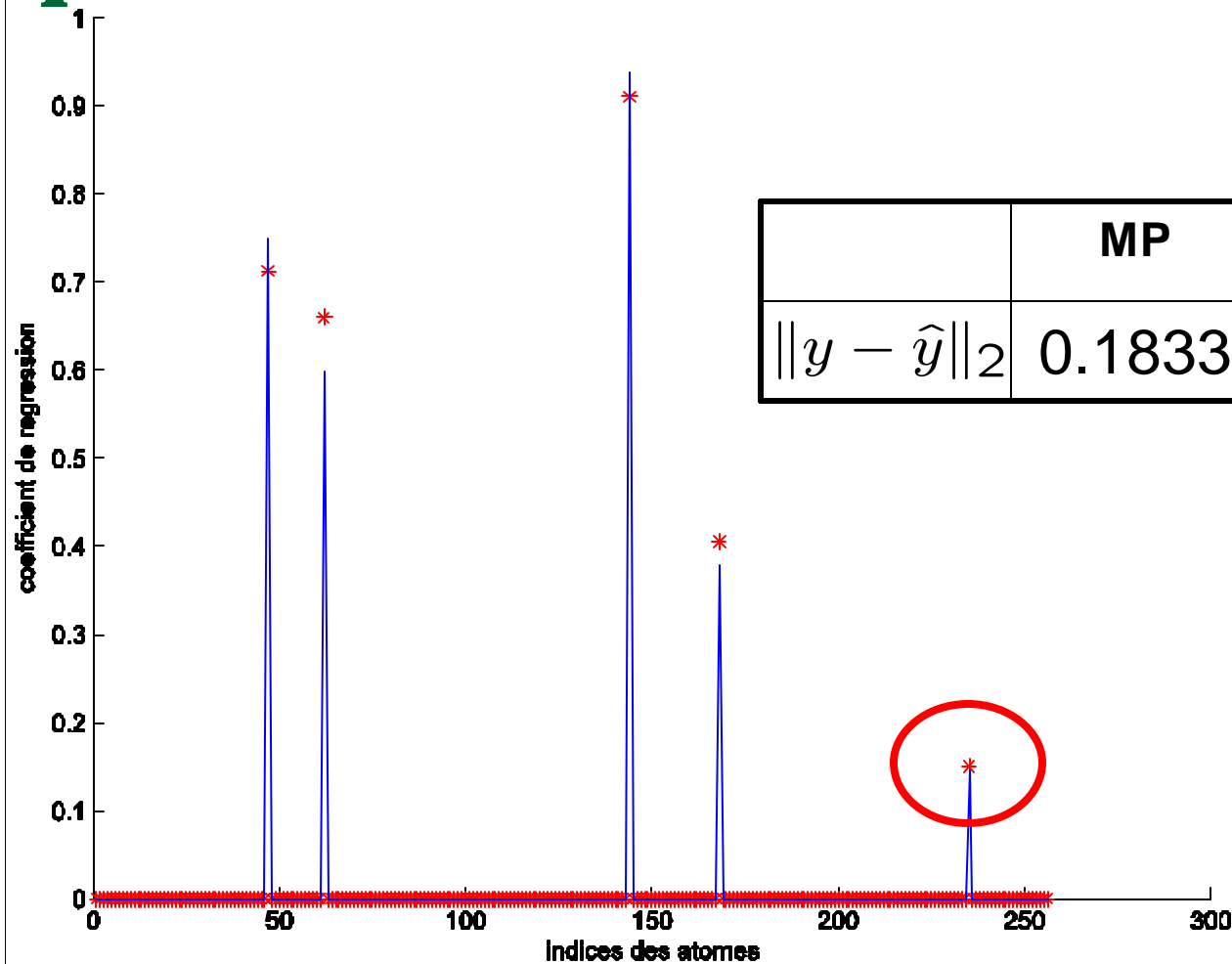
# Performances de MP en présence de bruit



# Performances de OMP en présence de bruit



# Performances de TLS-OMP en présence de bruit



	MP	OMP	TLS-OMP
$\ y - \hat{y}\ _2$	0.1833	0.2676	0.0911

---

# Conclusion

- Algorithme MP:



Itératif et peu coûteux, fonctionne très bien sans bruit



$n$  itérations ne donnent pas  $n$  atomes, peu robuste

- Algorithme OMP



Plus robuste,  $n$  itérations fournit  $n$  atomes



Plus coûteux

- Algorithme TLS-OMP



Robuste, à utiliser si incertitude sur les atomes



Plus cher qu'OMP

---

---

# Le modèle TLS

L'estimée du maximum de vraisemblance de  $x$  est obtenue en résolvant:

$$\min_{x, A} \|y - Ax\|_2^2 + \|Z - A\|_F^2$$

Ce qui est strictement équivalent à:

$$\min_{r, R, x} \|r\|_2^2 + \|R\|_F^2 \quad \text{sous} \quad y - r = (Z - R)x$$

---



# ANNEXES



---

# Le modèle TLS

L'estimée du maximum de vraisemblance de  $x$  est obtenue en résolvant:

$$\min_{x, A} \|y - Ax\|_2^2 + \|Z - A\|_F^2$$

Ce qui est strictement équivalent à:

$$\min_{r, R, x} \|r\|_2^2 + \|R\|_F^2 \quad \text{sous} \quad y - r = (Z - R)x$$

---

---

# Le modèle TLS

En utilisant les matrices augmentées suivantes

$$\hat{C} = [Z : y], \quad C = [A : Ax], \quad \Delta = [R : r]$$

Le problème peut alors se mettre sous la forme TLS habituelle

$$\min_{\Delta} \|\Delta\|_F^2 \quad \text{sous } \text{rang}(C) = \text{rang}(\hat{C}) - 1$$

---



# Le modèle TLS

En utilisant les matrices augmentées suivantes

$$\hat{C} = [Z \ : \ y], \quad C = [A \ : \ Ax], \quad \Delta = [R \ : \ r]$$

Le problème peut alors se mettre sous la forme TLS habituelle

$$\min_{\Delta} \|\Delta\|_F^2 \quad \text{sous} \quad \text{rang}(C) = \text{rang}(\hat{C}) - 1$$
$$\left( \begin{bmatrix} Z & y \\ R & r \end{bmatrix} - \begin{bmatrix} R & r \end{bmatrix} \right) \begin{bmatrix} x \\ -1 \end{bmatrix} = 0$$

# Analyse de la solution

- La solution est obtenue à partir de la décomposition en valeur singulière:  $\hat{C} = \hat{U}\hat{S}\hat{V}^T$

avec  $\hat{U}, \hat{V}$  matrices unitaires et  $\hat{S} =$

$$\begin{bmatrix} \hat{s}_1 & 0 & \dots & 0 \\ 0 & \hat{s}_2 & \dots & 0 \\ \vdots & \dots & \dots & 0 \\ 0 & \dots & 0 & \hat{s}_{min} \\ \\ 0 & \dots & \dots & 0 \\ \vdots & & & \vdots \\ 0 & \dots & \dots & 0 \end{bmatrix}$$

- On en déduit les estimées:

$$\hat{x} = -\frac{\hat{v}_{min}(1:n)}{\hat{v}_{min}(n+1)}, \quad \hat{N}_k = \hat{s}_{min}^{(k)} [\hat{u}_{min}^{(k)}] [\hat{v}_{min}^{(k)}(1:n)]^T$$

---

# Descriptif de l'algorithme TLS - OMP

## 1. Initialisation:

$$\hat{y}^{(0)} = 0, \quad r^{(0)} = y, \quad Z_0 = [], \quad k = 1.$$

## 2. A l'itération $k$ :

- Calcul de toutes les corrélations

$$i_k = \arg \max_i | \langle z_i, r^{(k-1)} \rangle |$$

- Choix de la plus forte de ces corrélations:

$$\{ \langle z_i, r^{(k-1)} \rangle = z_i^T \cdot r^{(k-1)} \}_{i=1..n}$$

---

# Descriptif de l'algorithme TLS - OMP

□ Mise à jour de l'estimée:

-  $z^{(k)} = z_{i_k}$  et  $Z_k = [Z_{k-1} \mid z^{(k)}]$

- Calcul de la SVD:  $\hat{C}_k = \hat{U}_k \hat{S}_k \hat{V}_k^T$

- Sélection de  $\hat{u}_{min}^{(k)}$ ,  $\hat{s}_{min}^{(k)}$ ,  $\hat{v}_{min}^{(k)}$

- Estimation de  $\hat{N}_k = \hat{s}_{min}^{(k)} [\hat{u}_{min}^{(k)}] [\hat{v}_{min}^{(k)}(1:n)]^T$

Puis de  $\hat{A}_k = Z_k - \hat{N}_k$

-  $\hat{x}^{(k)} = -\frac{v_{min}^{(k)}(1:n)}{v_{min}^{(k)}(n+1)} \Rightarrow \hat{y}^{(k)} = \hat{A}_k \hat{x}^{(k)}$

-  $r^{(k)} = y - \hat{y}^{(k)}$

---

# Descriptif de l'algorithme TLS - OMP

- Incrémentation  $k \leftarrow k + 1$  et retour à l'étape 2 jusqu'à qu'un critère d'arrêt soit atteint

