

Calcul Scientifique Parallèle

Cours AMS301 — Automne 2021 — Cours 4

Résolution de systèmes linéaires : Méthodes itératives (1)
Méthodes stationnaires & Application aux différences finies

Axel Modave

On cherche $\mathbf{x} \in \mathbb{R}^n$ tel que $\mathbf{Ax} = \mathbf{b}$ avec $\mathbf{A} \in \mathbb{R}^{n \times n}$ et $\mathbf{b} \in \mathbb{R}^n$.

Motivation

Les systèmes linéaires apparaissent dans beaucoup de procédures de calcul :

- problèmes physiques linéaires,
- problèmes physiques non-linéaires,
- problèmes d'optimisation,
- ...

pour beaucoup d'applications (*imagerie sismique, imagerie médicale, étude aéroacoustique, CFD, compatibilité électromagnétique, finance, ...*)

Du point de vue mathématique :

- problèmes elliptiques discrétisés
=> système linéaire à résoudre !
- problèmes hyperboliques discrétisés avec un schéma implicite en temps
=> système linéaire à résoudre à chaque pas de temps !

Résolution de systèmes linéaires — Généralités [2/3]

On cherche $\mathbf{x} \in \mathbb{R}^n$ tel que $\mathbf{Ax} = \mathbf{b}$ avec $\mathbf{A} \in \mathbb{R}^{n \times n}$ et $\mathbf{b} \in \mathbb{R}^n$.

Procédures de résolution

- ▶ **Méthodes directes** : Factorisation de \mathbf{A} en matrices triangulaires et/ou diagonales (ex. $\mathbf{A} = \mathbf{LU}$) et résolution de problèmes simples.

$$\mathbf{Ax} = \mathbf{b} \Leftrightarrow \mathbf{LUx} = \mathbf{b} \Leftrightarrow \begin{cases} \mathbf{Ly} = \mathbf{b} \\ \mathbf{Ux} = \mathbf{y} \end{cases}$$

Avantages : solution exacte obtenue après un nombre fini d'opérations
Difficultés : coût important (*calcul/mémoire*), parallélisation difficile

- ▶ **Méthodes itératives** : Procédure itérative permettant de minimiser une norme de l'erreur $\|\mathbf{x}^{(k)} - \mathbf{x}_{\text{ref}}\|$ et/ou d'une norme du résidu $\|\mathbf{Ax}^{(k)} - \mathbf{b}\|$.

$$\begin{cases} \mathbf{x}^{(0)} = \text{Iter}_{(0)}(\mathbf{A}, \mathbf{b}) \\ \mathbf{x}^{(k+1)} = \text{Iter}^{(k+1)}(\mathbf{x}^{(k)}, \mathbf{x}^{(k-1)}, \dots, \mathbf{A}, \mathbf{b}), \text{ pour } k \geq 0 \end{cases}$$

Avantages : coût limité par itération (*calcul/mémoire*), parallélisation facile/efficace
Difficultés : solution approchée, contrôle de la convergence du processus

2

Résolution de systèmes linéaires — Généralités [1/3]

Résolution de systèmes linéaires — Généralités [3/3]

On cherche $\mathbf{x} \in \mathbb{R}^n$ tel que $\mathbf{Ax} = \mathbf{b}$ avec $\mathbf{A} \in \mathbb{R}^{n \times n}$ et $\mathbf{b} \in \mathbb{R}^n$.

- ▶ **Méthodes itératives** :

$$\begin{cases} \mathbf{x}^{(0)} = \text{Iter}_{(0)}(\mathbf{A}, \mathbf{b}) \\ \mathbf{x}^{(k+1)} = \text{Iter}^{(k+1)}(\mathbf{x}^{(k)}, \mathbf{x}^{(k-1)}, \dots, \mathbf{A}, \mathbf{b}), \text{ pour } k \geq 0 \end{cases}$$

L'ordre de la méthode est le nombre de pas dont dépend l'itération courante.

Méthode **stationnaire** si les fonctions $\text{Iter}^{(k)}$ sont indép. de k . Sinon : **instationnaire**

Méthode **linéaire** si les fonctions $\text{Iter}^{(k)}$ sont linéaires. Sinon : **non-linéaire**

Pour cette séance, on considère des schémas stationnaires linéaires du 1^{er} ordre :

$$\begin{cases} \mathbf{x}^{(0)} \text{ donné} \\ \mathbf{x}^{(k+1)} = \mathbf{Bx}^{(k)} + \mathbf{f}, \text{ } k \geq 0 \end{cases}$$

où $\mathbf{B} \in \mathbb{R}^{n \times n}$ est la **matrice d'itération** et $\mathbf{f} \in \mathbb{R}^n$ dépend de \mathbf{b} .

1

3

Méthodes itératives pour les systèmes linéaires

Méthodes stationnaires

Système issu d'une discrétisation par différences finies

Méthodes stationnaires — Généralités [1/2]

On cherche $\mathbf{x} \in \mathbb{R}^n$ tel que $\mathbf{Ax} = \mathbf{b}$ avec $\mathbf{A} \in \mathbb{R}^{n \times n}$ et $\mathbf{b} \in \mathbb{R}^n$.

Pour commencer, on considère un schéma général de la forme :

$$\begin{cases} \mathbf{x}^{(0)} \text{ donné} \\ \mathbf{x}^{(k+1)} = \mathbf{Bx}^{(k)} + \mathbf{f}, \quad k \geq 0 \end{cases}$$

où $\mathbf{B} \in \mathbb{R}^{n \times n}$ est la **matrice d'itération** et $\mathbf{f} \in \mathbb{R}^n$ dépend de \mathbf{b} .

Définitions et propriétés

- La méthode est **consistante** si la sol. est un point fixe du schéma (i.e. $\mathbf{x} = \mathbf{Bx} + \mathbf{f}$).

OK si et seulement si $\mathbf{f} = (\mathbf{I} - \mathbf{B})\mathbf{A}^{-1}\mathbf{b}$

- La méthode est **convergente** si, $\forall \mathbf{x}^{(0)}$, $\lim_{k \rightarrow \infty} \mathbf{x}^{(k)} = \mathbf{x}$.

Pour une méthode consistante, OK si et seulement si $\rho(\mathbf{B}) < 1$

Le rayon spectral $\rho(\mathbf{B})$ est le plus grand des modules des valeurs propres de \mathbf{B} .

4

Méthodes stationnaires — Généralités [2/2]

On cherche $\mathbf{x} \in \mathbb{R}^n$ tel que $\mathbf{Ax} = \mathbf{b}$ avec $\mathbf{A} \in \mathbb{R}^{n \times n}$ et $\mathbf{b} \in \mathbb{R}^n$.

Pour commencer, on considère un schéma général de la forme :

$$\begin{cases} \mathbf{x}^{(0)} \text{ donné} \\ \mathbf{x}^{(k+1)} = \mathbf{Bx}^{(k)} + \mathbf{f}, \quad k \geq 0 \end{cases}$$

où $\mathbf{B} \in \mathbb{R}^{n \times n}$ est la **matrice d'itération** et $\mathbf{f} \in \mathbb{R}^n$ dépend de \mathbf{b} .

Définitions et propriétés (suite)

- On utilise des **critères d'arrêt** sur le nombre d'itération et la norme du résidu :

$$k \leq k_{\text{tol}} \quad \text{et} \quad \|\mathbf{r}^{(k)}\| / \|\mathbf{r}^{(0)}\| \leq \varepsilon_{\text{tol}}$$

avec le **résidu** $\mathbf{r}^{(k)} := \mathbf{b} - \mathbf{Ax}^{(k)}$.

Pour les méthodes stationnaires, on a :

$$\begin{aligned} \mathbf{Ae}^{(k)} &= \mathbf{Ax} - \mathbf{Ax}^{(k)} = \mathbf{r}^{(k)} \\ \Rightarrow \|\mathbf{r}^{(k)}\| &\leq \|\mathbf{A}\| \|\mathbf{e}^{(k)}\| \quad \text{et} \quad \|\mathbf{e}^{(k)}\| \leq \|\mathbf{A}^{-1}\| \|\mathbf{r}^{(k)}\| \end{aligned}$$

→ Quel \mathbf{B} pour une convergence rapide et une implémentation efficace ?

5

Méthodes stationnaires — Méthodes classiques [1/3]

On considère la décomposition régulière : $\mathbf{A} = \mathbf{M} - \mathbf{N}$ où $\mathbf{M} \in \mathbb{R}^{n \times n}$ est inversible

Méthode stationnaire

```
 $\mathbf{x}^{(0)} \in \mathbb{C}^n$ 
for  $k = 0, 1, \dots$  do
  |  $\mathbf{Mx}^{(k+1)} = \mathbf{Nx}^{(k)} + \mathbf{b}$ 
end
```

Choix pour les matrices :

	Par points	Par blocs
Jacobi	$\mathbf{M} = \mathbf{D}$	$\mathbf{M} = \mathbf{D}^{\text{blk}}$
Gauss-Seidel	$\mathbf{M} = \mathbf{D} + \mathbf{L}$	$\mathbf{M} = \mathbf{D}^{\text{blk}} + \mathbf{L}^{\text{blk}}$

$$\underbrace{\begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \end{bmatrix}}_{\mathbf{A}} = \underbrace{\begin{bmatrix} \times & & \\ & \times & \\ & & \times \end{bmatrix}}_{\mathbf{D}} + \underbrace{\begin{bmatrix} & \times & \\ & & \times \\ \times & & \times \end{bmatrix}}_{\mathbf{L}} + \underbrace{\begin{bmatrix} & & \times \\ & & \times \\ & & \times \end{bmatrix}}_{\mathbf{U}}$$

6

On considère la décomposition régulière : $\mathbf{A} = \mathbf{M} - \mathbf{N}$ où $\mathbf{M} \in \mathbb{R}^{n \times n}$ est inversible

Méthode stationnaire avec relaxation

```

 $\mathbf{x}^{(0)} \in \mathbb{C}^n$ 
for  $k = 0, 1, \dots$  do
  |  $\mathbf{M}\bar{\mathbf{x}} = \mathbf{N}\mathbf{x}^{(k)} + \mathbf{b}$ 
  |  $\mathbf{x}^{(k+1)} = \omega\bar{\mathbf{x}} + (1 - \omega)\mathbf{x}^{(k)}$ 
end
    
```

(ω est un paramètre réel)

Choix pour les matrices :

	Par points	Par blocs
Jacobi over relaxation (JOR)	$\mathbf{M} = \mathbf{D}$	$\mathbf{M} = \mathbf{D}^{\text{blk}}$
Successive over relaxation (SOR)	$\mathbf{M} = \mathbf{D} + \mathbf{L}$	$\mathbf{M} = \mathbf{D}^{\text{blk}} + \mathbf{L}^{\text{blk}}$

$$\underbrace{\begin{bmatrix} [\times] & [\times] & [\times] \\ [\times] & [\times] & [\times] \\ [\times] & [\times] & [\times] \end{bmatrix}}_{\mathbf{A}^{\text{blk}}} = \underbrace{\begin{bmatrix} [\times] \\ & [\times] \\ & & [\times] \end{bmatrix}}_{\mathbf{D}^{\text{blk}}} + \underbrace{\begin{bmatrix} [\times] \\ [\times] & [\times] \\ & [\times] & [\times] \end{bmatrix}}_{\mathbf{L}^{\text{blk}}} + \underbrace{\begin{bmatrix} [\times] & [\times] \\ & [\times] \end{bmatrix}}_{\mathbf{U}^{\text{blk}}}$$

Convergence des méthodes stationnaires

- ▶ Convergence si et seulement si $\rho(\mathbf{B}) < 1$ avec $\mathbf{B} = \mathbf{M}^{-1}\mathbf{N}$.
- ▶ Si \mathbf{A} est à diagonale strictement dominante (i.e. $|a_{ii}| > \sum_{j \neq i} |a_{ij}|, \forall i$)
 - Jacobi converge
 - GS converge
 - SOR converge si $0 < \omega \leq 1$
- ▶ Si \mathbf{A} est SDP (i.e. $\mathbf{A} = \mathbf{A}^*$ et $(\mathbf{A}\mathbf{x}, \mathbf{x}) > 0, \forall \mathbf{x} \neq 0$)
 - Jacobi converge si $(2\mathbf{D} - \mathbf{A})$ SDP
 - GS converge
 - SOR converge si et seulement si $0 < \omega < 2$

Rapidité de convergence (d'une manière générale) :

Jacobi < Gauss-Seidel < SOR
Par points < Par blocs

Méthode de Jacobi par points ($\mathbf{M} = \mathbf{D}$ et $\mathbf{N} = -(\mathbf{L} + \mathbf{U})$)

Méthode de Jacobi par points

```

 $\mathbf{x}^{(0)} \in \mathbb{R}^n$ 
for  $k = 0, 1, \dots$  do
  |  $\mathbf{D}\mathbf{x}^{(k+1)} = \mathbf{b} - (\mathbf{L} + \mathbf{U})\mathbf{x}^{(k)}$ 
end
    
```

Méthode de Jacobi par points (réécriture)

```

 $x_i^{(0)} \in \mathbb{R}$  pour  $i = 1 \dots n$ 
for  $k = 0, 1, \dots$  do
  | for  $i = 1 \dots n$  do
  | |  $x_i^{(k+1)} = a_{ii}^{-1} (b_i - \sum_{j \neq i} a_{ij} x_j^{(k)})$ 
  | end
end
    
```

Discussion

- Les itérations de la boucle intérieure sont indépendantes.
- Produit matrice-vecteur avec matrice pleine ($\mathbf{L} + \mathbf{U}$) (BLAS en //)
- Combinaisons linéaires (vu \mathbf{D} diagonale) (Combi. lin. en //)

Méthode de Gauss-Seidel par points ($\mathbf{M} = \mathbf{L} + \mathbf{D}$ et $\mathbf{N} = -\mathbf{U}$)

Méthode de Gauss-Seidel par points

```

 $\mathbf{x}^{(0)} \in \mathbb{R}^n$ 
for  $k = 0, 1, \dots$  do
  |  $(\mathbf{D} + \mathbf{L})\mathbf{x}^{(k+1)} = \mathbf{b} - \mathbf{U}\mathbf{x}^{(k)}$ 
  |  $\Leftrightarrow \mathbf{D}\mathbf{x}^{(k+1)} = \mathbf{b} - \mathbf{L}\mathbf{x}^{(k+1)} - \mathbf{U}\mathbf{x}^{(k)}$ 
end
    
```

Méthode de Gauss-Seidel par points (réécriture)

```

 $x_i^{(0)} \in \mathbb{R}$  pour  $i = 1 \dots n$ 
for  $k = 0, 1, \dots$  do
  | for  $i = 1 \dots n$  do
  | |  $x_i^{(k+1)} = a_{ii}^{-1} (b_i - \sum_{j < i} a_{ij} x_j^{(k+1)} - \sum_{j > i} a_{ij} x_j^{(k)})$ 
  | end
end
    
```

Discussion

- Les itérations de la boucle intérieure sont dépendantes. À chaque itération, on utilise la dernière information disponible pour mettre à jour la solution.
- À chaque itération k , résol. d'un système triangulaire inférieure (Descente en //)

Méthodes de Jacobi/Gauss-Seidel par blocs

Méthodes stationnaires par blocs (réécriture)

```

x_I^{(0)} \in \mathbb{R}^{n_I} \text{ pour } I = 1 \dots n^{\text{blk}}
for k = 0, 1, \dots \text{ do
  for I = 1 \dots n^{\text{blk}} \text{ do
    A_{II} x_I^{(k+1)} = b_I - \sum_{I \neq J} A_{IJ} x_J^{(k)}           Jacobi par blocs
    A_{II} x_I^{(k+1)} = b_I - \sum_{J < I} A_{IJ} x_J^{(k+1)} - \sum_{I < J} A_{IJ} x_J^{(k)}   G.-S. par blocs
  end
end
end
    
```

Discussion

- Boucle intérieure sur les n^{blk} blocs de $\mathbf{x}^{(k)}$, plus sur les n élém.
- Jacobi : Produit matrice-vecteur avec matrice pleine ($\mathbf{L}^{\text{blk}} + \mathbf{U}^{\text{blk}}$) (BLAS en //)
- Jacobi : Résolution d'un système bloc diagonal (Réso. des blocs en //)
- G.-S. : Résolution d'un système bloc triangulaire inférieure (Descente en //)

Méthodes itératives pour les systèmes linéaires

Méthodes stationnaires

Système issu d'une discrétisation par différences finies

Schéma de différence finies — Description [1/3]

Définition du problème

Le champ $u(x, y)$ gouverné par l'équation de Poisson sur un domaine carré :

$$\begin{cases} \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f(x, y), & \text{pour } (x, y) \in \Omega =]a, b[\times]a, b[, \\ u = 0, & \text{pour } (x, y) \in \partial\Omega. \end{cases}$$

Discrétisation et schéma numérique

Le problème est discrétisé sur une grille régulière :

- Points de discrétisation : $(x_i, y_j) = (a + ih, a + jh)$ ($i, j = 0, \dots, n+1$)
- Pas spatial : $h = (b - a) / (n + 1)$
- Approximation du champ : $u_{i,j} \approx u(x_i, y_j)$

On considère le schéma à 5 points :

$$\frac{1}{h^2} (u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j}) = f_{i,j} \quad (i, j = 1, \dots, n)$$

avec $u_{i,j} = 0$ (i et/ou $j \in \{0, n+1\}$)

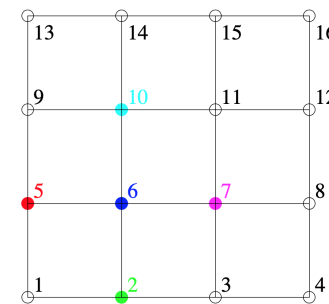
$$f_{i,j} = f(x_i, y_j)$$

Précision : $\mathcal{O}(h^2)$

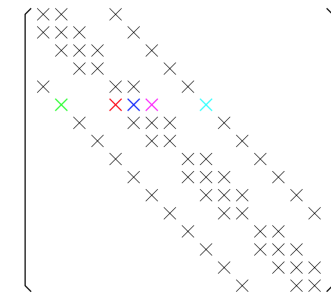
Schéma de différence finies — Description [2/3]

$$\begin{cases} \frac{1}{h^2} (u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j}) = f_{i,j} & (i, j = 1, \dots, n) \\ u_{i,j} = 0 & (i \text{ et/ou } j \in \{0, n+1\}) \end{cases}$$

Représentation matricielle du problème : $\mathbf{A}\mathbf{u} = \mathbf{f}$ $\mathbf{A} \in \mathbb{R}^{n^2 \times n^2}$ $\mathbf{u}, \mathbf{f} \in \mathbb{R}^{n^2}$



Finite difference grid



Matrix of the system

Algorithme parallèle avec Jacobi (partition 1D)

Sur chaque processus p :

$u_{i,j}^{(0)} \in \mathbb{R}$ pour $i = i_{\text{start},p}, \dots, i_{\text{end},p}$ et $j = 1, \dots, n$

for $k = 0, 1, \dots$ do

Phase de communication :

- Si $p > 0$: envoi $u_{i_{\text{start},p},*}$ à processus $p-1$
- Si $p > 0$: reçoit $u_{i_{\text{start},p-1},*}$ de processus $p-1$
- Si $p < (P-1)$: envoi $u_{i_{\text{end},p},*}$ à processus $p+1$
- Si $p < (P-1)$: reçoit $u_{i_{\text{end},p+1},*}$ de processus $p+1$

Phase de calcul :

for $i = i_{\text{start},p}, \dots, i_{\text{end},p}$ do

for $j = 1, \dots, n$ do

$$u_{i,j}^{(k+1)} = \frac{1}{4} \left(u_{i+1,j}^{(k)} + u_{i-1,j}^{(k)} + u_{i,j+1}^{(k)} + u_{i,j-1}^{(k)} \right) - \frac{h^2}{4} f_{i,j}$$

end

end

end

(Dans les communication, * indique que la ligne complète est envoyée.)

Construction du schéma

$$\mathbf{Mx}^{(k+1)} = \mathbf{Nx}^{(k)} + \mathbf{b}$$

avec $\mathbf{A} = (\mathbf{M} - \mathbf{N}) = \frac{1}{h^2}$

Algorithme séquentiel avec Gauss-Seidel

$u_{i,j}^{(0)} \in \mathbb{R}$ pour $i, j = 1 \dots n$

for $k = 0, 1, \dots$ do

for $i = 1, \dots, n$ do

for $j = 1, \dots, n$ do

$$u_{i,j}^{(k+1)} = \frac{1}{4} \left(u_{i+1,j}^{(k)} + u_{i-1,j}^{(k+1)} + u_{i,j+1}^{(k)} + u_{i,j-1}^{(k+1)} \right) - \frac{h^2}{4} f_{i,j}$$

end

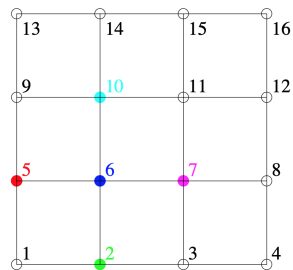
end

end

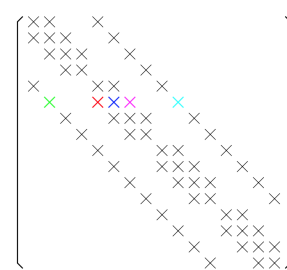
Parallélisation du schéma

La méthode de GS utilise les dernières valeurs disponibles pour la mise à jour.

⇒ Procédure (a priori) essentiellement séquentielle



Finite difference grid

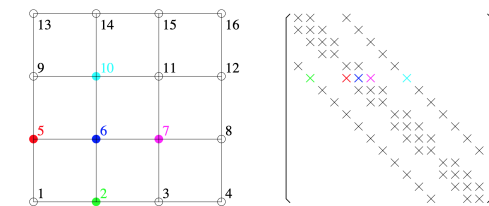


Matrix of the system

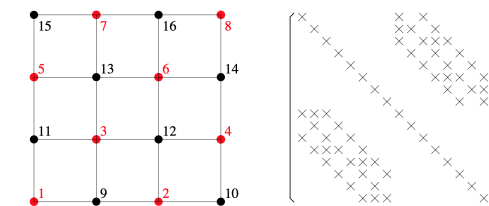
Idee : changer l'ordre d'évaluation des inconnues pour rendre la procédure parallélisable (i.e. permuter les lignes)

Parallélisation du schéma (avec coloriage)

Numérotation naturelle



Numérotation avec coloriage "Rouge-Noir"



Les inconnues associées à une même couleur peuvent être mises à jour en parallèle.

Algorithme séquentiel avec Gauss-Seidel (avec coloriage rouge-noir)

```

 $u_{i,j}^{(0)} \in \mathbb{R}$  pour  $i, j = 1 \dots n$ 
for  $k = 0, 1, \dots$  do
  \ \ Mise à jour des inconnues rouges
  for  $i = 1, \dots, n$  do
    for  $j = 1, \dots, n$  do
      Si  $(i, j)$  rouge :  $u_{i,j}^{(k+1)} = \frac{1}{4} (u_{i+1,j}^{(k)} + u_{i-1,j}^{(k)} + u_{i,j+1}^{(k)} + u_{i,j-1}^{(k)}) - \frac{h^2}{4} f_{i,j}$ 
    end
  end
  \ \ Mise à jour des inconnues noires
  for  $i = 1, \dots, n$  do
    for  $j = 1, \dots, n$  do
      Si  $(i, j)$  noir :  $u_{i,j}^{(k+1)} = \frac{1}{4} (u_{i+1,j}^{(k+1)} + u_{i-1,j}^{(k+1)} + u_{i,j+1}^{(k+1)} + u_{i,j-1}^{(k+1)}) - \frac{h^2}{4} f_{i,j}$ 
    end
  end
end

```

22

Algorithme parallèle avec Gauss-Seidel (avec coloriage rouge-noir)

Sur chaque processus p :

```

 $u_{i,j}^{(0)} \in \mathbb{R}$  pour  $i = i_{\text{start},p}, \dots, i_{\text{end},p}$  et  $j = 1, \dots, n$ 
for  $k = 0, 1, \dots$  do
  Phase de communication (comme pour Jacobi)
  \ \ Mise à jour des inconnues rouges
  for  $i = i_{\text{start},p}, \dots, i_{\text{end},p}$  do
    for  $j = 1, \dots, n$  do
      Si  $(i, j)$  rouge :  $u_{i,j}^{(k+1)} = \frac{1}{4} (u_{i+1,j}^{(k)} + u_{i-1,j}^{(k)} + u_{i,j+1}^{(k)} + u_{i,j-1}^{(k)}) - \frac{h^2}{4} f_{i,j}$ 
    end
  end
  Phase de communication (comme pour Jacobi)
  \ \ Mise à jour des inconnues noires
  for  $i = i_{\text{start},p}, \dots, i_{\text{end},p}$  do
    for  $j = 1, \dots, n$  do
      Si  $(i, j)$  noir :  $u_{i,j}^{(k+1)} = \frac{1}{4} (u_{i+1,j}^{(k+1)} + u_{i-1,j}^{(k+1)} + u_{i,j+1}^{(k+1)} + u_{i,j-1}^{(k+1)}) - \frac{h^2}{4} f_{i,j}$ 
    end
  end
end

```

23

Quelques commentaires sur parallélisation avec coloriage

- ▶ Principe de base :
 - Chaque couleur = Inconnues mises à jour en parallèle.
 - Phase de communication entre chaque couleur.
- ▶ Numérotation différente ...
 - Algorithme différent, mais toujours Gauss-Seidel.
 - Solution numérique différente, mais schéma avec les mêmes propriétés.
- ▶ Questions extensions :
 - Stencil plus étendu → Coloriage avec plus de couleurs.
 - Maillage non-structuré → Algorithmes de coloriage automatique.

24

Quelques ressources

- ▶ *Méthodes Numériques : Algorithmes, analyse et applications*
A. Quarteroni, R. Sacco, F. Saleri (2007), Springer
- ▶ *Calcul scientifique parallèle*
F. Magoulès et F.-X. Roux (2017), Dunod
- ▶ *Calcul scientifique parallèle*
P. Ciarlet et E. Jamelot, polycopié de cours
- ▶ Slides du cours "CME342 - Parallel Methods in Numerical Analysis" de Stanford University (<http://adl.stanford.edu/cme342/>)