

Méthodes multigrilles

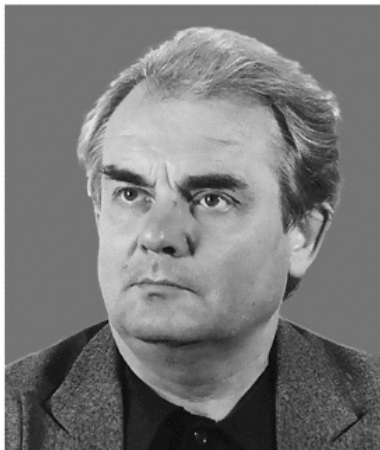
Gérard MEURANT

Novembre 2014

- 1 Introduction
- 2 Multigrille géométrique
- 3 Multigrille algébrique
- 4 Exemple

Multigrille géométrique

- Recherche d'une méthode optimale pour les EDP elliptiques : nombre d'opérations proportionnel à n pour une précision de l'ordre de l'erreur de troncature
- MG a été étudiée en URSS dans les années 60 ([Fedorenko](#) (1962))
- La méthode a reçu peu d'attention jusqu'à ce que [A. Brandt](#) la popularise dans les années 70 en résolvant efficacement des problèmes divers et difficiles
- Livres : [W. Hackbush](#) et [U. Trottenberg](#) (1982), [Hackbush](#) (1985) et (1993), [Briggs](#) (1987), nouvelle édition (2001), [Trottenberg](#) (2001)



Radii Petrovich Fedorenko, 1930–2009

EDP elliptique du 2nd ordre :

$$Lu = f$$

dans $\Omega =]0, 1[^2$ avec c.l. Dirichlet

- Exemple standard : eq. de Poisson $-\Delta u = f$
- maillage régulier avec $h = 1/(m + 1)$, m impair, différences finies à 5 points

$$A_h u_h = b_h$$

- grille grossière avec $H = 2h$

On mélange deux mauvais ingrédients :

- ▶ 1) Méthode itérative “classique” **Jacobi** ou **Gauss-Seidel** qui ont une mauvaise vitesse de convergence,
 $[\rho(M^{-1}A) = 1 - O(h^2)]$
- ▶ 2) Méthode à deux grilles (non convergente) [on ne peut pas réduire certaines fréquences de l'erreur]

Ces deux méthodes sont “complémentaires”

On les combine pour obtenir une méthode efficace

Méthode à 2 grilles

Si on a une approximation u_k de u tq $Au = b$
soit l'erreur $\varepsilon_k = u - u_k$, alors

$$A\varepsilon_k = Au - Au_k = b - Au_k = r_k$$

- On calcule une approximation w_k de ε_k sur la grille grossière Ω_H
- Il faut passer de Ω_h à Ω_H : opérateur de restriction R

$$R : \Omega_h \rightarrow \Omega_H, \quad (\mathbb{R}^{m^2} \rightarrow \mathbb{R}^{p^2})$$

et de Ω_H à Ω_h

$$P : \Omega_H \rightarrow \Omega_h, \quad (\mathbb{R}^{p^2} \rightarrow \mathbb{R}^{m^2})$$

- Comment définir A_H opérateur sur la grille grossière ?

1) On utilise la même approximation que pour A_h mais sur Ω_H

2) $A_H = RA_hP$

- Une étape de la méthode à deux grilles est définie par :

1) $r_k = b - Au_k$

2) $r_k^H = Rr_k$

3) on résout exactement $A_H \varepsilon_k^H = r_k^H$

4) $v_k = P\varepsilon_k^H$

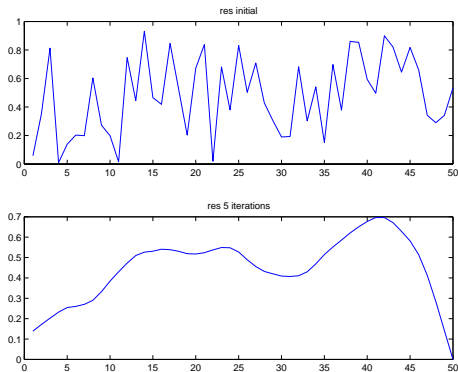
5) $u_{k+1} = v_k + u_k$

Cette méthode ne converge pas (les hautes fréquences ne peuvent pas être représentées sur la grille grossière)

Il faut régulariser le résidu avant d'aller sur Ω_H (lisseur)

Lisseurs

- Ex : Laplacien 1D, Gauss–Seidel



Méthode à 2 grilles + lissage

- 1) Partant de u_k , on fait ν_1 itérations d'un lisseur $S \rightarrow \bar{u}_k$
- 2) $\bar{r}_k = b - A\bar{u}_k$
- 3) $\bar{r}_k^H = R\bar{r}_k$
- 4) on résout exactement $A_H \varepsilon_k^H = \bar{r}_k^H$
- 5) $v_k = P\varepsilon_k^H$
- 6) Partant de $\bar{u}_k + v_k$, on fait ν_2 itérations avec $S \rightarrow u_{k+1}$

On doit définir :

- ▶ le lisseur S
- ▶ les entiers ν_1 et ν_2
- ▶ comment construire la grille grossière
- ▶ l'opérateur de restriction R
- ▶ l'opérateur de prolongement P
- ▶ comment construire A_H

Méthode multigrille

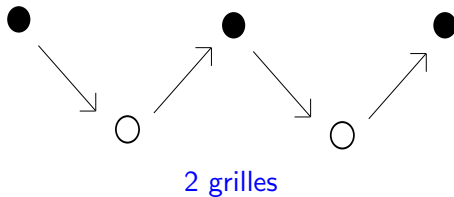
- On résout le problème grossier $A_H \epsilon_k^H = \tilde{r}_k^H$ avec la même méthode récursivement

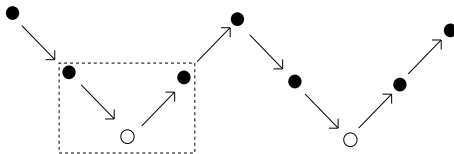
On a une suite de grilles Ω_ℓ de pas h_ℓ , de $\ell = 0$ la plus grossière jusqu'à $\ell = L$ (entier fixé) la plus fine

- A_ℓ approximation de A sur Ω_ℓ
- R_ℓ opérateur de restriction : $\Omega_\ell \rightarrow \Omega_{\ell-1}$
- $P_{\ell-1}$ opérateur de prolongement : $\Omega_{\ell-1} \rightarrow \Omega_\ell$
- S_ℓ lisseur sur Ω_ℓ

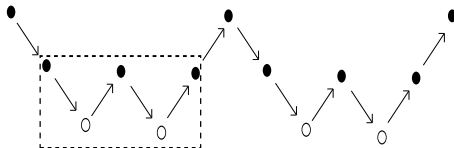
Il existe différentes possibilités d'itérations :

- ↘ restriction
- ↗ prolongement
- lissage
- solution "exacte"





3 grilles, V-cycle



3 grilles, W-cycle

Multigrille algébrique

La méthode multigrille algébrique a été introduite par [J. Ruge](#) et [K. Stuben](#) (1987)

On cherche à imiter MG géométrique

- Une “grille” \equiv (sous) ensemble d’inconnues du graphe de la matrice
- Le problème principal est d’avoir une méthode efficace et parallèle

Préconditionneur multiniveaux

En partant du vecteur nul :

- ▶ 0– sur le niveau le plus grossier, on résout exactement par Gauss (ou autre), sinon
- ▶ 1– On fait ν itérations de lissage
- ▶ 2– On restreint le résidu r à $r_c = Rr$
- ▶ 3– On résout récursivement $A_c e_c = r_c$, $A_c = RAP$, $R = P^T$
- ▶ 4– On prolonge (interpole) e_c vers $e = Pe_c$
- ▶ 5– On ajoute la correction e à l'itéré courant
- ▶ 6– On fait ν itérations de lissage (pour la symétrie si nécessaire)

- Il existe de très nombreuses possibilités de choix :
 - Lisseurs
 - Déraffinement

Lisseurs

- Gauss–Seidel (symétrique)
- Cholesky incomplet
- Inverse approché AINV
- qq itérations d'une méthode de Krylov

Matrice d'influence

Comment définir les grilles grossières?

On utilise le graphe de la matrice. Pour un niveau de grille donnée

$$\mathcal{N} = \{1, \dots, n\}, \quad \mathcal{N} = F \cup C$$

F noeuds “fins”, C noeuds “grossiers”

- La grille grossière suivante est constituée des noeuds C
- Lors du prolongement, les valeurs aux noeuds F seront calculées par interpolation à partir des valeurs aux noeuds C venant de la grille grossière
- On construit une matrice S représentant l'influence réciproque des noeuds

Ensemble des indices de la ligne i de S : choix AMG standard
(inspiré de Ruge-Stuben)

$$S_i^A = \{j \neq i \mid |a_{i,j}| > \tau \max_k |a_{i,k}|, \quad \tau < 1\}$$

τ paramètre fixé

Le noeud i dépend du noeud j (j influence i)

On construit S ($S_{i,j} = 1$ si $j \in S_i$, 0 sinon)

Algorithme de déraffinement

- Connaissant S on veut construire F et C pour avoir le moins de noeuds C possibles mais une “bonne” interpolation
- Basé sur deux principes :
 - 1) $\forall i \in F$, chaque noeud $j \in S_i$ doit être dans C ou doit dépendre d'au moins un noeud de C_i ensemble des noeuds grossiers utilisés pour l'interpolation au noeud i
 - 2) C doit être (autant que faire se peut) un sous-ensemble maximal tel que aucun noeud de C ne dépende d'un autre noeud de C

Algorithme "standard" (variation de Ruge-Stuben)

Poids w_i = nb de noeuds qui dépendent de i (d'après S)

- ▶ 1- On choisit un noeud i de poids maximal comme noeud C
- ▶ 2- On désigne les noeuds que i influence (d'après S) comme noeuds F
- ▶ 3- On incrémente de 1 les poids des noeuds influençant ces nouveaux noeuds F (pour leur donner plus de chance d'être choisis comme noeuds C dans la suite)
- ▶ 4- On décrémente de 1 les poids des noeuds qui dépendent de i
- ▶ On répète les étapes 1-4 jusqu'à ce que tous les noeuds soient choisis

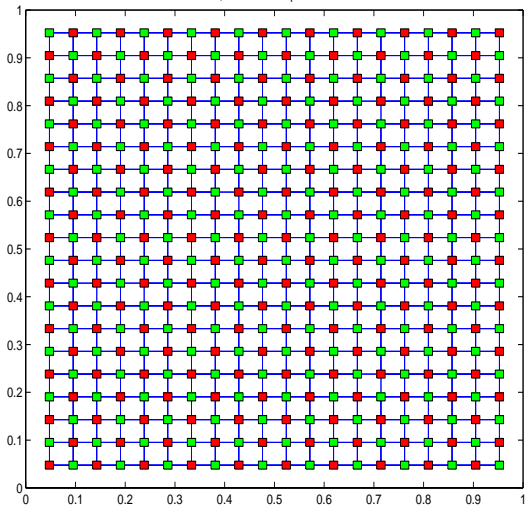
Exemple : équation de Poisson dans un carré, différences finies à 5 points

Le graphe de A est identique au maillage (moins les points du bord)

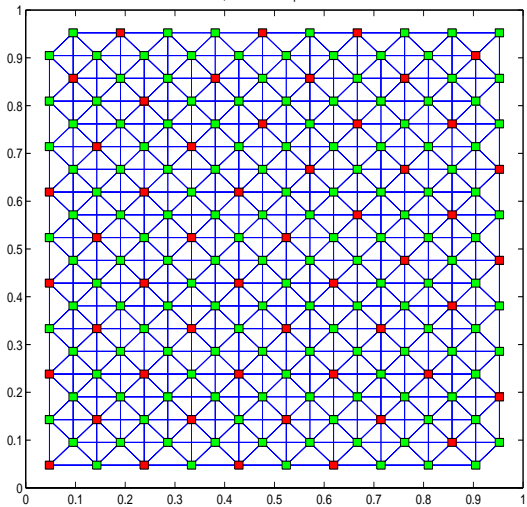
La matrice est normalisée avec des 1 sur la diagonale ($\tau = 0.1$)

Noeuds C et F + graphe de A_ℓ pour les différents niveaux

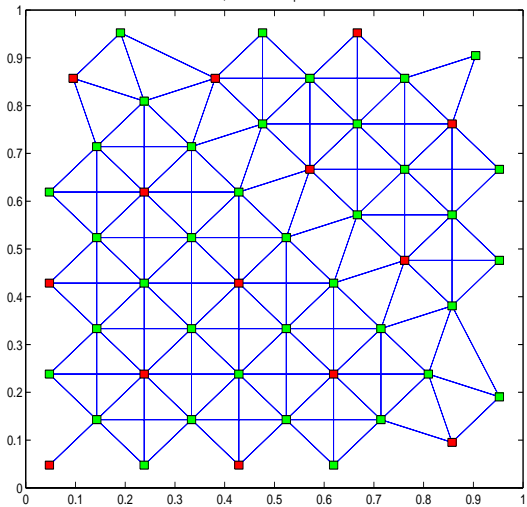
$l = 1$, nb of coarse points = 200



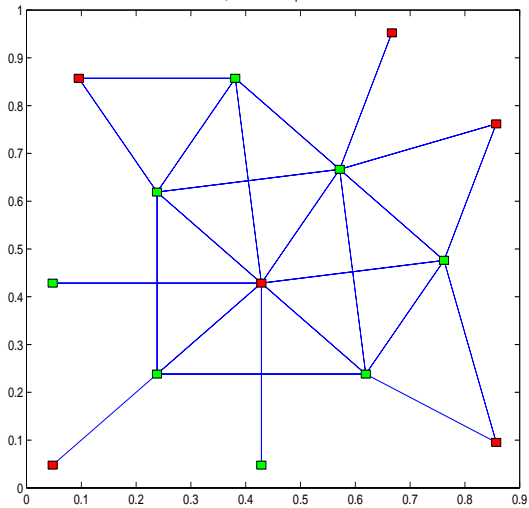
$l = 2$, nb of coarse points = 51



$l = 3$, nb of coarse points = 14



$l = 4$, nb of coarse points = 6



Il existe de nombreux autres algorithmes de déraffinement, par exemple les techniques d'agrégation

Considérons un exemple pour terminer

Nos7 (Matrix Market), $n=729$, $\text{nnz}=4617$, $\kappa(A) = 2.37 \cdot 10^9$,
 $\text{epss}=10^{-6}$

méthode	nb it	$\ r_k\ $	$\ \varepsilon_k\ $
CG, M=I	3776	$1.0649 \cdot 10^{-5}$	$1.9434 \cdot 10^{-4}$
CG, M=diag	89	$9.3199 \cdot 10^{-6}$	$6.3311 \cdot 10^{-6}$
CG, M=IC(0)	23	$2.2085 \cdot 10^{-5}$	0.0018
CG, M=SSOR $\omega = 1$	38	$3.4097 \cdot 10^{-6}$	$2.7747 \cdot 10^{-6}$
CG, M=AINV $\tau = 0.05$	27	$3.4963 \cdot 10^{-5}$	0.0011
CG, M=SAINV $\tau = 0.05$	26	$2.1718 \cdot 10^{-5}$	0.0036
CG, M=ml 'gs,b,st,st' $\tau = 0.05$	8	$1.6675 \cdot 10^{-5}$	0.0042
CG, M=ml 'ic,b,st,st' $\tau = 0.05$	7	$5.1063 \cdot 10^{-7}$	$7.8618 \cdot 10^{-4}$