Adaptive Aitken-Schwarz DDM for large scale computing

D.Tromeur-Dervout[1]

[1] CDCSP/ICJ UMR5208, University Lyon 1, 15 Bd Latarjet, 69622 Villeurbanne, France.

people involved :
F. Oudin(Aitken-Schwarz), P. Linel(mixed FE),
S. Pomarede(Aitken-Schwarz & grid computing)
D. Fogliani (Computer/library maintenance),
? ? ? (phd ANR MICAS : Aitken with Mortar)

Réunion de démarrage : ANR Caclul Intensif et Simulation 2007 MICAS
INRIA Rennes 31 Janvier 2008

Outline

1. The Dirichlet-Neumann Map
2. The Generalized Schwarz Alternating Method
3. The Aitken-Schwarz Method
4. Non separable operator , non regular mesh, adaptive Aitken-Schwarz

Outline

Trace theorem : γ_0

Let $\Omega \subset \mathbf{R}^n$ a bounded domain with Lipschitz boundary $\Gamma := \partial\Omega$. $\forall u \in H^1(\Omega),\ \exists \gamma_0 u \in H^{1/2}(\Gamma)$ the trace satisfying

$$||\gamma_0 u||_{H^{1/2}(\Gamma)} \leq c_T.||u||_{H^1(\Omega)}. \tag{1}$$

Bounded extension : ϵ

vice versa, $\forall u \in H^{1/2}(\Gamma),\ \exists \varepsilon u \in H^1(\Omega)$ a bounded extension satisfying $\gamma_0 \varepsilon u = u$ and

$$||\varepsilon u||_{H^1(\Omega)} \leq c_{IT}.||u||_{H^{1/2}(\Gamma)}. \tag{2}$$

Trace theorem : γ_0

Let $\Omega \subset \mathbf{R}^n$ a bounded domain with Lipschitz boundary $\Gamma := \partial\Omega$. $\forall u \in H^1(\Omega),\ \exists \gamma_0 u \in H^{1/2}(\Gamma)$ the trace satisfying

$$||\gamma_0 u||_{H^{1/2}(\Gamma)} \leq c_T.||u||_{H^1(\Omega)}. \quad (1)$$

Bounded extension : ϵ

vice versa, $\forall u \in H^{1/2}(\Gamma),\ \exists \varepsilon u \in H^1(\Omega)$ a bounded extension satisfying $\gamma_0 \varepsilon u = u$ and

$$||\varepsilon u||_{H^1(\Omega)} \leq c_{IT}.||u||_{H^{1/2}(\Gamma)}. \quad (2)$$

$$\text{Set } L(x)u(x) = -\Sigma_{i,j=1}^{n} \frac{\partial}{\partial x_j}[a_{ji}(x)\frac{\partial}{\partial x_i}u(x)],\ a_{ji} \in L_\infty(\Omega) \quad (3)$$

$L(.)$ is assumed to be uniformly elliptic,

$$\Sigma_{i,j=1}^{n} a_{ji}(x)\xi_j\xi_l \geq c_0.|\xi|^2, \forall \xi \in \mathbf{R}^n, \forall x \in \Omega$$

The conormal derivative γ_1 is given by

$$\gamma_1 u(x) := \Sigma_{i,j=1}^{n} n_j(x)[a_{ji}(x)\frac{\partial}{\partial x_i}u(x)],\ \forall x \in \Gamma$$

where $n(x)$ is the exterior unit normal vector.

$$\begin{aligned} a(u,v) &= \sum_{i,j=1}^{n} \int_\Omega \frac{\partial}{\partial x_j}v(x)a_{ji}(x)\frac{\partial}{\partial x_i}u(x) \\ &= \int_\Omega Lu(x)v(x)dx + \int_\Gamma \gamma_1 u(x)\gamma_0 v(x)dS_x \end{aligned}$$

$$\text{Set } L(x)u(x) = -\Sigma_{i,j=1}^{n} \frac{\partial}{\partial x_j}[a_{ji}(x)\frac{\partial}{\partial x_i}u(x)],\ a_{ji} \in L_\infty(\Omega) \tag{3}$$

$L(.)$ is assumed to be uniformly elliptic,

$$\Sigma_{i,j=1}^{n} a_{ji}(x)\xi_j\xi_l \geq c_0.|\xi|^2, \forall \xi \in \mathbf{R}^n, \forall x \in \Omega$$

The conormal derivative γ_1 is given by

$$\gamma_1 u(x) := \Sigma_{i,j=1}^{n} n_j(x)[a_{ji}(x)\frac{\partial}{\partial x_i}u(x)],\ \forall x \in \Gamma$$

where $n(x)$ is the exterior unit normal vector.

$$\begin{aligned} a(u,v) &= \sum_{i,j=1}^{n} \int_\Omega \frac{\partial}{\partial x_j} v(x) a_{ji}(x) \frac{\partial}{\partial x_i} u(x) \\ &= \int_\Omega Lu(x)v(x)dx + \int_\Gamma \gamma_1 u(x)\gamma_0 v(x) dS_x \end{aligned}$$

$$\text{Set } L(x)u(x) = -\Sigma_{i,j=1}^{n} \frac{\partial}{\partial x_j}[a_{ji}(x)\frac{\partial}{\partial x_i}u(x)],\ a_{ji} \in L_\infty(\Omega) \quad (3)$$

$L(.)$ is assumed to be uniformly elliptic,

$$\Sigma_{i,j=1}^{n} a_{ji}(x)\xi_j\xi_l \geq c_0.|\xi|^2, \forall \xi \in \mathbf{R}^n, \forall x \in \Omega$$

The conormal derivative γ_1 is given by

$$\gamma_1 u(x) := \Sigma_{i,j=1}^{n} n_j(x)[a_{ji}(x)\frac{\partial}{\partial x_i}u(x)],\ \forall x \in \Gamma$$

where $n(x)$ is the exterior unit normal vector.

$$\begin{aligned} a(u,v) &= \sum_{i,j=1}^{n} \int_\Omega \frac{\partial}{\partial x_j}v(x)a_{ji}(x)\frac{\partial}{\partial x_i}u(x) \\ &= \int_\Omega Lu(x)v(x)dx + \int_\Gamma \gamma_1 u(x)\gamma_0 v(x)dS_x \end{aligned}$$

Necas Lemma $\Rightarrow \exists! u \in H^1(\Omega)$ solution of Dirichlet Pb :

$$L(x)u(x) = f(x), \text{for } x \in \Omega, \gamma_0 u(x) = g(x) \text{ for } x \in \Gamma \quad (4)$$

Then setting :

$$l(w) = a(u, \varepsilon w) - \int_\Omega f(x)\varepsilon w(c)dx \; \forall w \in H^{1/2}(\Gamma).$$

Riez thm : $\exists \lambda \in H^{-1/2}(\Gamma)$: $\langle \lambda, w \rangle_{L_2(\Gamma)} = l(w) \; \forall w \in H^{1/2}(\Gamma)$.

Hence, the conormal derivative $\lambda \in H^{-1/2}(\Gamma)$ satisfies

$$\int_\Gamma \lambda(x)w(x)ds_x = a(u_0 + \varepsilon g, \varepsilon w) - \int_\Omega f(x)\varepsilon w(x)dx \; \forall w \in H^{1/2}(\Gamma)$$

$\Rightarrow f$ fixed, we have a DtoN map : $g = \gamma_0 u \mapsto \lambda := \gamma_1 u$

$$\gamma_1 u(x) = Sg(x) - Nf(x), \forall w \in \Gamma \quad (5)$$

Necas Lemma $\Rightarrow \exists! u \in H^1(\Omega)$ solution of Dirichlet Pb :

$$L(x)u(x) = f(x), \text{for } x \in \Omega, \gamma_0 u(x) = g(x) \text{ for } x \in \Gamma \quad (4)$$

Then setting :

$$l(w) = a(u, \varepsilon w) - \int_\Omega f(x)\varepsilon w(c)dx \ \forall w \in H^{1/2}(\Gamma).$$

Riez thm : $\exists \lambda \in H^{-1/2}(\Gamma) : \ \langle \lambda, w \rangle_{L_2(\Gamma)} = l(w) \ \forall w \in H^{1/2}(\Gamma).$

Hence, the conormal derivative $\lambda \in H^{-1/2}(\Gamma)$ satisfies

$$\int_\Gamma \lambda(x)w(x)ds_x = a(u_0 + \varepsilon g, \varepsilon w) - \int_\Omega f(x)\varepsilon w(x)dx \ \forall w \in H^{1/2}(\Gamma)$$

$\Rightarrow f$ fixed, we have a DtoN map : $g = \gamma_0 u \mapsto \lambda := \gamma_1 u$

$$\gamma_1 u(x) = Sg(x) - Nf(x), \forall w \in \Gamma \quad (5)$$

Necas Lemma $\Rightarrow \exists! u \in H^1(\Omega)$ solution of Dirichlet Pb :

$$L(x)u(x) = f(x), \text{for } x \in \Omega, \gamma_0 u(x) = g(x) \text{ for } x \in \Gamma \quad (4)$$

Then setting :

$$l(w) = a(u, \varepsilon w) - \int_\Omega f(x)\varepsilon w(c)dx \ \forall w \in H^{1/2}(\Gamma).$$

Riez thm : $\exists \lambda \in H^{-1/2}(\Gamma) : \ \langle \lambda, w \rangle_{L_2(\Gamma)} = l(w) \ \forall w \in H^{1/2}(\Gamma).$

Hence, the conormal derivative $\lambda \in H^{-1/2}(\Gamma)$ satisfies

$$\int_\Gamma \lambda(x)w(x)ds_x = a(u_0 + \varepsilon g, \varepsilon w) - \int_\Omega f(x)\varepsilon w(x)dx \ \forall w \in H^{1/2}($$

$\Rightarrow f$ fixed, we have a DtoN map : $g = \gamma_0 u \mapsto \lambda := \gamma_1 u$

$$\gamma_1 u(x) = Sg(x) - Nf(x), \forall w \in \Gamma \quad (5)$$

Necas Lemma $\Rightarrow \exists ! u \in H^1(\Omega)$ solution of Dirichlet Pb :

$$L(x)u(x) = f(x), \text{for } x \in \Omega, \gamma_0 u(x) = g(x) \text{ for } x \in \Gamma \quad (4)$$

Then setting :

$$l(w) = a(u, \varepsilon w) - \int_\Omega f(x) \varepsilon w(c) dx \; \forall w \in H^{1/2}(\Gamma).$$

Riez thm : $\exists \lambda \in H^{-1/2}(\Gamma) : \langle \lambda, w \rangle_{L_2(\Gamma)} = l(w) \; \forall w \in H^{1/2}(\Gamma).$

Hence, the conormal derivative $\lambda \in H^{-1/2}(\Gamma)$ satisfies

$$\int_\Gamma \lambda(x) w(x) ds_x = a(u_0 + \varepsilon g, \varepsilon w) - \int_\Omega f(x) \varepsilon w(x) dx \; \forall w \in H^{1/2}($$

$\Rightarrow$ f fixed, we have a DtoN map : $g = \gamma_0 u \mapsto \lambda := \gamma_1 u$

$$\gamma_1 u(x) = Sg(x) - Nf(x), \forall w \in \Gamma \quad (5)$$

eq(4) (with Ω_1 and Ω_2, $\Gamma_{12} = \bar{\Omega}_1 \cap \bar{\Omega}_2 \backslash \partial\Omega$) leads to :

$$Ax = \begin{pmatrix} A_{11} & 0 & A_{13} \\ 0 & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33}^{(1)} + A_{33}^{(2)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ f_3^{(1)} + f_3^{(2)} \end{pmatrix}$$

$$x_1 = A_{11}^{-1}(f_1 - A_{13}x_3), \; x_2 = A_{22}^{-1}(f_2 - A_{23}x_3),$$

setting :

$$S_i = A_{33}^{(i)} - A_{3i}A_{ii}^{-1}A_{i3}, g_i = f_3^{(i)} - A_{3i}A_{ii}^{-1}f_i$$

we obtain the interface Schur complement system

$$Sx_3 = (S_1 + S_2)x_3 = g_1 + g_2 = g \qquad (6)$$

Note that the following identity holds : (R. Natajaran, SIAM J. Sci. Comput., 18(4) :1187-1199,1997)

$$\begin{pmatrix} A_{11} & A_{13} \\ A_{31} & A_{33}^{(1)} \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ S_1 x_3 \end{pmatrix} = \begin{pmatrix} A_{11} & A_{13} \\ 0 & I \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ x_3 \end{pmatrix} \qquad (7)$$

Conormal derivative is not the optimal interface condition in Schwarz

The Generalized Schwarz Alternating Method (GSAM)

B. Engquist and H.-K. Zhao, Appl. Numer. Math. **27** (1998), no. 4, 341–365.

Consider $\Omega = \Omega_1 \cup \Omega_2$ with the two artificial boundaries Γ_1, Γ_2 intersecting $\partial\Omega$.

Algorithm

$$
\begin{aligned}
L(x)u_1^{2n+1}(x) &= f(x),\ \forall x \in \Omega_1,\ \ u_1^{2n+1}(x) = g(x),\ \forall x \in \partial\Omega_1 \backslash \Gamma_1, \\
\Lambda_1 u_1^{2n+1} &+ \lambda_1 \frac{\partial u_1^{2n+1}(x)}{\partial n_1} = \Lambda_1 u_2^{2n} + \lambda_1 \frac{\partial u_2^{2n}(x)}{\partial n_1},\ \forall x \in \Gamma_1 \\
L(x)u_2^{2n+2}(x) &= f(x),\ \forall x \in \Omega_2,\ \ u_2^{2n+2}(x) = g(x),\ \forall x \in \partial\Omega_2 \backslash \Gamma_2, \\
\Lambda_2 u_2^{2n+2} &+ \lambda_2 \frac{\partial u_2^{2n+2}(x)}{\partial n_2} = \Lambda_2 u_1^{2n+1} + \lambda_2 \frac{\partial u_1^{2n+1}(x)}{\partial n_2},\ \forall x \in \Gamma_2.
\end{aligned}
$$

where Λ_i's are some operators and λ_i's are constants.

If $\lambda_1 = 1$ and Λ_1 is the DtoN operator at Γ_1 associated to the homogeneous PDE in Ω_2 with homogeneous boundary condition on $\partial\Omega_2 \cap \partial\Omega$ then GSAM converge in two steps.
proof Let $e_i^n = u - u^n, i = 1, 2,$, then

$$\begin{aligned} L(x)e_1^1(x) &= 0, \ \forall x \in \Omega_1, \ e_1^1(x) = 0, \ \forall x \in \partial\Omega_1 \backslash \Gamma_1, \\ \Lambda_1 e_1^1 &+ \frac{\partial e_1^1(x)}{\partial n_1} = \Lambda_1 e_2^0 + \frac{\partial e_2^0(x)}{\partial n_1}, \ \forall x \in \Gamma_1 \end{aligned}$$

since Λ_1 is the DtoN operator at Γ_1 in Ω_2

$$\frac{\partial e_2^0}{\partial n_1} + \Lambda_1 e_2^0 = -\frac{\partial e_2^0}{\partial n_2} + \frac{\partial e_2^0}{\partial n_2} = 0, \ \Rightarrow e_1^1 = 0 \text{in } \Omega_1$$

Hence we get the exact solution in two steps []

Let $\Omega = \Omega_1 \cup \Omega_2,\ \Omega_{12} = \Omega_1 \cap \Omega_2,\ \Omega_{ii} = \Omega_i \backslash \Omega_{12}$
$e_i^n = u - u_i^n$ in Ω_i satisfies :

$$
\begin{aligned}
(\Lambda_1 + \lambda_1 S_1) R_1 e_1^{2n+1} &= (\Lambda_1 - \lambda_1 S_{22}) R_{22} P_2 e_2^{2n} \\
(\Lambda_2 + \lambda_2 S_2) R_2 e_2^{2n+2} &= (\Lambda_2 - \lambda_2 S_{22}) R_{11} P_1 e_1^{2n+1}
\end{aligned}
$$

with

- $P_i : H^1(\Omega_i) \to H^1(\Omega_{ii})$
- S_i (S_{ii}) the DtoN map operator in Ω_i (Ω_{ii}) on Γ_i ($\Gamma_{mod(i,2)+1}$).
- $R_i : H^1(\Omega_i) \to H^{1/2}(\Gamma_i)$, $R_{ii} : H^1(\Omega_{ii}) \to H^{1/2}(\Gamma_{mod(i,2)+1})$,
- $R_i^* \ : \ R_i R_i^* = I$,
  $\forall g \in H^{1/2}(\Gamma_i),\ L(x) R_i^* g = 0, R_i^* g = g\ on \Gamma_i, R_i^* g =$
  $0\ on\ \partial\Omega_i \backslash \Gamma_i$

Thus the convergence of GSAM is purely linear ! ! Aitken-Schwarz DDM uses this property to accelerate the convergence :

- Pb : Λ_i DtoN operators are global operators (linking all the subdomains when > 3).
- In practice, the algebraical approximations of this operators are used (see Nataf, Gander).
- On the other hand, the convergence property of the Schwarz Alternating methodology is used to define the Aitken-Schwarz methodology.
- Consequently, no direct approximation of the DtoN map is used, but an approximation of the operator of error linked to this DtoN map is performed.

- Pb : Λ_i DtoN operators are global operators (linking all the subdomains when > 3).
- In practice, the algebraical approximations of this operators are used (see Nataf, Gander).
- On the other hand, the convergence property of the Schwarz Alternating methodology is used to define the Aitken-Schwarz methodology.
- Consequently, no direct approximation of the DtoN map is used, but an approximation of the operator of error linked to this DtoN map is performed.

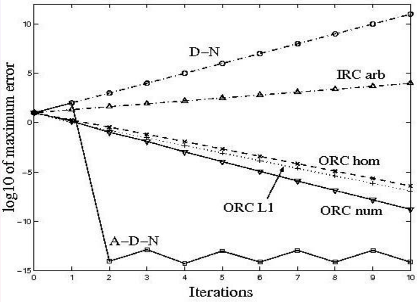
- Pb : Λ_i DtoN operators are global operators (linking all the subdomains when > 3).
- In practice, the algebraical approximations of this operators are used (see Nataf, Gander).
- On the other hand, the convergence property of the Schwarz Alternating methodology is used to define the Aitken-Schwarz methodology.
- Consequently, no direct approximation of the DtoN map is used, but an approximation of the operator of error linked to this DtoN map is performed.

- Pb : Λ_i DtoN operators are global operators (linking all the subdomains when > 3).
- In practice, the algebraical approximations of this operators are used (see Nataf, Gander).
- On the other hand, the convergence property of the Schwarz Alternating methodology is used to define the Aitken-Schwarz methodology.
- Consequently, no direct approximation of the DtoN map is used, but an approximation of the operator of error linked to this DtoN map is performed.

Exemple of linear convergence

D. Calugaru & D.Tromeur-Dervout, LNCSE, 40 :529-536,2004

$$-k_1 \Delta u_1 = f \text{ in } \Omega_1, \ -k_2 \Delta u_2 = f \text{ in } \Omega_2, \ u_1|_\Gamma = u_2|_\Gamma, \ k_1 \frac{\partial u_1}{\partial n_1}|_\Gamma = -k_2 \frac{\partial u_2}{\partial n_2}$$



$$\rho(\xi) = \left| \frac{\alpha_1 - \beta_1 k_2 |\xi|}{\alpha_1 + \beta_1 k_1 |\xi|} \cdot \frac{\alpha_2 - \beta_2 k_1 |\xi|}{\alpha_2 + \beta_2 k_2 |\xi|} \right| .$$

- IRCarb : $\alpha_i = k_i$
- ORChom : $\alpha_1 = 3, \alpha_2 = 30$
  $\alpha_{1,opt} = k_2 \sqrt{\xi_{min} \xi_{max}}, \ \alpha_{2,opt} = k_1 \sqrt{\xi_m}$
- ORCL1 : $\alpha \simeq 7$ optimizing the convergence

M. J. Gander, F. Magoulès, and F. Nataf,

SIAM J. Sci. Comput., 24(1) :38-60, 2002

$f(x, y) = 2k_1 k_2 \sin x \sin y$, and Dirichlet B.C. $u = 10$

$$\begin{cases} -k_1[(\hat{u}_1^{n+1})''_{xx}(x,\xi) - \xi^2 \hat{u}_1^{n+1}(x,\xi)] = \hat{f}(x,\xi), & \text{in } (-\infty, 0) \times \mathbb{R} \\ \alpha_1 \hat{u}_1^{n+1}(0,\xi) + \beta_1 k_1 (\hat{u}_1^{n+1})'_x(0,\xi) = \alpha_1 \hat{u}_2^n(0,\xi) + \beta_1 k_2 (\hat{u}_2^n)'_x(0,\xi), & \xi \in \mathbb{R} \end{cases}$$

$$\begin{cases} -k_2[(\hat{u}_2^{n+1})''_{xx}(x,\xi) - \xi^2 \hat{u}_2^{n+1}(x,\xi)] = \hat{f}(x,\xi), & \text{in } (0, \infty) \times \mathbb{R} \\ \alpha_2 \hat{u}_2^{n+1}(0,\xi) - \beta_2 k_2 (\hat{u}_2^{n+1})'_x(0,\xi) = \alpha_2 \hat{u}_1^{n+1}(0,\xi) - \beta_2 k_1 (\hat{u}_1^{n+1})'_x(0,\xi), & \xi \in \mathbb{R} \end{cases}$$

1. The Dirichlet-Neumann Map
2. The Generalized Schwarz Alternating Method
3. The Aitken-Schwarz Method
4. Non separable operator , non regular mesh, adaptive Aitken-Schwarz

Acceleration of Schwarz Method for Elliptic Problems

M.Garbey and D.Tromeur-Dervout : *On some Aitken like acceleration of the Schwarz method*,

Int. J. for Numerical Methods in Fluids, 40(12) :1493-1513,2002

- additive Schwarz algorithm :
  - $L[u_1^{n+1}] = f \ in \ \Omega_1, \ u_{1|\Gamma_1}^{n+1} = u_{2|\Gamma_1}^n,$
  - $L[u_2^{n+1}] = f \ in \ \Omega_2, \ u_{2|\Gamma_2}^{n+1} = u_{1|\Gamma_2}^n.$
- the interface error operator T is **linear**, i.e
  - $u_{1|\Gamma_2}^{n+1} - U_{|\Gamma_2} = \delta_1(u_{2|\Gamma_1}^n - U_{|\Gamma_1}),$
  - $u_{2|\Gamma_1}^{n+1} - U_{|\Gamma_1} = \delta_2(u_{1|\Gamma_2}^n - U_{|\Gamma_2}).$
- Consequently
  - $u_{1|\Gamma_2}^2 - u_{1|\Gamma_2}^1 = \delta_1(u_{2|\Gamma_1}^1 - u_{2|\Gamma_1}^0),$
  - $u_{2|\Gamma_1}^2 - u_{2|\Gamma_1}^1 = \delta_2(u_{1|\Gamma_2}^1 - u_{1|\Gamma_2}^0),$
- Computation of $\delta_{1/2}$:
  $L[v_{1/2}] = 0 \ in \ \Omega_{1/2}, \ v_{\Gamma_{1/2}} = 1.$ thus $\delta_{1/2} = v_{\Gamma_{2/1}}.$
- iff $\delta \neq 1$ **Aitken-Schwarz** gives the solution with exactly **3** iterations and possibly **2** in the analytical case.

M.Garbey and D.Tromeur-Dervout : *On some Aitken like acceleration of the Schwarz method*,

Int. J. for Numerical Methods in Fluids, 40(12) :1493-1513,2002

- additive Schwarz algorithm :
  - $L[u_1^{n+1}] = f \; in \; \Omega_1, \; u_{1|\Gamma_1}^{n+1} = u_{2|\Gamma_1}^n,$
  - $L[u_2^{n+1}] = f \; in \; \Omega_2, \; u_{2|\Gamma_2}^{n+1} = u_{1|\Gamma_2}^n.$
- the interface error operator T is **linear**, i.e
  - $u_{1|\Gamma_2}^{n+1} - U_{|\Gamma_2} = \delta_1(u_{2|\Gamma_1}^n - U_{|\Gamma_1}),$
  - $u_{2|\Gamma_1}^{n+1} - U_{|\Gamma_1} = \delta_2(u_{1|\Gamma_2}^n - U_{|\Gamma_2}).$
- Consequently
  - $u_{1|\Gamma_2}^2 - u_{1|\Gamma_2}^1 = \delta_1(u_{2|\Gamma_1}^1 - u_{2|\Gamma_1}^0),$
  - $u_{2|\Gamma_1}^2 - u_{2|\Gamma_1}^1 = \delta_2(u_{1|\Gamma_2}^1 - u_{1|\Gamma_2}^0),$
- Computation of $\delta_{1/2}$:
  $L[v_{1/2}] = 0 \; in \; \Omega_{1/2}, \; v_{\Gamma_{1/2}} = 1$. thus $\delta_{1/2} = v_{\Gamma_{2/1}}$.
- iff $\delta \neq 1$ **Aitken-Schwarz** gives the solution with exactly **3** iterations and possibly **2** in the analytical case.

M.Garbey and D.Tromeur-Dervout : *On some Aitken like acceleration of the Schwarz method*,
Int. J. for Numerical Methods in Fluids, 40(12) :1493-1513,2002

- additive Schwarz algorithm :
  - $L[u_1^{n+1}] = f \ in \ \Omega_1, \ u_{1|\Gamma_1}^{n+1} = u_{2|\Gamma_1}^n,$
  - $L[u_2^{n+1}] = f \ in \ \Omega_2, \ u_{2|\Gamma_2}^{n+1} = u_{1|\Gamma_2}^n.$
- the interface error operator T is **linear**, i.e
  - $u_{1|\Gamma_2}^{n+1} - U_{|\Gamma_2} = \delta_1 (u_{2|\Gamma_1}^n - U_{|\Gamma_1}),$
  - $u_{2|\Gamma_1}^{n+1} - U_{|\Gamma_1} = \delta_2 (u_{1|\Gamma_2}^n - U_{|\Gamma_2}).$
- Consequently
  - $u_{1|\Gamma_2}^2 - u_{1|\Gamma_2}^1 = \delta_1 (u_{2|\Gamma_1}^1 - u_{2|\Gamma_1}^0),$
  - $u_{2|\Gamma_1}^2 - u_{2|\Gamma_1}^1 = \delta_2 (u_{1|\Gamma_2}^1 - u_{1|\Gamma_2}^0),$
- Computation of $\delta_{1/2}$:
  $L[v_{1/2}] = 0 \ in \ \Omega_{1/2}, \ v_{\Gamma_{1/2}} = 1.$ thus $\delta_{1/2} = v_{\Gamma_{2/1}}.$
- iff $\delta \neq 1$ **Aitken-Schwarz** gives the solution with exactly **3** iterations and possibly **2** in the analytical case.

Acceleration of Schwarz Method for Elliptic Problems

M.Garbey and D.Tromeur-Dervout : *On some Aitken like acceleration of the Schwarz method*,

Int. J. for Numerical Methods in Fluids, 40(12) :1493-1513,2002

- additive Schwarz algorithm :
  - $L[u_1^{n+1}] = f \ in \ \Omega_1, \ u_{1|\Gamma_1}^{n+1} = u_{2|\Gamma_1}^n,$
  - $L[u_2^{n+1}] = f \ in \ \Omega_2, \ u_{2|\Gamma_2}^{n+1} = u_{1|\Gamma_2}^n.$
- the interface error operator T is **linear**, i.e
  - $u_{1|\Gamma_2}^{n+1} - U_{|\Gamma_2} = \delta_1(u_{2|\Gamma_1}^n - U_{|\Gamma_1}),$
  - $u_{2|\Gamma_1}^{n+1} - U_{|\Gamma_1} = \delta_2(u_{1|\Gamma_2}^n - U_{|\Gamma_2}).$
- Consequently
  - $u_{1|\Gamma_2}^2 - u_{1|\Gamma_2}^1 = \delta_1(u_{2|\Gamma_1}^1 - u_{2|\Gamma_1}^0),$
  - $u_{2|\Gamma_1}^2 - u_{2|\Gamma_1}^1 = \delta_2(u_{1|\Gamma_2}^1 - u_{1|\Gamma_2}^0),$
- Computation of $\delta_{1/2}$:
  $L[v_{1/2}] = 0 \ in \ \Omega_{1/2}, \ v_{\Gamma_{1/2}} = 1.$ thus $\delta_{1/2} = v_{\Gamma_{2/1}}.$
- iff $\delta \neq 1$ **Aitken-Schwarz** gives the solution with exactly **3** iterations and possibly **2** in the analytical case.

Acceleration of Schwarz Method for Elliptic Problems

M.Garbey and D.Tromeur-Dervout : *On some Aitken like acceleration of the Schwarz method*,

Int. J. for Numerical Methods in Fluids, 40(12) :1493-1513,2002

- additive Schwarz algorithm :
  - $L[u_1^{n+1}] = f \ in \ \Omega_1, \ u^{n+1}_{1|\Gamma_1} = u^n_{2|\Gamma_1},$
  - $L[u_2^{n+1}] = f \ in \ \Omega_2, \ u^{n+1}_{2|\Gamma_2} = u^n_{1|\Gamma_2}.$
- the interface error operator T is **linear**, i.e
  - $u^{n+1}_{1|\Gamma_2} - U_{|\Gamma_2} = \delta_1 (u^n_{2|\Gamma_1} - U_{|\Gamma_1}),$
  - $u^{n+1}_{2|\Gamma_1} - U_{|\Gamma_1} = \delta_2 (u^n_{1|\Gamma_2} - U_{|\Gamma_2}).$
- Consequently
  - $u^2_{1|\Gamma_2} - u^1_{1|\Gamma_2} = \delta_1 (u^1_{2|\Gamma_1} - u^0_{2|\Gamma_1}),$
  - $u^2_{2|\Gamma_1} - u^1_{2|\Gamma_1} = \delta_2 (u^1_{1|\Gamma_2} - u^0_{1|\Gamma_2}),$
- Computation of $\delta_{1/2}$:
  $L[v_{1/2}] = 0 \ in \ \Omega_{1/2}, \ v_{\Gamma_{1/2}} = 1.$ thus $\delta_{1/2} = v_{\Gamma_{2/1}}.$
- iff $\delta \neq 1$ **Aitken-Schwarz** gives the solution with exactly **3** iterations and possibly **2** in the analytical case.

Additive Schwarz alg. with : $\Omega = \cup\Omega_i$, $\Omega_{i+1} \cap \Omega_i \neq \emptyset$,

for $i = 1..q$, *do*

$$L[u_i^{n+1}] = f \ in\ \Omega_i,,$$

$$u_i^{n+1}(x_i^l) = u_{i-1}^n(x_i^l),\ u_i^{n+1}(x_i^r) = u_{i+1}^n(x_i^r),$$

enddo

- interfaces : $\tilde{u}^n = (u_2^{l,n}, u_1^{r,n}, u_3^{l,n}, u_2^{r,n}, ..., u_q^{l,n}, u_{q-1}^{r,n})$
- matrix corresponding to iterations for interfaces :

$$\begin{vmatrix} 0 & \delta_1^r & 0 & 0 & .... & & & & \\ \delta_2^{l,l} & 0 & 0 & \delta_2^{l,r} & ... & & & & \\ \delta_2^{r,l} & 0 & 0 & \delta_2^{r,r} & ... & & & & \\ & & & & & & & & \\ & & & ... & \delta_{q-1}^{l,l} & 0 & 0 & \delta_{q-1}^{l,r} \\ & & & ... & \delta_{q-1}^{r,l} & 0 & 0 & \delta_{q-1}^{r,r} \\ & & & ... & 0 & 0 & \delta_q^r & 0 \end{vmatrix}$$

- if $||P|| < 1$,

$$\tilde{u}^\infty = (Id - P)^{-1}(\tilde{u}^{n+1} - P\tilde{u}^n).$$

Additive Schwarz alg. with : $\Omega = \cup \Omega_i,\ \Omega_{i+1} \cap \Omega_i \neq \emptyset$,
for $i = 1..q$, *do*

$$L[u_i^{n+1}] = f \text{ in } \Omega_i,,$$

$$u_i^{n+1}(x_i^l) = u_{i-1}^n(x_i^l),\ \ u_i^{n+1}(x_i^r) = u_{i+1}^n(x_i^r),$$

enddo

- interfaces : $\tilde{u}^n = (u_2^{l,n}, u_1^{r,n}, u_3^{l,n}, u_2^{r,n}, ..., u_q^{l,n}, u_{q-1}^{r,n})$
- matrix corresponding to iterations for interfaces :

$$\begin{vmatrix} 0 & \delta_1^r & 0 & 0 & .... & & & & \\ \delta_2^{l,l} & 0 & 0 & \delta_2^{l,r} & ... & & & & \\ \delta_2^{r,l} & 0 & 0 & \delta_2^{r,r} & ... & & & & \\ & & & & & & & & \\ & & & ... & \delta_{q-1}^{l,l} & 0 & 0 & \delta_{q-1}^{l,r} \\ & & & ... & \delta_{q-1}^{r,l} & 0 & 0 & \delta_{q-1}^{r,r} \\ & & & ... & 0 & 0 & \delta_q^r & 0 \end{vmatrix}$$

- if $||P|| < 1$,

$$\tilde{u}^\infty = (Id - P)^{-1}(\tilde{u}^{n+1} - P\tilde{u}^n).$$

AS Method : One D - Arbitrary number of subdomains

Additive Schwarz alg. with : $\Omega = \cup \Omega_i$, $\Omega_{i+1} \cap \Omega_i \neq \emptyset$,
for $i = 1..q$, *do*

$$L[u_i^{n+1}] = f \ in \ \Omega_i,,$$

$u_i^{n+1}(x_i^l) = u_{i-1}^n(x_i^l), \ u_i^{n+1}(x_i^r) = u_{i+1}^n(x_i^r)$,
enddo

- interfaces : $\tilde{u}^n = (u_2^{l,n}, u_1^{r,n}, u_3^{l,n}, u_2^{r,n}, ..., u_q^{l,n}, u_{q-1}^{r,n})$
- matrix corresponding to iterations for interfaces :

$$\begin{vmatrix} 0 & \delta_1^r & 0 & 0 & .... & & & & \\ \delta_2^{l,l} & 0 & 0 & \delta_2^{l,r} & ... & & & & \\ \delta_2^{r,l} & 0 & 0 & \delta_2^{r,r} & ... & & & & \\ & & & & & & & & \\ & & & ... & \delta_{q-1}^{l,l} & 0 & 0 & \delta_{q-1}^{l,r} \\ & & & ... & \delta_{q-1}^{r,l} & 0 & 0 & \delta_{q-1}^{r,r} \\ & & & ... & 0 & 0 & \delta_q^r & 0 \end{vmatrix}$$

- if $||P|| < 1$,

$$\tilde{u}^\infty = (Id - P)^{-1}(\tilde{u}^{n+1} - P\tilde{u}^n).$$

AS Method : One D - Arbitrary number of subdomains

Additive Schwarz alg. with : $\Omega = \cup\Omega_i,\ \Omega_{i+1} \cap \Omega_i \neq \emptyset$,
for $i = 1..q$, *do*

$$L[u_i^{n+1}] = f \text{ in } \Omega_i,,$$

$$u_i^{n+1}(x_i^l) = u_{i-1}^n(x_i^l),\ u_i^{n+1}(x_i^r) = u_{i+1}^n(x_i^r),$$

enddo

- interfaces : $\tilde{u}^n = (u_2^{l,n}, u_1^{r,n}, u_3^{l,n}, u_2^{r,n}, ..., u_q^{l,n}, u_{q-1}^{r,n})$
- matrix corresponding to iterations for interfaces :

$$\left|\begin{array}{ccccccccc}
0 & \delta_1^r & 0 & 0 & .... & & & & \\
\delta_2^{l,l} & 0 & 0 & \delta_2^{l,r} & ... & & & & \\
\delta_2^{r,l} & 0 & 0 & \delta_2^{r,r} & ... & & & & \\
 & & & & & & & & \\
 & & & ... & \delta_{q-1}^{l,l} & 0 & 0 & \delta_{q-1}^{l,r} & \\
 & & & ... & \delta_{q-1}^{r,l} & 0 & 0 & \delta_{q-1}^{r,r} & \\
 & & & ... & 0 & 0 & \delta_q^r & 0 &
\end{array}\right|$$

- if $||P|| < 1$,

$$\tilde{u}^\infty = (Id - P)^{-1}(\tilde{u}^{n+1} - P\tilde{u}^n).$$

Aitken-Schwarz algorithm : One D - Arbitrary number of subdomains

- step1 : compute in parallel each subblocks $P_i = \begin{pmatrix} \delta_i^{l,l} & \delta_i^{l,r} \\ \delta_i^{r,l} & \delta_i^{r,r} \end{pmatrix}$ from each subproblems.
- step2 : apply **one** additive Schwarz iterate.
- step3 : apply generalized Aitken acceleration on the interfaces.
- step4 : compute in **parallel** the solution for each subdomain.
- Again the overlap can be minimum !

Aitken-Schwarz algorithm : One D - Arbitrary number of subdomains

- step1 : compute in parallel each subblocks $P_i = \begin{pmatrix} \delta_i^{l,l} & \delta_i^{l,r} \\ \delta_i^{r,l} & \delta_i^{r,r} \end{pmatrix}$ from each subproblems.
- step2 : apply **one** additive Schwarz iterate.
- step3 : apply generalized Aitken acceleration on the interfaces.
- step4 : compute in **parallel** the solution for each subdomain.
- Again the overlap can be minimum !

Aitken-Schwarz algorithm : One D - Arbitrary number of subdomains

- step1 : compute in parallel each subblocks $P_i = \begin{pmatrix} \delta_i^{l,l} & \delta_i^{l,r} \\ \delta_i^{r,l} & \delta_i^{r,r} \end{pmatrix}$ from each subproblems.
- step2 : apply **one** additive Schwarz iterate.
- step3 : apply generalized Aitken acceleration on the interfaces.
- step4 : compute in **parallel** the solution for each subdomain.
- Again the overlap can be minimum !

Aitken-Schwarz algorithm : One D - Arbitrary number of subdomains

- step1 : compute in parallel each subblocks $P_i = \begin{pmatrix} \delta_i^{l,l} & \delta_i^{l,r} \\ \delta_i^{r,l} & \delta_i^{r,r} \end{pmatrix}$ from each subproblems.
- step2 : apply **one** additive Schwarz iterate.
- step3 : apply generalized Aitken acceleration on the interfaces.
- step4 : compute in **parallel** the solution for each subdomain.
- Again the overlap can be minimum !

Aitken-Schwarz algorithm : One D - Arbitrary number of subdomains

- step1 : compute in parallel each subblocks $P_i = \begin{pmatrix} \delta_i^{l,l} & \delta_i^{l,r} \\ \delta_i^{r,l} & \delta_i^{r,r} \end{pmatrix}$ from each subproblems.
- step2 : apply **one** additive Schwarz iterate.
- step3 : apply generalized Aitken acceleration on the interfaces.
- step4 : compute in **parallel** the solution for each subdomain.
- Again the overlap can be minimum !

Aitken-Schwarz algorithm : One D - Arbitrary number of subdomains

- step1 : compute in parallel each subblocks $P_i = \begin{pmatrix} \delta_i^{l,l} & \delta_i^{l,r} \\ \delta_i^{r,l} & \delta_i^{r,r} \end{pmatrix}$ from each subproblems.
- step2 : apply **one** additive Schwarz iterate.
- step3 : apply generalized Aitken acceleration on the interfaces.
- step4 : compute in **parallel** the solution for each subdomain.
- Again the overlap can be minimum !

Non overlapping subdomains : $[\alpha, \Gamma_1] \cup [\Gamma_1, \Gamma_2] \cup [\Gamma_2, \beta]$, $\Gamma_1 < \Gamma_2$.
Then the Schwarz algorithm writes :

$$\left\{\begin{array}{l} \Delta u_1^{(j)} = f \text{ on } [\alpha, \Gamma_1] \\ u_1^{(j)}(\alpha) = 0 \\ u_1^{(j)}(\Gamma_1) = u_1^{(j-1/2)}(\Gamma_1) \end{array}\right. , \quad \left\{\begin{array}{l} \Delta u_2^{(j+1/2)} = f \text{ on } [\Gamma_1, \Gamma_2] \\ \dfrac{\partial u_2^{(j+1/2)}(\Gamma_1)}{\partial n} = \dfrac{\partial u_1^{(j)}(\Gamma_1)}{\partial n} \\ u_2^{(j+1/2)}(\Gamma_2) = u_3^{(j)}(\Gamma_2) \end{array}\right. \quad (8)$$

$$\left\{\begin{array}{l} \Delta u_3^{(j)} = f \text{ on } [\Gamma_2, \beta] \\ \dfrac{\partial u_3^{(j)}(\Gamma_2)}{\partial n} = \dfrac{\partial u_2^{(j-1/2)}(\Gamma_2)}{\partial n} \\ u_3^{(j)}(\beta) = 0 \end{array}\right. .$$

The error on subdomain i writes $e_i(x) = c_i x + d_i$.

Num. analysis for the Neumann-Dirichlet algo. (3 subdomains)

For subdomains 1 and 3 we have

$$\begin{pmatrix} \alpha & 1 & 0 & 0 \\ \Gamma_1 & 1 & 0 & 0 \\ 0 & 0 & \beta & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} c_1 \\ d_1 \\ c_3 \\ d_3 \end{pmatrix} = \begin{pmatrix} 0 \\ e_2^{(j-1/2)}(\Gamma_1) \\ 0 \\ \frac{\partial e_2^{(j-1/2)}(\Gamma_2)}{\partial n} \end{pmatrix} = \begin{pmatrix} 0 \\ e_2 g_1^{(j-1/2)} \\ 0 \\ d e_2 g_2^{(j-1/2)} \end{pmatrix} \quad (9)$$

This equation gives

$$e_1^{(j)}(x) = -\frac{e_2 g_1^{(j-1/2)}}{\alpha - \Gamma_1} x + \frac{\alpha e_2 g_1^{(j-1/2)}}{\alpha - \Gamma_1} \quad (10)$$

$$e_3^{(j)}(x) = d e_2 g_2^{(j-1/2)} x - \beta d e_2 g_2^{(j-1/2)} \quad (11)$$

Error on the second subdomain satisfies

$$\begin{pmatrix} 1 & 0 \\ \Gamma_2 & 1 \end{pmatrix} \begin{pmatrix} c_2 \\ d_2 \end{pmatrix} = \begin{pmatrix} \frac{\partial e_1^{(j)}(\Gamma_1)}{\partial n} \\ e_3^{(j)}(\Gamma_2) \end{pmatrix} = \begin{pmatrix} de_1 g_1^{(j)} \\ e_3 g_2^{(j)} \end{pmatrix} \tag{12}$$

$$e_2^{(j+1/2)}(x) = de_1 g_1^{(j)} x - de_1 g_1^{(j)} \Gamma_2 + e_3 g_2^{(j)} \tag{13}$$

Replacing $e_3 g_2^{(j)}$ and $de_1 g_1^{(j)}$, $e_2^{(j+1/2)}(x)$ writes :

$$e_2^{(j+1/2)}(x) = -\frac{x - \Gamma_2}{\alpha - \Gamma_1} e_2 g_1^{(j-1/2)} + (\Gamma_2 - \beta) de_2 g_2^{(j-1/2)} \tag{14}$$

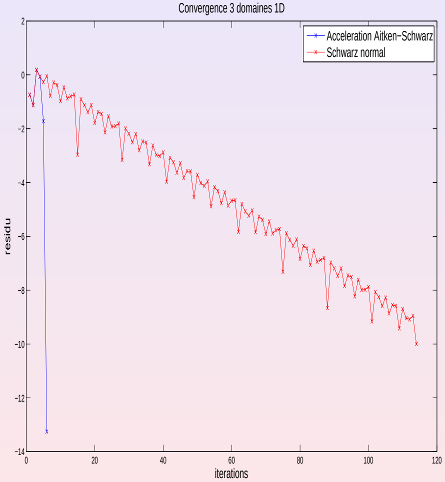
Consequently, the following identity holds :

$$\begin{pmatrix} e_2 g_1^{(j)} \\ de_2 g_2^{(j)} \end{pmatrix} = \begin{pmatrix} \frac{\Gamma_2 - \Gamma_1}{\alpha - \Gamma_1} & \Gamma_2 - \beta \\ \frac{-1}{\alpha - \Gamma_1} & 0 \end{pmatrix} \begin{pmatrix} e_2 g_1^{(j-1)} \\ de_2 g_2^{(j-1)} \end{pmatrix} \tag{15}$$

Consequently the matrix do not depends of the solution, neither of the iteration, but only of the operator and the shape of the domain.

Num. analysis for the Neumann-Dirichlet algo. (3 subdomains)

$\alpha = 0$, $\beta = 1$, $\Gamma_1 = 0.44$,$\Gamma_2 = 0.7$.



Cvg for 1D Poisson pb with 3 non-overlapping subdomains

- $\vec{x}_{i+1} - \vec{\xi} = P(\vec{x}_i - \vec{\xi}),\ i = 0, 1, \ldots$
- $\begin{pmatrix} \vec{x}_{N+1} - \vec{x}_N & \ldots & \vec{x}_2 - \vec{x}_1 \end{pmatrix} = P\begin{pmatrix} \vec{x}_N - \vec{x}_{N-1} & \ldots & \vec{x}_1 - \vec{x}_0 \end{pmatrix}$
- Thus if $\begin{pmatrix} \vec{x}_N - \vec{x}_{N-1} & \ldots & \vec{x}_1 - \vec{x}_0 \end{pmatrix}$ is non singular then $P = \begin{pmatrix} \vec{x}_{N+1} - \vec{x}_N & \ldots & \vec{x}_2 - \vec{x}_1 \end{pmatrix}\begin{pmatrix} \vec{x}_N - \vec{x}_{N-1} & \ldots & \vec{x}_1 - \vec{x}_0 \end{pmatrix}^{-1}$
  If $||P|| < 1$ then $\vec{\xi} = (Id - P)^{-1}(\vec{x}_{N+1} - P\vec{x}_N)$
- The construction of P claims at least $N + 1$ iterates if the error components are linked together. $\Rightarrow$
  - write the solution in a functional basis were the components error are decoupled
  - Construct an approximation of P

Aitken acceleration of convergence in n-D

- $\vec{x}_{i+1} - \vec{\xi} = P(\vec{x}_i - \vec{\xi}),\ i = 0, 1, \ldots$
- $\left(\vec{x}_{N+1} - \vec{x}_N \quad \ldots \quad \vec{x}_2 - \vec{x}_1 \right) = P\left(\vec{x}_N - \vec{x}_{N-1} \quad \ldots \quad \vec{x}_1 - \vec{x}_0 \right)$
- Thus if $\left(\vec{x}_N - \vec{x}_{N-1} \quad \ldots \quad \vec{x}_1 - \vec{x}_0 \right)$ is non singular then $P = \left(\vec{x}_{N+1} - \vec{x}_N \quad \ldots \quad \vec{x}_2 - \vec{x}_1 \right)\left(\vec{x}_N - \vec{x}_{N-1} \quad \ldots \quad \vec{x}_1 - \vec{x}_0 \right)^{-1}$
  If $||P|| < 1$ then $\vec{\xi} = (Id - P)^{-1}(\vec{x}_{N+1} - P\vec{x}_N)$
- The construction of P claims at least $N+1$ iterates if the error components are linked together. $\Rightarrow$
  - write the solution in a functional basis were the components error are decoupled
  - Construct an approximation of P

Aitken acceleration of convergence in n-D

- $\vec{x}_{i+1} - \vec{\xi} = P(\vec{x}_i - \vec{\xi}),\ i = 0, 1, \ldots$
- $\left(\vec{x}_{N+1} - \vec{x}_N \quad \ldots \quad \vec{x}_2 - \vec{x}_1 \right) = P\left(\vec{x}_N - \vec{x}_{N-1} \quad \ldots \quad \vec{x}_1 - \vec{x}_0 \right)$
- Thus if $\left(\vec{x}_N - \vec{x}_{N-1} \quad \ldots \quad \vec{x}_1 - \vec{x}_0 \right)$ is non singular then $P = \left(\vec{x}_{N+1} - \vec{x}_N \quad \ldots \quad \vec{x}_2 - \vec{x}_1 \right)\left(\vec{x}_N - \vec{x}_{N-1} \quad \ldots \quad \vec{x}_1 - \vec{x}_0 \right)^{-1}$
  If $||P|| < 1$ then $\vec{\xi} = (Id - P)^{-1}(\vec{x}_{N+1} - P\vec{x}_N)$
- The construction of P claims at least $N+1$ iterates if the error components are linked together. $\Rightarrow$
  - write the solution in a functional basis were the components error are decoupled
  - Construct an approximation of P

Aitken acceleration of convergence in n-D

- $\vec{x}_{i+1} - \vec{\xi} = P(\vec{x}_i - \vec{\xi}),\ i = 0, 1, \ldots$
- $\begin{pmatrix} \vec{x}_{N+1} - \vec{x}_N & \ldots & \vec{x}_2 - \vec{x}_1 \end{pmatrix} = P\begin{pmatrix} \vec{x}_N - \vec{x}_{N-1} & \ldots & \vec{x}_1 - \vec{x}_0 \end{pmatrix}$
- Thus if $\begin{pmatrix} \vec{x}_N - \vec{x}_{N-1} & \ldots & \vec{x}_1 - \vec{x}_0 \end{pmatrix}$ is non singular then $P = \begin{pmatrix} \vec{x}_{N+1} - \vec{x}_N & \ldots & \vec{x}_2 - \vec{x}_1 \end{pmatrix}\begin{pmatrix} \vec{x}_N - \vec{x}_{N-1} & \ldots & \vec{x}_1 - \vec{x}_0 \end{pmatrix}^{-1}$
  If $||P|| < 1$ then $\vec{\xi} = (Id - P)^{-1}(\vec{x}_{N+1} - P\vec{x}_N)$
- The construction of P claims at least $N + 1$ iterates if the error components are linked together. $\Rightarrow$
  - write the solution in a functional basis were the components error are decoupled
  - Construct an approximation of P

second order finite differences in y and z directions
the point interface between two subdomains in 1D is replaced by a 2D interface in 3D
Laplace Operator with Homogeneous **Dirichlet** BC in
$y = 0, z = 0,\ y = \Pi, z = \Pi$, writes

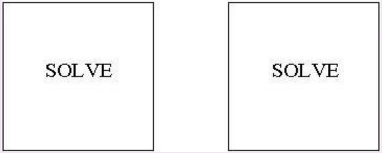
$$\sum_{j,k=1}^{n} (\hat{u}_{jk,xx} - (\frac{4}{h_y^2}\ sin^2(\frac{jh_y}{2}) + \frac{4}{h_z^2}\ sin^2(\frac{kh_z}{2}))\ \hat{u}_{jk}) sin(i.k.h_z) sin(i.j.h_y).$$

No coupling between the modes thus the operator P for the speed up is a block diagonal matrix and multi-D is analogous to the one D

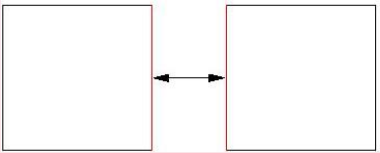
1. for relaxation method such Schwarz each wave has is own linear rate of convergence and high frequencies are damped first.
2. for high modes the matrix P can be approximate with neglecting far Macro-Domains interactions. (Less data to be send)

For a separable operator in 2D or 3D :

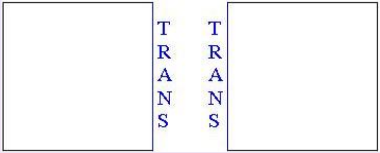
- step1 : build P analytically or numerically from data given by two Schwarz iterates
- step2 : apply one Jacobi Schwarz iterate to the differential problem with block solver of choice i.e multigrids, FFT etc...



- step3 : exchange boundary information :



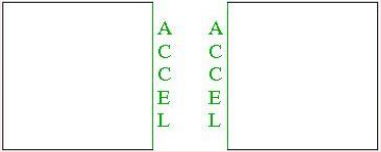
- step4 : compute the Fourier expansion $\hat{u}^n_{j|\Gamma_i}, n = 0, 1$ of **the traces on the artificial interface** $\Gamma_i, i = 1..nd$ for the initial boundary condition $u^0_{|\Gamma_i}$ and the Schwarz iterate solution $u^1_{|\Gamma_i}$.



- step5 : apply generalized Aitken acceleration based on

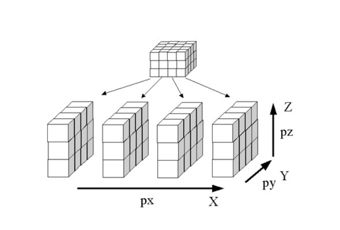
$$\hat{u}^\infty = (Id - P)^{-1}(\hat{u}^1 - P\hat{u}^0)$$

in order to get $\hat{u}^\infty_{|\Gamma_i}$.



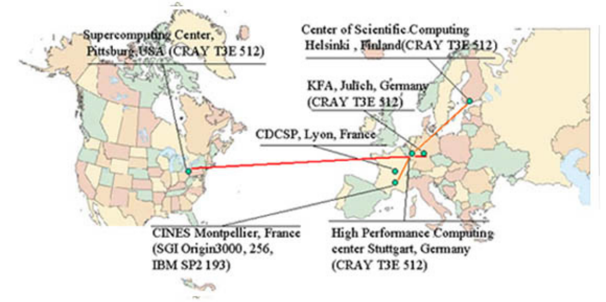
Configuration of the following results :

- 3D Domain decomposition $P_x \times P_y \times P_z$ (1D Aitken-Schwarz in x (Macrodomains M, 2D PCD3D in y and z subdomains)
- regular mesh in y and z , different sizes in x following the parallel computer power.

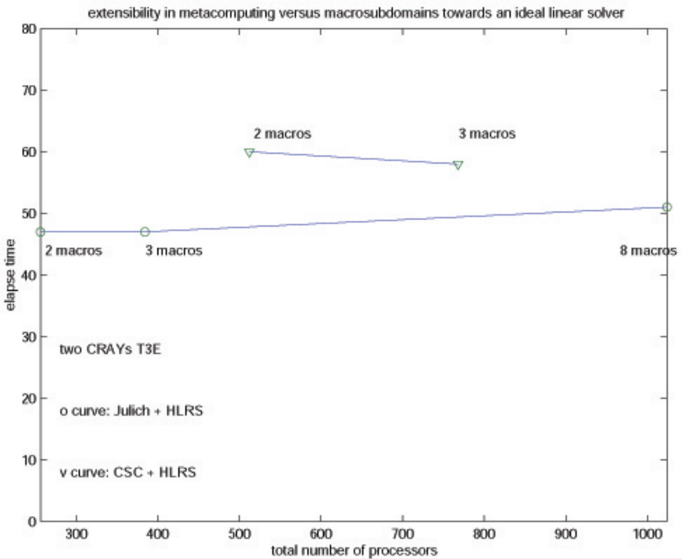


Two-level 3D domain decomposition

Large scale computing framework



- 3 Crays with 1280 procs (2 Germany, 1 USA) ,
- 732 10^6 unknowns Pb solved in less than 60s with $||e||_\infty < 10^{-8}$
- network 3-5 Mb/s (communication between 17s and 23s)
- *Barberou, Garbey, Hess, Resch, Rossi, Toivanen and Tromeur-Dervout, J. of Parallel and Distributed Computing, special issue on Grid computing, 63(5) :564-577, 2003*



Scalability of AS (with PDC3D as inner solver) in a metacomputing framework in comparison with an ideal linear solver.

For GSAM with two subdomains, errors $e^i_{\Gamma^j_h} = U^{i+1}_{\Gamma^j_h} - U^i_{\Gamma^j_h}$ satisfy

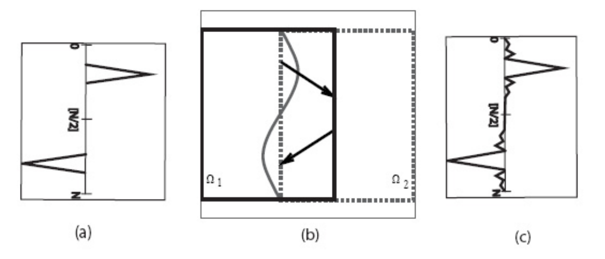
$$\begin{pmatrix} e^{i+1}_{\Gamma^1_h} \\ e^{i+1}_{\Gamma^2_h} \end{pmatrix} = P \begin{pmatrix} e^{i}_{\Gamma^1_h} \\ e^{i}_{\Gamma^2_h} \end{pmatrix} \quad (16)$$

- Γ^j_h a discretisation of the interfaces
  Γ_h to be the coarsest discretisation in the sense that it produces V the smallest set of orthonormal vectors Φ_k that belong to Γ_h with respect to a discrete hermitian form $[[.,.]]$.
- Let U_{Γ_h} be the decomposition of U_Γ with respect to the orthogonal basis V.
  $U_{\Gamma_h} = \sum_{k=0}^{N} \alpha_k \Phi_k$
- The α_k represents the "Fourier" coefficients of the solution with respect to the basis V.
  The orthogonality $\Rightarrow \alpha_k = [[U_\Gamma, \Phi_k]]$
- Then

$$\begin{pmatrix} \beta^{i+1}_{\Gamma^1_h} \\ \beta^{i+1}_{\Gamma^2_h} \end{pmatrix} = P_{[[.,.]]} \begin{pmatrix} \beta^{i}_{\Gamma^1_h} \\ \beta^{i}_{\Gamma^2_h} \end{pmatrix} \quad (17)$$

Explicit building of $P_{[[.,.]]}$

uses how basis Φ_k are modified by the Schwarz iterate.



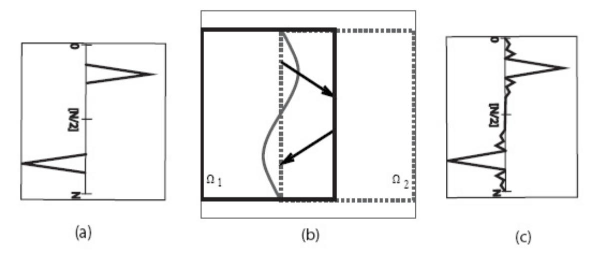
Steps to build the $P_{[[.,.]]}$ matrix

a starts from the the basis function Φ_k and get its value on interface in the physical space

b performs two schwarz iterates with zeros local right hand sides and homogeneous boundary condition on $\partial\Omega = \partial(\Omega_1 \cap \Omega_2)$

c decomposes the trace solution on the interface in the basis V. We then obtains the column k of the matrix $P_{[[.,.]]}$

uses how basis Φ_k are modified by the Schwarz iterate.

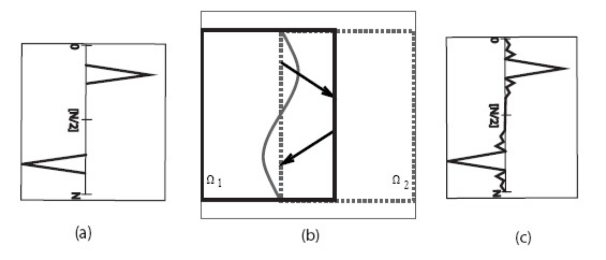


Steps to build the $P_{[[.,.]]}$ matrix

- a starts from the the basis function Φ_k and get its value on interface in the physical space
- b performs two schwarz iterates with zeros local right hand sides and homogeneous boundary condition on $\partial\Omega = \partial(\Omega_1 \cap \Omega_2)$
- c decomposes the trace solution on the interface in the basis V. We then obtains the column k of the matrix $P_{[[.,.]]}$

Explicit building of $P_{[[.,.]]}$

uses how basis Φ_k are modified by the Schwarz iterate.



Steps to build the $P_{[[.,.]]}$ matrix

- a starts from the the basis function Φ_k and get its value on interface in the physical space
- b performs two schwarz iterates with zeros local right hand sides and homogeneous boundary condition on $\partial\Omega = \partial(\Omega_1 \cap \Omega_2)$
- c decomposes the trace solution on the interface in the basis V. We then obtains the column k of the matrix $P_{[[.,.]]}$

- $P_{[[.,.]]}$ can be compute in parallel, (# local subdomain solve = # interface points, and the number of columns computed during the Schwarz iterates can be set according to the computer architecture
- Its adaptive computation is required to save computing.
- The Fourier mode convergence gives a tool to select the Fourier modes that slow the convergence.

- $P_{[[.,.]]}$ can be compute in parallel, (# local subdomain solve = # interface points, and the number of columns computed during the Schwarz iterates can be set according to the computer architecture
- Its adaptive computation is required to save computing.
- The Fourier mode convergence gives a tool to select the Fourier modes that slow the convergence.

- $P_{[[.,.]]}$ can be compute in parallel, (# local subdomain solve = # interface points, and the number of columns computed during the Schwarz iterates can be set according to the computer architecture
- Its adaptive computation is required to save computing.
- The Fourier mode convergence gives a tool to select the Fourier modes that slow the convergence.

Nonuniform Cartesian grids and/or non separable differential operator $\Rightarrow$ P is no longer diagonal

- Select Fourier modes higher than a fixed tolerance and through their decreasing factor between 2 Schwarz iterations. Index = array containing the list of selected modes.
- Take the subset $\tilde{v}$ of Fourier modes from 1 to max(Index).
- Approximate $P_{[[.,.]]}$ with $P^*_{[[.,.]]}$ using only $\tilde{v}$.
- Accelerate $\tilde{v}$ through the equation :

$$\tilde{v}^{\infty} = (Id - P^*_{[[.,.]]})^{-1}(\tilde{v}^{n+1} - P^*_{[[.,.]]}\tilde{v}^{n})$$

  Other modes are not accelerated.
- $P^*_{[[.,.]]}$ columns can be built in parallel and the number of columns computed during the Schwarz iterates can be set according to the computer architecture

Nonuniform Cartesian grids and/or non separable differential operator $\Rightarrow$ P is no longer diagonal

- Select Fourier modes higher than a fixed tolerance and through their decreasing factor between 2 Schwarz iterations. Index = array containing the list of selected modes.
- Take the subset $\tilde{v}$ of Fourier modes from 1 to max(Index).
- Approximate $P_{[[.,.]]}$ with $P^*_{[[.,.]]}$ using only $\tilde{v}$.
- Accelerate $\tilde{v}$ through the equation :

$$\tilde{v}^\infty = (Id - P^*_{[[.,.]]})^{-1}(\tilde{v}^{n+1} - P^*_{[[.,.]]}\tilde{v}^n)$$

  Other modes are not accelerated.
- $P^*_{[[.,.]]}$ columns can be built in parallel and the number of columns computed during the Schwarz iterates can be set according to the computer architecture

Nonuniform Cartesian grids and/or non separable differential operator $\Rightarrow$ P is no longer diagonal

- Select Fourier modes higher than a fixed tolerance and through their decreasing factor between 2 Schwarz iterations. Index = array containing the list of selected modes.
- Take the subset $\tilde{v}$ of Fourier modes from 1 to max(Index).
- Approximate $P_{[[.,.]]}$ with $P^*_{[[.,.]]}$ using only $\tilde{v}$.
- Accelerate $\tilde{v}$ through the equation :

$$\tilde{v}^\infty = (Id - P^*_{[[.,.]]})^{-1}(\tilde{v}^{n+1} - P^*_{[[.,.]]}\tilde{v}^n)$$

  Other modes are not accelerated.
- $P^*_{[[.,.]]}$ columns can be built in parallel and the number of columns computed during the Schwarz iterates can be set according to the computer architecture

Nonuniform Cartesian grids and/or non separable differential operator $\Rightarrow$ P is no longer diagonal

- Select Fourier modes higher than a fixed tolerance and through their decreasing factor between 2 Schwarz iterations. Index = array containing the list of selected modes.
- Take the subset $\tilde{v}$ of Fourier modes from 1 to max(Index).
- Approximate $P_{[[.,.]]}$ with $P^*_{[[.,.]]}$ using only $\tilde{v}$.
- Accelerate $\tilde{v}$ through the equation :

$$\tilde{v}^\infty = (Id - P^*_{[[.,.]]})^{-1}(\tilde{v}^{n+1} - P^*_{[[.,.]]}\tilde{v}^n)$$

  Other modes are not accelerated.
- $P^*_{[[.,.]]}$ columns can be built in parallel and the number of columns computed during the Schwarz iterates can be set according to the computer architecture

Nonuniform Cartesian grids and/or non separable differential operator $\Rightarrow$ P is no longer diagonal

- Select Fourier modes higher than a fixed tolerance and through their decreasing factor between 2 Schwarz iterations. Index = array containing the list of selected modes.
- Take the subset $\tilde{v}$ of Fourier modes from 1 to max(Index).
- Approximate $P_{[[.,.]]}$ with $P^*_{[[.,.]]}$ using only $\tilde{v}$.
- Accelerate $\tilde{v}$ through the equation :

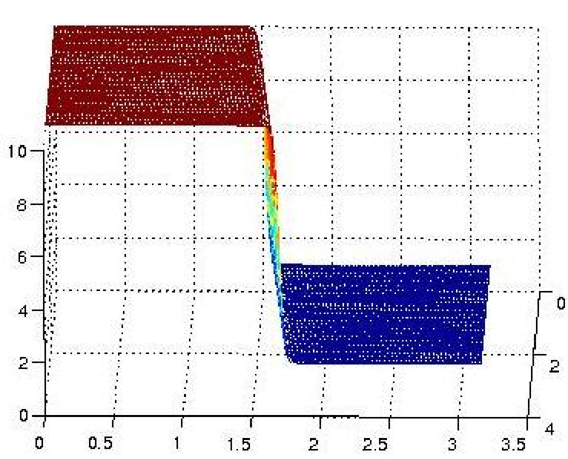
$$\tilde{v}^\infty = (Id - P^*_{[[.,.]]})^{-1}(\tilde{v}^{n+1} - P^*_{[[.,.]]}\tilde{v}^n)$$

  Other modes are not accelerated.
- $P^*_{[[.,.]]}$ columns can be built in parallel and the number of columns computed during the Schwarz iterates can be set according to the computer architecture

AS-DDM on a strongly non separable operator and irregular matching grids

$$\begin{cases} \nabla.(a(x,y)\nabla)u(x,y) = f(x,y), & \text{on } \Omega =]0,1[^2 \\ u(x,y) = 0, & (x,y) \in \partial\Omega \end{cases}$$

$a(x,y) = a_0 + (1-a_0)(1 + tanh((x - (3h * y + 1/2 - h))/\mu))/2$,
and $a_0 = 10^1, \mu = 10^{-2}$.



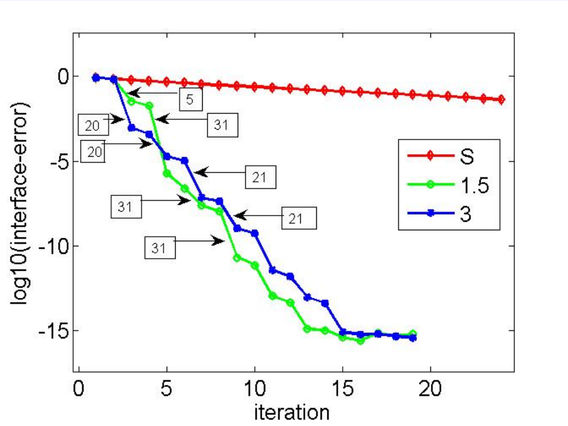


FIG.: adaptive acceleration using sub-blocks of $P_{[[.,.]]}$, with 100 points on the interface, overlap$=1$, $\epsilon = h_u/8$ and Fourier modes tolerance = $||\hat{u}^k||_\infty/10^i$ for $i = 1.5$ and 3 for 1st iteration and $i = 4$ for successive iterations.

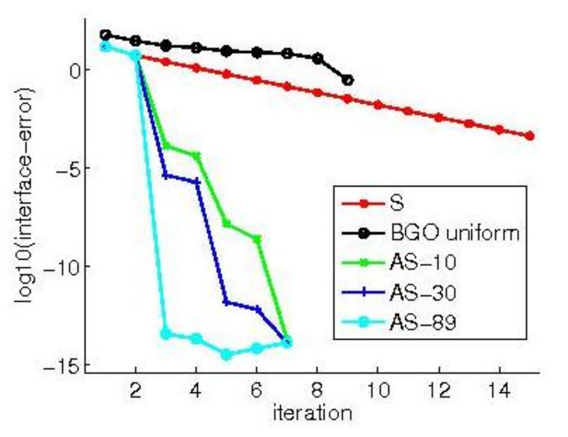
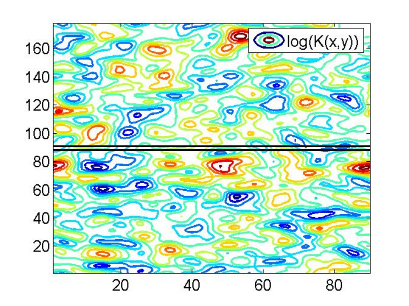


FIG.: acceleration using sub-blocks of $P_{[[.,.]]}$ with 90 points on the interface, overlap$=5$ and $\epsilon = h_u/2$. Black line refers to results for a uniform grid and overlap=5 in Baranger & al., *The Aitken-Like Acceleration of the Schwarz Method on Non-Uniform Cartesian Grids*, Technical Report Number UH-CS-05-18, 2005.

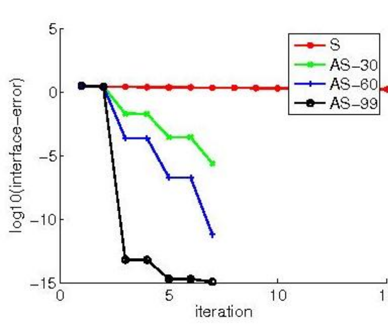
Convergence of AS in random porous media

K follows a log-normal random process

$$\nabla.(K(x,y)\nabla u) = f, \ \text{on}\Omega$$
$$u = 0, \ \text{on } \partial\Omega$$



$K(x,y) \in [0.0091, 242.66]$

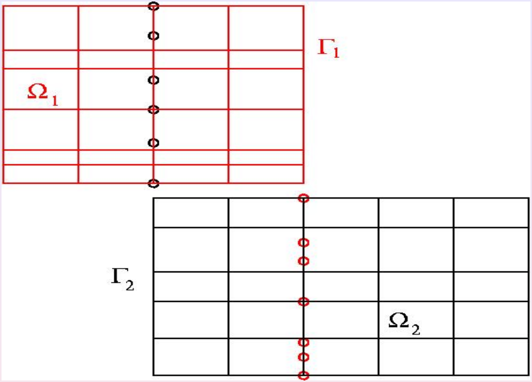


Convergence of AS

Simon Pomarede work (end engineering training period 6 months) :

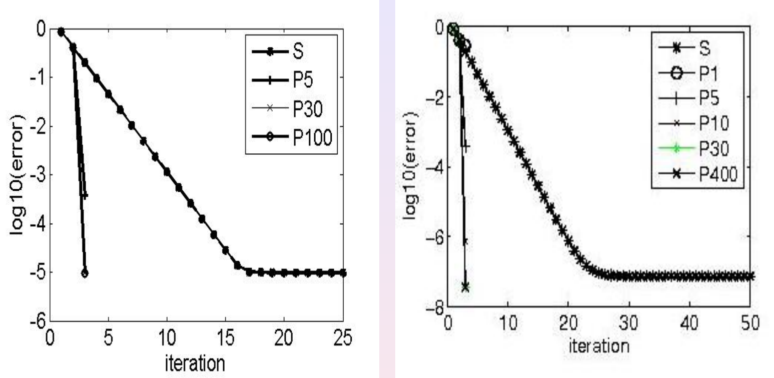
- A "block building" of the matrix $P_{[[.,.]]}$ in the same time than the Schwarz iterate.
- Implementation of the block Aitken-Schwarz in Paradis
- define different paterns of communication with respect to the mode values
- Deployment on the grid 5000 with mipch madeleine

Aitken-Schwarz for non conforming DDM

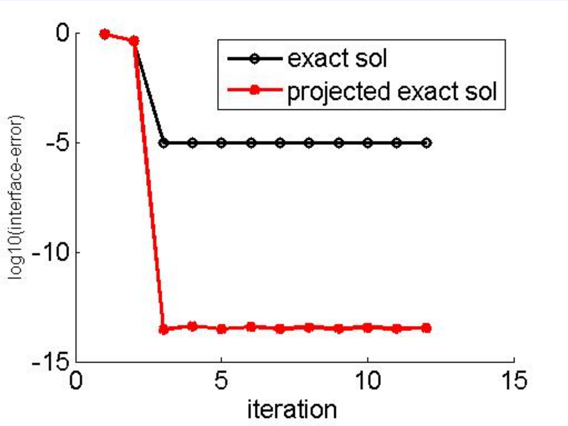


⇒ needs projection of interface solutions
⇒ use NUDFT for spectral interpolation :

$$u_2(\Gamma_2)(y_2) = \sum_{k=0}^{N_1} \hat{u}_{1,k}(\Gamma_2)\phi_{1,k}(y_2)$$



Aitken-Schwarz convergence for the Poisson problem on non uniform non conforming meshes for 100 (right) and 400 (left) interface points. The interface error is computed with respect to the exact discretised solution. $\Rightarrow$ Limitation not in the Aitken acceleration, rather on the solution representation



Aitken-Schwarz convergence for the Poisson problem on non uniform non conforming meshes for 100 interface points when the interface error is computed with respect to the exact discretised solution (black line) vs. the projected exact discretised solution (red line).

Aitken-Schwarz for Fractured network ?

- how to define a decomposition of the interface ?
  - POD of the trace ?
  - local representation ?