INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2002; **40**:1493–1513 (DOI: 10.1002/fld.407)

On some Aitken-like acceleration of the Schwarz method

M. Garbey[1,2,*,†] and D. Tromeur-Dervout[2,‡]

[1] *MCS-CDCSP-ISTIL, Université Lyon 1, 15 Boulevard Latarjet, 69622 Villeurbanne, France*
[2] *COSC Department, University of Houston, Houston, TX 77204, U.S.A.*

SUMMARY

In this paper we present a family of domain decomposition based on Aitken-like acceleration of the Schwarz method seen as an iterative procedure with a linear rate of convergence. We first present the so-called Aitken–Schwarz procedure for linear differential operators. The solver can be a direct solver when applied to the Helmholtz problem with five-point finite difference scheme on regular grids. We then introduce the Steffensen–Schwarz variant which is an iterative domain decomposition solver that can be applied to linear and nonlinear problems. We show that these solvers have reasonable numerical efficiency compared to classical fast solvers for the Poisson problem or multigrids for more general linear and nonlinear elliptic problems. However, the salient feature of our method is that our algorithm has high tolerance to slow network in the context of distributed parallel computing and is attractive, generally speaking, to use with computer architecture for which performance is limited by the memory bandwidth rather than the flop performance of the CPU. This is nowadays the case for most parallel. computer using the RISC processor architecture. We will illustrate this highly desirable property of our algorithm with large-scale computing experiments. Copyright © 2002 John Wiley & Sons, Ltd.

KEY WORDS: domain decomposition; elliptic solver; distributed computing; metacomputing

1. INTRODUCTION

Fast solvers for elliptic linear and nonlinear operators are an essential feature of modern code to solve fluid flow or heat transfer problems. Fast solvers of the Helmholtz operator might be, for example, the driving horse for incompressible Navier–Stokes code [1]. However, the definition of what one calls fast solver is nowadays very much dependent on the architecture of the computer. As a matter of fact, an optimal number of arithmetic operations to achieve a

* Correspondence to: M. Garbey, MCS-CDCSP-ISTIL, Université Lyon 1, 15 Boulevard Latarjet, 69622 Villeurbanne, France.
† E-mail: garbey@cdcsp.univ-lyon1.fr
‡ dtromeur@cdcsp.univ-lyonl.fr
http:\\cdcsp.univ-lyonl.fr

Contract/grant sponsor: ANVAR; contract/grant number: Abondement
Contract/grant sponsor: Région Rhône Alpes
Contract/grant sponsor: Rechenzentrum Universitat of Stuttgart
Contract/grant sponsor: National de l'Enseignement Supérieur

Received 1 November 2001
Revised 12 June 2002

result is not the best criterion of efficiency because it is more difficult on modern architecture to access the data than to get flops. The power of CPUs double every 18 months (Moore law) but the memory bandwidth increases at a rate of 8% per year. Further, there are one or two orders of magnitude in bandwidth performance between the first level cache memory and the main memory. In this context, it is important to design fast solvers that are numerically efficient and in the mean time cache friendly and/or highly tolerant to low bandwidth and high latency networks.

We present in this paper a family of domain decomposition technique that leads to the coding of fast solvers for computer architecture used nowadays. The basic idea is to start from the standard Schwarz algorithm [2–4] that is a rather popular method in CFD because of its robustness and simplicity to code, and to accelerate the convergences of the trace of the solution at artificial interfaces between sub-domains, via the Aitken-like method.

The two main results presented in this paper are as follows: First, a fast Helmholtz solver on Cartesian grid that has an arithmetic complexity of the same order than the classical fast Fourier transform (FFT) is presented, but in contrast to FFT it works efficiently on parallel computers with slow network and eventually in meta-computing situations. Second, our methodology is rather general and leads to a new family of linear and nonlinear elliptic solver, the so-called Steffensen–Schwarz method. Our first numerical results with this new family of method are very promising for classical test case as the Bratu problem or the p-Laplacian. We believe that our approach will lead to many different generalizations and improved variants in the near future.

The plan of this article is as follows. Section 2 presents the Aitken–Schwarz domain decomposition algorithm in the linear case. Section 3 analyses various numerical experiences with the Steffensen–Schwarz method applied to multidimensional elliptic operator with irregular geometry and non-linear elliptic operators. Section 4 reports on the performance of our solver in large-scale parallel computing. Section 5 is our conclusion.

2. AITKEN–SCHWARZ METHOD FOR LINEAR OPERATORS

We describe our domain decomposition method for linear PDE discretized on grids that are tensorial products, in square (or cubic) domains decomposed into strips or boxes.

2.1. Two sub-domains with Dirichlet–Dirichlet BC

Let us consider a linear problem

$$L[U] = f \quad \text{in } \Omega, \quad U_{|\partial\Omega} = 0 \tag{1}$$

L can be the continuous problem or the discrete one. We restrict ourselves to two sub-domains and consider the additive Schwarz algorithm. For simplicity of the description of the method, we assume implicitly in the following notations that the homogeneous Dirichlet boundary condition in (1) is satisfied by all intermediate sub-problems:

$$L[u_1^{n+1}] = f \quad \text{in } \Omega_1, \quad u_{1|\Gamma_1}^{n+1} = u_{2|\Gamma_1}^{n} \tag{2}$$

$$L[u_2^{n+1}] = f \quad \text{in } \Omega_2, \quad u_{2|\Gamma_2}^{n+1} = u_{1|\Gamma_2}^{n} \tag{3}$$

We observe that the operator T,

$$(u^n_{1|\Gamma_1} - U_{\Gamma_1}, u^n_{2|\Gamma_2} - U_{\Gamma_2}) \to (u^{n+1}_{1|\Gamma_1} - U_{\Gamma_1}, u^{n+1}_{2|\Gamma_2} - U_{\Gamma_2}) \tag{4}$$

is *linear*.

Let us consider first the one-dimensional (1D) case $\Omega = (0,1)$: the sequence $u^n_{i|\Gamma_i}$ is a sequence of real numbers. Note that as long as the operator T is linear, the sequence $(u^n_{1|\Gamma_1}, u^n_{2|\Gamma_2})$ has a pure linear convergence (or divergence); that is, it satisfies the identities

$$u^{n+1}_{1|\Gamma_2} - U_{|\Gamma_2} = \delta_1 (u^n_{2|\Gamma_1} - U_{|\Gamma_1}) \tag{5}$$

$$u^{n+1}_{2|\Gamma_1} - U_{|\Gamma_1} = \delta_2 (u^n_{1|\Gamma_2} - U_{|\Gamma_2}) \tag{6}$$

where δ_1 (respectively, δ_2) is the damping factor associated to the operator L in sub-domain Ω_1 (respectively, Ω_2) [5]. The matrix P associated to the operator T has the characteristic structure

$$\begin{pmatrix} 0 & \delta_1 \\ \delta_2 & 0 \end{pmatrix}$$

From Equations (5) and (6), we have

$$u^2_{1|\Gamma_2} - u^1_{1|\Gamma_2} = \delta_1 (u^1_{2|\Gamma_1} - u^0_{2|\Gamma_1})$$
$$u^2_{2|\Gamma_1} - u^1_{2|\Gamma_1} = \delta_2 (u^1_{1|\Gamma_2} - u^0_{1|\Gamma_2})$$

So, except if the initial boundary conditions $u^0_{2|\Gamma_1}$ or $u^0_{1|\Gamma_2}$ matches with the exact solution U at the interfaces Γ_i, the amplification factors δ_1 and δ_2 can be computed from (5) and (6). Then, if $\delta_1 \delta_2 \neq 1$ the limit $U_{|\Gamma_i}$, $i = 1,2$ is obtained as the solution of the linear systems in (5) and (6). The Aitken acceleration procedure gives, therefore, the *exact* limit of the sequence on the interface Γ_i based on two successive Schwarz iterates $u^j_{i|\Gamma_i}$, $j = 1,2$, and the initial condition $u^0_{i|\Gamma_i}$.

An additional solve of each of the subproblems in (2) and (3) with boundary conditions $u^\infty_{\Gamma_i}$ gives the solution of (1). The Aitken acceleration thus transforms the additive Schwarz procedure into an *exact* solver regardless of the speed of convergence of the original Schwarz method.

We observe that δ_1, δ_2 are dependent only on the operator and the partitioning of the domain. $\delta_{1/2}$, for example, can be computed before hand as follows: Let $v_{1/2}$ be the solution of

$$L[v_{1/2}] = 0 \quad \text{in } \Omega_{1/2}, \quad v_{|\Gamma_{1/2}} = 1 \tag{7}$$

We have then $\delta_{1/2} = v_{|\Gamma_{2/1}} \cdot v_{1/2}$ can be computed numerically and possibly analytically if the differential operator L is simple enough.

When $\delta_{1/2}$ is known, we need only *one* Schwarz iterate to accelerate the interface and an additional solvers for each sub-problems. This is a total of two solvers per sub-domain. This

feature is particularly attractive when the elliptic problem (1) has to be solved many times; a typical application can be a pressure solver or, alternatively, a stream function solver in a Navier–Stokes code.

Next let us consider a multidimensional case. We focus our attention on the Helmholtz problem $L[u]=u_{xx}+u_{yy}-\lambda u=f$ in the square $(0,\pi)^2$ with homogeneous Dirichlet boundary conditions and suppose that λ is a positive constant. For simplicity, we present the method in two space dimension; however, the three-dimensional (3D) case is a straightforward generalization. We partition the domain into two overlapping strips: $\Omega=\Omega_1\cup\Omega_2$. We introduce the regular discretization in the y direction, $y_i=(i-1)h,\ h=\pi/(N-1)$, and central second-order finite differences of the u_{yy} derivative. Let us denote $\hat{u}_k$ and $\hat{f}_k$, respectively, as the coefficient of the sine expansion of u and f, and $u_i=u(x,\mathrm{i}h)$. The Helmholtz problem decomposes into N-independent semi-discretized equations corresponding to the sine wave $\sin(ky),\ k=1,\ldots,N$,

$$\hat{u}_{k,xx}-\mu_k\hat{u}_k=\hat{f}_k \tag{8}$$

with $\mu_k=4/h^2\sin^2(kh/2)+\lambda$. Let us denote $\hat{w}_k^{j,n},\ j=1,2$ the kth term of the sine expansion of the y function $u^n_{|\Gamma_j}(y),\ y\in(0,\pi)$, generated by the Schwarz algorithm (2) and (3). Each wave $w_k^{j,n}$ is damped linearly exactly, i.e. there exists a set of damping factors δ_j^k that satisfies the identities

$$\hat{w}_k^{1,n+1}-\hat{U}^k_{|\Gamma_2}=\delta_1^k(\hat{w}_k^{2,n}-\hat{U}^k_{|\Gamma_1}) \tag{9}$$

$$\hat{w}_k^{2,n+1}-\hat{U}^k_{|\Gamma_1}=\delta_2^k(\hat{w}_k^{1,n}-\hat{U}^k_{|\Gamma_2}) \tag{10}$$

where $\hat{U}^k_{|\Gamma_j},\ j=1,2$ denotes the kth term of the sine expansion of the trace $U_{|\Gamma_j}(y),\ y\in(0,\pi)$ of the exact solution on the artificial interfaces Γ_j. Note that $\delta_1^k,\ \delta_2^k,\ k=1,\ldots,N$ depend only on the operator and the partitioning of the domain. Since λ is a constant, these damping factors can be computed analytically for the semi-discretized operator

$$L[u^h]=u_{j,xx}+\frac{u_{j+1}-2u_j+u_{j-1}}{h^2}-\lambda u_j$$

as well as for the classical five-point finite difference operator. Using Equations (9) and (10), the exact solution at the interface becomes

$$U^k_{|\Gamma_1}=[\delta_2^k(-\hat{w}_k^{1,1}+\delta_1^k\hat{w}_k^{2,0})+\delta_2^k\hat{w}_k^{1,0}-\hat{w}_k^{2,1}]/(\delta_1^k\delta_2^k-1) \tag{11}$$

$$U^k_{|\Gamma_2}=[-w_k^{1,1}+\delta_1^k w_k^{2,0}+\delta_1^k(\delta_2^k w_k^{1,0}-w_k^{2,1})]/(\delta_1^k\delta_2^k-1) \tag{12}$$

Denote by d the measure of the overlap in the strip domain decomposition $\Omega=\Omega_1\cup\Omega_2$. We remark that for small overlap $\delta_j^k\sim\exp(-\sqrt{\mu^k}d)$. The coefficients of each wave number of the trace of the solutions generated by the Schwarz algorithm has its own linear rate of convergence and the high frequencies terms are damped the fastest.

The Aitken–Schwarz algorithm is therefore very similar to the algorithm derived in the 1D case. Although computation in Ω_1 and Ω_2 can be done in parallel. The algorithm writes

- *Step* 1: Compute analytically or numerically, in parallel, each damping factor δ_j^k from each sub-problems (7) and each operator $L_k[v] = v_{xx} - \mu_k v$.
- *Step* 2: Apply one additive Schwarz iterate to the PDE problem with block solver of choice, i.e. multigrids, FFT, etc.
- *Step* 3:
  - Compute the sine expansion $\hat{u}^n_{k|\Gamma_i},\ n=0,1,\ k=1,\ldots,N$ of the traces on the artificial interface $\Gamma_i,\ i=1,2$ for the initial boundary condition $u^0_{|\Gamma_i}$ and the solution given by one Schwarz iterate $u^1_{|\Gamma_i},\ i=1,2$.
  - Apply the generalized Aitken acceleration based on (11–12) *separately* to each wave coefficients in order to get $\hat{u}^\infty_{j|\Gamma_i}$.
  - Recompose the trace $u^\infty_{|\Gamma_i}$ in physical space.
- *Step* 4: Compute, in parallel, the solution in each sub-domains $\Omega_j,\ j=1,2$ with new inner BCs and block solver of choice.

We remark that this procedure works independently of the discretization and grids in x direction as long as the block solvers for each sub-problems are exact.

Let us notice that for the Helmholtz problem with homogeneous Neumann BC instead of Dirichlet BC, one has to accelerate the cosine expansion of the interface's sequence. With non-homogeneous BC it is convenient to accelerate a shifted sequence that satisfies the homogeneous BC [6].

So far, we have restricted ourselves to domain decomposition with two sub-domains. Next, we will introduce a generalized Aitken acceleration technique that can be applied to an arbitrary number $q>2$ of sub-domains.

2.2. *More than 2 sub-domains with Dirichlet–Dirichlet BC*

First, we consider the 1D case $\Omega=(0,1)$. Let $\Omega_i=(x_i^l,x_i^r),\ i=l\ldots q$ be a partition of Ω with $x_2^l<x_1^r<x_3^l<x_2^r,\ldots,x_q^l<x_{q-1}^r$. We consider the additive Schwarz algorithm

for $i=1\ldots q$, *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

Let us denote $u_i^{l,n+1}=u_i^{n+1}(x_i^l),\ u_i^{r,n+1}=u_i^{n+1}(x_i^r)$ and let $\tilde{u}^n$ (respectively, $\tilde{u}$) be the n iterated (respectively, exact) solution restricted at the interface, i.e.

$$\tilde{u}^n=(u_2^{l,n},u_1^{r,n},u_3^{l,n},u_2^{r,n},\ldots,u_q^{l,n},u_{q-1}^{r,n})$$

The operator $\tilde{u}^n - \bar{u} \to \tilde{u}^{n+1} - \tilde{u}$ is linear. Let us denote P as its matrix. P has the following penta-diagonal structure:

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

where δ_l^r and δ_q^r can be computed as in the two sub-domains case.

The sub-blocks

$$P_i = \begin{vmatrix} \delta_i^{l,l} & \delta_i^{l,r} \\ \delta_i^{r,l} & \delta_i^{r,r} \end{vmatrix}, \quad i = 2, \dots, q-1$$

can be computed with three Schwarz iterates as follows:

We have $(u_{i-1}^{r,n+1} - \tilde{u}_{i-1}^r,\ u_{i+1}^{l,n+1} - \tilde{u}_{i+1}^l)^t = P_i(u_i^{l,n} - \tilde{u}_i^l, u_i^{r,n} - \tilde{u}_i^r)^t$. Therefore,

$$\begin{pmatrix} u_{i-1}^{r,n+3} - u_{i-1}^{r,n+2} & u_{i-1}^{r,n+2} - u_{i-1}^{r,n+1} \\ u_{i+1}^{l,n+3} - u_{i+1}^{l,n+2} & u_{i+1}^{l,n+2} - u_{i+1}^{l,n+1} \end{pmatrix} = P_i \begin{pmatrix} u_i^{l,n+2} - u_i^{l,n+1} & u_i^{l,n+1} - u_i^{l,n} \\ u_i^{r,n+2} - u_i^{r,n+1} & u_i^{r,n+1} - u_i^{r,n} \end{pmatrix}$$

In practice, the last matrix on the right-hand side of the previous equation is non-singular and P_i can be computed, but it cannot be guaranteed. However, one can always compute before hand the coefficients of P_i as follows: Let v be the solution of

$$L[v] = 0 \quad \text{in } \Omega_i, \quad v(x_i^l) = 1, \quad v(x_i^r) = 0 \tag{13}$$

and w be the solution of

$$L[w] = 0 \quad \text{in } \Omega_i, \quad w(x_i^l) = 0, \quad w(x_i^r) = 1 \tag{14}$$

We have then $\delta_i^{l,l} = v(x_{i-1}^r)$, $\delta_i^{l,r} = v(x_{i+1}^l)$, $\delta_i^{r,l} = w(x_{i-1}^r)$ and $\delta_i^{r,r} = w(x_{i+1}^l)$. We observe that this computation of the sub-blocks P_i can be done in parallel.

In addition, for the Helmholtz operator $L[u] = u'' - \lambda u$, or generally speaking elliptic problems with constant coefficients, the matrix P is known analytically.

From the equality

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

one writes the generalized Aitken acceleration as follows:

$$\tilde{u}^\infty = (\mathrm{Id} - P)^{-1}(\tilde{u}^{n+1} - P\tilde{u}^n) \tag{15}$$

If the additive Schwarz method converges, then $\|P\| < 1$ and $\mathrm{Id} - P$ is non-singular.

The algorithm is then

Step 1: Compute analytically or numerically in parallel each sub-blocks P_i from each sub-problems (13,14).

Step 2: Apply one additive Schwarz iterate.

Step 3: Apply generalized Aitken acceleration on the Interfaces based on (15) with $n=0$.

Step 4: Compute in parallel the solution for each sub-domain.

From the point of view of parallelism, steps 1 and 4 does not require any communication. Step 2 requires local communication between sub-domains that overlap. Step 3 on the contrary requires global communication. We will see in the next section how to minimize the global communications involved in step 3.

Let us return now to the two-dimensional (2D) problem with the Helmholtz operator and strip domain decomposition. Once again, we restrict ourselves to homogeneous Dirichlet boundary conditions.

We partition the domain into an arbitrary number q of overlapping strips: $\Omega = \bigcup_{i=1,\ldots,q} \Omega_i$. The algorithm is similar to the one described above in one space dimension for arbitrary q overlapping sub-domains in the following sense: the sub-domains are solved as multidimensional problems, but the acceleration is applied *independently* on each wave coefficient of the sine expansion of the interfaces as in the 1D case. Now, the algorithm becomes

- *Step* 1: For each wave number k, compute analytically or numerically in parallel each sub-blocks P_i of P from each sub-problems (13,14) and each operator $L_k[v] = v_{xx} - \mu_k v$.
- *Step* 2: Apply one additive Schwarz iterate to the PDE problem with a 2D block solver of choice, i.e. multigrids, FFT, etc.
- *Step* 3:
  - Compute the sine expansion $\hat{u}^n_{k|\Gamma_i}$, $n=0,1$ of the traces on the artificial interface Γ_i, $i=1,\ldots,q$ for the initial boundary condition $u^0_{|\Gamma_i}$ and the solution given by one Schwarz iterate $u^1_{|\Gamma_i}$.
  - Apply the generalized Aitken acceleration based on (15) with $n=0$ *separately* to each wave coefficients in order to get $\hat{u}^\infty_{k|\Gamma_i}$.
  - Recompose the trace $u^\infty_{|\Gamma_i}$ in physical space.
- *Step* 4: Compute in parallel the solution in each sub-domains Ω_i, with new inner BCs $u^\infty_{|\Gamma_i}$ and block solver of choice.

The arithmetic complexity of this algorithm can be given analytically, provided the knowledge of the arithmetic complexity of the linear solver used in each sub-domain. Let us assume for simplicity that the arithmetic complexity of a fast sine transform (or its inverse) of a vector of size N is *exactly* $5\ N\log_2(N)$ [7]. With q strip sub-domains and a problem of global size $N_x \times N_y$, the Aitken acceleration requires the sine transform and its inverse of the artificial interfaces at two iteration levels. It results into $20\times(q-1)N_y\log_2(N_y)$ operations. The solution of the penta-diagonal linear system corresponding to the acceleration procedure itself costs $36\ N_y\ (q-1)$ operations [8]. We recall that we need to solve each sub-domain problems twice.

If one uses a sparse Gaussian elimination for each sub-domain solve, the overall arithmetic complexity is therefore of order

$$6qN_xN_y\left(\frac{N_x}{q}+3\right)^2 + 20(q-1)N_y\log_2(N_y) + 36N_y(q-1)$$

If one uses a fast Poisson or Helmholtz solver [7], the arithmetic complexity becomes of order

$$20qN_y\left(\frac{N_x}{q}+2\right)\left(\log_2\left(\frac{N_x}{q}+2\right)+\log_2(N_y)\right)+20(q-1)N_y\log_2(N_y)+36N_y(q-1)$$

A numerical investigation of these two formulas shows that our method speed up significantly than the sparse Gaussian elimination for increasing number of sub-domains but can be at most two times slower than a Fast Poisson solver. A true comparison of costs depends on the architecture of the processor and we observe that our method is cache friendly. Moreover, this algorithm has a very high potential of parallelism. Steps 1 and 4 are fully parallel. Step 2 requires only local communication and scale well with the number of processors. Step 3 requires global communication of interfaces in Fourier space. In addition, the arithmetic complexity of step 3 that is the kernel of the method is negligible compared to step 2.

It is important to note from the point of view of parallelism that the main bottleneck is due to the global communication in step 3. However, the high-frequency waves have very fast convergence with the Schwarz algorithm itself because the damping factor is asymptotically small [4, 5]. In addition, the high frequency may have little influence on the final solution if the exact discrete solution is a second-order approximation of the exact solution of the PDE problem. Therefore, one can restrict adaptively the Aitken acceleration process of step 3 to a subset $\hat{u}_k^n,\ k=1,\ldots,M$, with $M<N$, and minimize the amount of global communications. In order to define quantitatively M, we need to proceed with the stability analysis of the Aitken–Schwarz algorithm.

2.3. *Sensitivity analysis*

This sensitivity analysis should be focussed on the acceleration process of the sine expansion of the interfaces generated by the Schwarz algorithm. Since the multidimensional problems are decomposed into N-independent problems corresponding to each sine wave, we will analyse the stability of problem (15).

For convenience, we restrict ourselves to the Helmholtz operator discretized with five points differences and a uniform strip domain decomposition. We use

$$\begin{pmatrix}\delta_1 & 0 & 0 & \delta_2\\ \delta_2 & 0 & 0 & \delta_1\end{pmatrix}$$

to denote the generic sub-block of P for a given wave number k.

Let $\tilde{P}$ be an approximation of P. The relative error on the artificial interface vector $\tilde{u}$ is then bounded by

$$2\frac{\|(\mathrm{Id}-P)^{-1}\|^2\|(P-\tilde{P})\|}{1-\|(\mathrm{Id}-P)^{-1}(P-\tilde{P})\|}+\|(Id-P)^{-1}(P-\tilde{P})\|$$

Since the operator L satisfies a maximum principle, this corresponds to the global error. A straightforward application of this estimate is the minimization of the communication constraints in step 3 of the Aitken–Schwarz algorithm, if one neglects interactions between subdomains that are not neighbours. It is equivalent to approximate P with the following matrix

$\tilde{P}$ for acceleration:

$$\begin{vmatrix} 0 & \delta_1 & 0 & 0 & \ldots & & & \\ \delta_1 & 0 & 0 & 0 & \ldots & & & \\ 0 & 0 & 0 & \delta_1 & \ldots & & & \\ & & & \ldots & \delta_1 & 0 & 0 & 0 \\ & & & \ldots & 0 & 0 & 0 & \delta_1 \\ & & & \ldots & 0 & 0 & \delta_1 & 0 \end{vmatrix}$$

The error on the corresponding predicted wave amplitude of the interface given by the incomplete Aitken acceleration is then bounded by

$$\eta_k = \left(2\delta_2 \frac{1+\delta_1}{1-\delta_1} + \delta_2\right) \Big/ (1-\delta_1^2)$$

for the wave number k, with

$$\delta_1 = \sinh(\sqrt{\mu}(d_x - h_x))/\sinh(\sqrt{\mu}d_x), \quad \delta_2 = \sinh(\sqrt{\mu}h_x)/\sinh(\sqrt{\mu}d_x)$$

and

$$\mu = 4/h_y^2 \sin^2\left(k\frac{h_y}{2}\right) + \lambda$$

It is clear that δ_1 and δ_2 decrease as the corresponding frequency increases. One can therefore neglect the coupling between far away sub-domains for large enough frequencies k, preserving the overall accuracy of the method. The criterion for adaptivity is to compare η_k with the truncated error due to the discretization of the continuous operator.

One can also analyse the impact of an inexact sub-block solver. Since P is known analytically for the Helmholtz operator for each wave number, the additional sources of instability in the Aitken–Schwarz procedure come from the linear solve of (15).

We have $\|P_k\| = O(1)$ and the condition number of $\mathrm{Id} - P_k$ is bounded by [9]

$$\mathrm{cond}(\mathrm{Id} - P_k) \leqslant 2\left(\frac{1}{1-\delta_1} + \frac{\delta_2(1-\delta_1)^{-2}}{1-\delta_2(1-\delta_1)^{-1}}\right)$$

with

$$\delta_1 = \sinh(\sqrt{\lambda}(d_x - h_x))/\sinh(\sqrt{\lambda}d_x), \quad \delta_2 = \sinh(\sqrt{\lambda}h_x)/\sinh(\sqrt{\lambda}d_x)$$

where d_x is the size of the Ω_i strip in x direction. The condition number is then of order h_x^{-1} at most for $\lambda = 0(1)$. A direct numerical simulation to test the sensitivity of our algorithm to perturbation on the right-hand side of the linear differential problem confirms this estimate. In addition, the linear stability of the solver deteriorates very slowly as the number of sub-domains increases as expected [6].

The elementary domain decomposition method described so far for the Poisson problem or the Helmholtz operator case in a rectangular box fails to be an exact solver if the grid has a non-constant space step in the y direction or if the operator has coefficients depending on the x and y variable, because the sine waves in y direction are no longer decoupled. We will now discuss briefly some generalizations of our method.

2.4. Remarks on the generalization of the Aitken–Schwarz procedure

First let us observe that for the Helmholtz operator case in a rectangular domain with Cartesian grid and five points finite differences, our procedure works independently of the discretization and grids in x direction as long as the block solvers for each subproblems are exact. A similar idea can be applied to Dirichlet–Neumann procedure [4] and/or various version of the Schwarz algorithm such as the red/black procedure as long as the sequence of interfaces generated by the iterative domain decomposition procedure has exact linear convergence or divergence. Furthermore, we have introduced in Reference [6] a (recursive) two levels version of this algorithm that allows domain decomposition in both space directions. However, we loose the nice decoupling in sine modes for the y interfaces when the grid has a variable spatial step in the y direction or if the operator has coefficients depending on the x and y variables.

One may use, if possible, a change of variable and construct P with a new set of basis function, to represent the interface solution, that makes P diagonal.

It is shown in Reference [10] that such a change of basis function exists for the Laplace operator discretized with finite elements with $P1$ approximation, on a tensorial grid that has *arbitrary irregular* space step in y direction.

Further, if one considers the operator $L=-\Delta u+a(x,y)u$, with a coefficient $a(x,\cdot)$ that has the cosine expansion

$$a(x,y)\approx\sum_{k=1,\ldots,N}\hat{a}_k(x)\cos((k-1)y) \tag{16}$$

then we have instead of (8), the following equation on vector $\hat{U}=(\hat{u}_i)_{i=1,\ldots,N}$;

$$-\hat{U}_{xx}+\mu\hat{U}+A\hat{U}=\hat{F} \tag{17}$$

where μ is the diagonal matrix of coefficients $\mu_k,\ k=1,\ldots,N,\ \hat{F}$ is the vector of components $\hat{f}_k,\ k=1,\ldots,N$ and A is the symmetric matrix of coefficient

$$A_{k,k}=\hat{a}_0-\tfrac{1}{2}\hat{a}_{2k}$$

and for $k\neq p$,

$$A_{k,p}=\tfrac{1}{2}\hat{a}_{|p-k|}-\tfrac{1}{2}\hat{a}_{p+k}$$

with the convention $\hat{a}_j\equiv 0$, *for* $j<0$ *or* $j>N$.

If $A=B^{-1}DB$ denotes the decomposition of A such that D is diagonal, we rewrite (17) into

$$-\hat{V}_{xx}+\mu\hat{V}+D\hat{V}=\hat{G} \tag{18}$$

where $\hat{V}=B\hat{U}$ and $\hat{G}=B\hat{F}$. Modulo the change of variable B; we are therefore back to the case where P is diagonal and the Aitken–Schwarz method works provided that the acceleration is applied independently to each $\hat{V}_k$.

However, for a more general elliptic problem, with varying coefficients they may not be a simple transform of the variables that makes the matrix P diagonal. We look then at our method in the following general framework.

2.5. A general formal framework

Let us consider the multidimensional case with the discretized version of problem (1). We restrict ourselves for simplicity to the two overlapping sub-domain cases and the additive Schwarz algorithm (2,3). Let us denote $E_i^h,\ i=1,2$, some finite vector space used to approximate the solution restricted to the artificial interface $\Gamma_i,\ i=1,2$. Let $b_i^j,\ j=1,\ldots,N$ be a set of basis functions for this vector space and P be the corresponding matrix of the linear operator T

$$u_{i|\Gamma_i}^n - U_{\Gamma_i} \to u_{i|\Gamma_i}^{n+2} - U_{\Gamma_i}$$

We denote by $u_{i,j}^n,\ j=1,\ldots,N$ the components of $u_{i|\Gamma_i}^n$. We then have

$$(u_{i,j}^{n+2} - U_{j|\Gamma_i})_{j=1,\ldots,N} = P(u_{i,j}^n - U_{j|\Gamma_i})_{j=1,\ldots,N}$$

Let us suppose that the interface sequence is such that the matrix $(u_{k,i}^{2(j+1)} - u_{k,i}^{2j})_{i=1,\ldots,N,\ j=0,\ldots,N-1}$ is non-singular. Let Id be the matrix for the identity operator. We introduce a generalized Aitken acceleration with the following formula: first

$$P = (u_{k,i}^{2(j+1)} - u_{k,i}^{2j})_{i=1,\ldots,N,\ j=1,\ldots,N} (u_{k,i}^{2(j+1)} - u_{k,i}^{2j})_{i=1,\ldots,N,\ j=0,\ldots,N-1}^{-1}, \quad k=1,2$$

and second, if $\mathrm{Id} - P$ is non-singular, the trace of the exact solution $(u_{k,i})_{i=1,\ldots,N}$ on interface $\Gamma_k,\ k=1,2$ is the solution of the linear system

$$(Id - P)(u_{k,i}^{\infty})_{i=1,\ldots,N} = (u_{k,i}^{2N+2})_{i=1,\ldots,N} - P(u_{k,i}^{2N})_{i=1,\ldots,N}$$

If this generalized Aitken procedure works, it should be *a priori* independent of the spectral radius of P, that is, the convergence of the underlined Schwarz additive iterative procedure is not needed. In conclusion, $2N+1$ Schwarz iterates produce *a priori* enough data to compute via this generalized Aitken acceleration the interface value $U_{|\Gamma_k},\ k=1,\ldots,2$. This computation is amenable to $N+1$ Schwarz iterates if one accelerates the sequence of coupled interfaces corresponding to the linear mapping

$$(u_{1|\Gamma_1}^n - U_{\Gamma_1}, u_{2|\Gamma_2}^n - U_{\Gamma_2}) \to (u_{1|\Gamma_1}^{n+1} - U_{\Gamma_1}, u_{2|\Gamma_2}^{n+1} - U_{\Gamma_2})$$

However, we can expect that the matrix $(u_{k,i}^{2(j+1)} - u_{k,i}^{2j})_{i=1,\ldots,N,\ j=0,\ldots,N-1}$ is ill-conditioned and that the computed value of P is very sensitive to the data. In addition, $2N+1$ Schwarz iterates is too many to be considered as an efficient procedure.

Nevertheless, we have numerical evidences that this procedure can perform on 2D linear elliptic problems with stiff coefficient if we use a direct solve of each subproblems [11].

We are going to present in the next section some experimental results based on strategies that compute band approximation of P.

3. STEFFENSEN–SCHWARZ METHOD FOR LINEAR AND NON-LINEAR ELLIPTIC OPERATORS

We focus our attention now to an iterative acceleration of the Schwarz procedure when the Aitken acceleration is no longer giving the exact interface condition. We will apply the same

cycle of Aitken–Schwarz procedure in a loop until appropriate convergence if any. We name this procedure a Steffensen–Schwarz algorithm. We will also restrict ourselves in this paper to the case of regular grids for which the sine or cosine expansion of the traces generated by additive Schwarz are a natural tool.

We are going to show that Steffensen–Schwarz is suitable to solve elliptic problems more complicated, than the Poisson problem in a box. In all numerical experiments thereafter, except when it is specified differently, we will consider domain decomposition with *minimum overlap*, i.e. one mesh overlap.

3.1. Linear elliptic operator

Let us consider first the linear case $L = -\Delta u + a(x, y)u$ with a varying smooth coefficient a. For simplicity of the presentation, we consider (4) with only two overlapping sub-domains.

Let us assume (16). If one approximates the coefficients a by its Z truncated cosine expansions as follows:

$$a(x, y) \approx \sum_{k=1\ldots Z} \hat{a}_k(x) \cos((k-1)y)$$

matrix P is then a sparse matrix of bandwidth $2Z+1$. Our heuristic strategy is therefore to try to rebuild from the sequence of $2Z+1$ consecutive interfaces generated by Schwarz, a band approximation P_Z of P. We look then for P_Z such that

$$(\hat{u}_i^{2Z+2} - \hat{u}_i^{2Z+1}, \ldots, \hat{u}_i^3 - \hat{u}_i^2, \hat{u}_i^2 - \hat{u}_i^1) = (P_{i,i-Z,\ldots,}P_{i,i+Z}) \times S_{\mathrm{B}} \tag{22}$$

where S_{B} is the following sub-block:

$$\begin{pmatrix} \hat{u}_{k-Z}^{2Z+1} - \hat{u}_{k-Z}^{2Z} & \ldots \hat{u}_{k-Z}^{1} - \hat{u}_{k-Z}^{0} \\ \vdots & \cdots \\ & \cdots \\ \hat{u}_{k+Z}^{2Z+1} - \hat{u}_{k+Z}^{2Z} & \ldots \hat{u}_{k+Z}^{1} - \hat{u}_{k+Z}^{0} \end{pmatrix} \tag{23}$$

provided by the Schwarz iterative process. Equation (22) holds for $Z < i < N - Z$. A similar equation can be written with appropriate reduced dimension for the end terms of the diagonal of P_Z, that is when $i \leqslant Z$ or $i \geqslant N - Z$. If S_{B} is non-singular, the kth row of P_Z is well defined. Otherwise, we have to decrease Z for this specific row until the sub-block is non-singular. In practice, the conditioning of the sub-block deteriorates when the frequency increases but only low frequencies needed to be accelerated since high frequencies are damped very fast by the Schwarz method itself.

We have shown in Reference [9] an illustration of this method for different coefficient functions $a(x, y)$ and different choices for the bandwidths Z. It can be generally observed that the faster the cosine expansion (16) converges, the smaller the Z should be.

Another source of failure in our previous Aitken–Schwarz method comes from the fact that the domain is no longer a box. Let us consider for example the Poisson problem on a polygonal domain that is a square from which we have cut off a large triangle based on one of the square side. This modification of the domain makes all modes on the artificial interfaces mixed. Figures 1 and 2 report on our numerical experiment with the Steffensen–

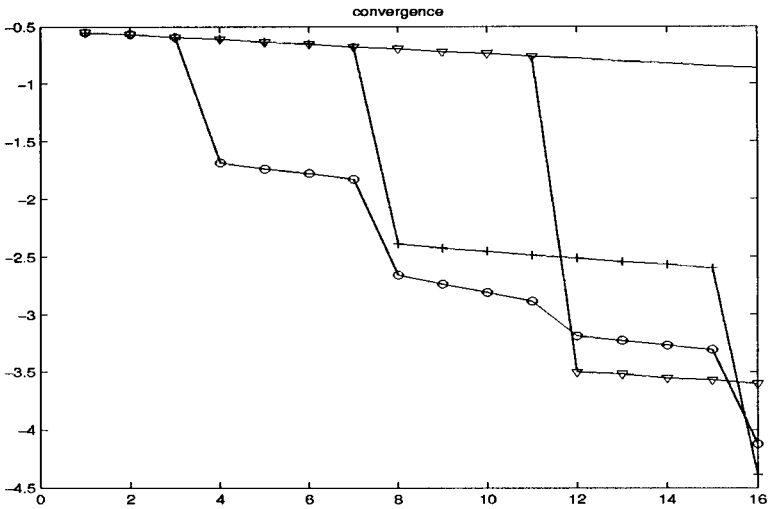

Figure 1. Convergence history for irregular geometry.

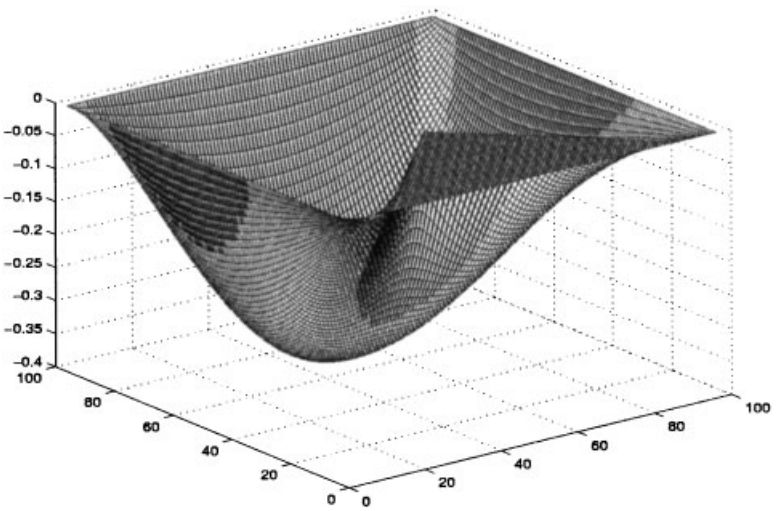

Figure 2. Solution of the Poisson problem with a reentry corner.

Schwarz algorithm with the reconstruction of band approximation of P. Convergence curves are commented with + sign for $Z=1$, o sign for $Z=2$ and v sign for $Z=3$. We see in this example that all methods seems to converge, but it is difficult to predict what is the best method. It is a dilemma between using as often as possible an acceleration that may not be the most efficient because P is approximated by a diagonal matrix or using an acceleration of the interface a priori more efficiently because it uses a larger band approximation of P, but more expensive since more Schwarz iterates are required to construct P_Z.

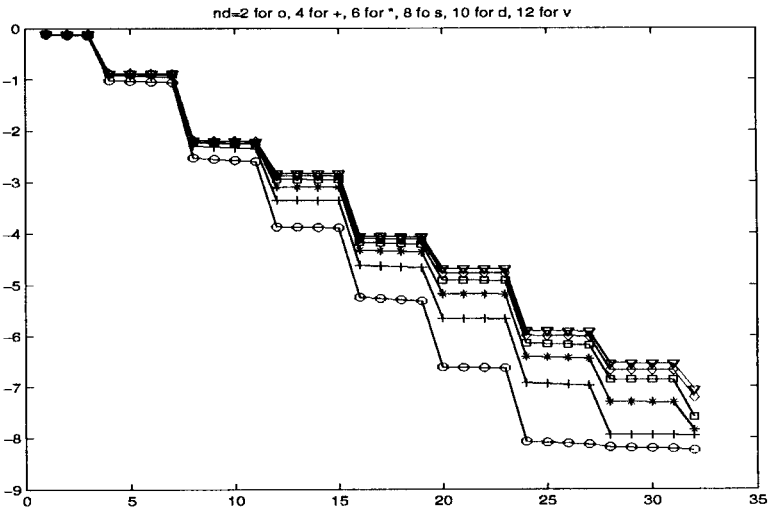

Figure 3. Convergence for the 2D Bratu problem.

We plan to develop an adaptive procedure to improve this rather heuristic method.

The Steffensen acceleration process is classically considered as an accelerate method to solve *non-linear problem* [12]. It is based on the assumption of the linear rate of convergence of the iterative process to be accelerated. We will consider next the acceleration of the Schwarz iterative procedure for a non-linear problem.

3.2. The non-linear case

Let us consider first the 2D boundary-value problems that can be seen as a second-order perturbation of the Laplacian. From this point of view the *Bratu* problem [13]

$$-\Delta u = \lambda e^{u} \quad \text{in } \Omega = (0,1)^2, \quad u_{|\partial\Omega} = 0 \tag{24}$$

is a good test case. It corresponds to a simplified model of solid combustion. Other similar test case from semi-conductor modelling can be considered [14, 9]. This Bratu problem has a smooth solution for $\lambda \in (0, 6.81)$. We have experimented the Steffensen–Schwarz algorithm for the classical five points finite difference scheme with strip domain decomposition, an arbitrary number of sub-domains and $\lambda = 6$. In each sub-domain, we use a Newton procedure to solve the sub-problem. We observe that the preconditioner of the corresponding linear system might be computed only once in a while since the non-linear exponential term is a small perturbation of the Laplacian operator. We have used a conjugate gradient method with incomplete LU preconditioner. We have observed in numerical experiments that the Steffensen–Schwarz algorithm with diagonal approximation of P is best. However, non-homogeneous boundary conditions might lead to a different conclusion.

Figure 3 shows the convergence history of our methods with a grid of approximatively fixed size 60×60 and an increasing number of sub-domains from 2 to 12. We observe that each step between two plateau in the convergence history has about the same size. It can be seen that, unfortunately, the 1D quadratic convergence property that might be obtained from

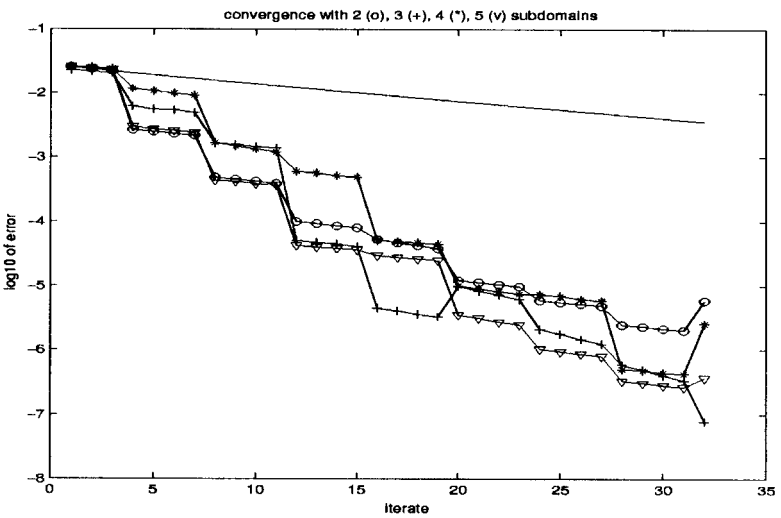

Figure 4. Convergence for the p-Laplacian problem with $p = 2.5$.

the Steffensen method in the 1D case [12] is lost. As a matter of fact, the linear approximation of the operator is not separable; therefore, no matter whether the Schwarz iterates get close to the exact solution, the Aitken acceleration efficiency based on a diagonal approximation of P does not improve. However, it is most interesting to notice that the number of Steffensen–Schwarz iterates required to reach a given level of accuracy depends slightly on the number of sub-domains. The total number of Schwarz iterates to reach an error less that 10^{-7} in maximum norm is 24 with three sub-domains, and 32 with 12 sub-domains. An alternative solution would have been to use Steffensen–Schwarz as a linear solver for the global linear problem corresponding to each Newton step on the global problem. This solution will be reported in Section 4.

We consider now a more difficult situation with the p-Laplacian operator

$$-\mathrm{div}[a(|\nabla u|)\nabla u] = f \quad \text{in } (0,\pi)^2$$

with homogeneous boundary conditions and $a(u) = |u|^{p-2}$. This operator is a first-order perturbation of the Poisson operator, since we have non-linear terms associated to the first-order derivatives. In each sub-domain, we adopt the following non-linear solver based on the iterative procedure:

$$-\mathrm{div}[a(|\nabla u^n|)\nabla u^{n+1}] = f \quad \text{in } (0,\pi)^2$$

It can be shown that the corresponding fixed point formulation converges iff $1 < p < 3$, [15]. We show in Figure 4 (respectively, Figure 5) the convergence history with $p = 2.5$ (respectively, $p = 1.6$). In these experiments, we solve a 48×48 problems with strip domain decomposition and 2 to 5 domains. Our experiments show that we should take larger overlap than one mesh step in order to obtain a fast convergence. The result presented here are for three meshes overlap. As for the Bratu problem, the convergence is relatively insensitive to the number of sub-domains for p larger than 2. However, the convergence history may be-

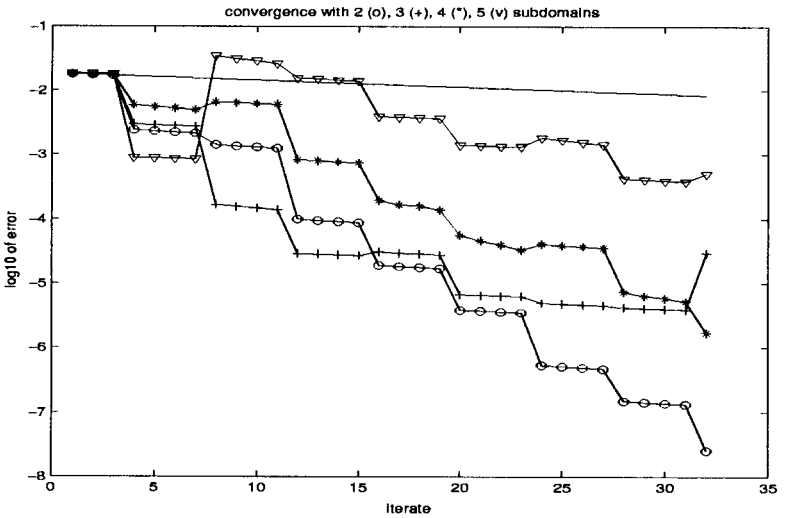

Figure 5. Convergence for the p-Laplacian problem with $p = 1.6$.

come chaotic for p less than 2 and large number of sub-domains. Once again, we will have to develop an adaptive procedure to make this acceleration procedure more robust.

We are now going to present the parallel efficiency of this new family of domain decomposition algorithm. Further, we would like to notice that non-homogeneous boundary conditions are amenable to homogeneous boundary conditions on the non-linear case via a shifting technique [9].

4. APPLICATION TO DISTRIBUTED COMPUTING

Let us consider first the performances of the Aitken–Schwarz algorithm for a 3D Poisson problem discretized on a Cartesian grid. The domain is decomposed into a small number of overlapping cubic sub-domains with minimum overlap in one space direction, let us say x direction. As mentioned in Section 2, the algorithm for the 3D case is entirely analogous to the 2D case, except that the interfaces are some (y,z) square plan, and that one has to compute 2D sine expansions of these interfaces, in order to accelerate individually each pair of sine waves $[\sin(k_1 y), \sin(k_2 z)]$. To mention an application of this solver, it can be used as a preconditioner in a pressure solve using the fictitious domain decomposition approach for flow in complex geometry [1]. This pressure solve can be the most time consuming part of the flow simulation running on a multicluster.

Each sub-domain is solved in parallel with a parallel fast direct solution method for linear systems with separable tridiagonal matrices. This software, the so-called PDC3D, developed by Rossi and Toivanen [16, 17] is one of the most numerically efficient Poisson solver among all the fast solvers. This method has the arithmetical complexity $O(N \log^2 N)$ and is closely related to the cyclic reduction method, but instead of using the matrix polynomial factorization, the so-called partial solution technique is employed [18, 19]. The PDC3D is scalable and has

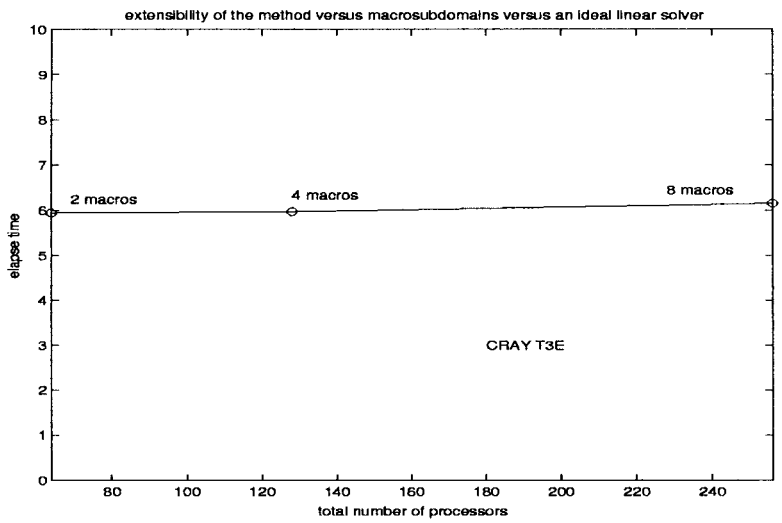

Figure 6. Extensibility of the Aitken–Schwarz algorithm for the 3D Poisson problem.

almost perfect speedup on large problems run by a Cray T3E as described in References [16, 17]. However, the parallel performance of this fast solver is strongly affected by slow network of multiclusters or in meta-computing situations [20, 1].

Figure 6 gives the elapse time for the following growing size of Poisson problems $158\times192\times384$, $316\times192\times384$, $633\times192\times384$. When one increases the number of the domains in the same proportion as the number of processors, the elapse time remains constant. Our solver has therefore good scalability properties on the Cray T3E. Further, our method requires no more than 6 s to solve the problem with 46×10^6 unknowns on a Cray T3E with 256 processors running at 450 MHz. Figure 7 shows also that the speedup of our solver is fairly good.

Similar results have been obtained with the Bratu problem using a Newton method. We apply the Aitken–Schwarz algorithm on the linear problem at each Newton step. However, since the linearized operator is no longer separable, we use instead of PDC3S a parallel multigrid algorithm for each linear sub-problems. Let us notice that this incomplete multigrid is less performant than PDC3D for the separable linear operator by a factors two roughly, but is numerically efficient in more general situations. On the contrary, our 3D linear Poisson solver that combines multigrid with Aitken–Schwarz is relatively insensitive to slow networks and give good performances on a network of few parallel servers connected by a 2 Mb/s connection.

Figures 8–10 give the percentage of the elapsed time devoted to solve the linearized problem on each macro-domain in the Aitken–Schwarz for each Newton iteration for different runs. In other words, this percentage represents the percentage of the total time spent in the multigrid solver and is noted μ. Rest of the percentage of elapse time is spent in Schwarz communication time between the macro-domains, FFT of interfaces and Aitken acceleration. The number of processors in y (respectively, z) direction is p_y (respectively, p_z). The local number of points per processor in x (respectively, y and z) is set to be N_x (respectively, N_y and N_z). The number of macro-domain is set to be p_x. The total number of points in the computational domain

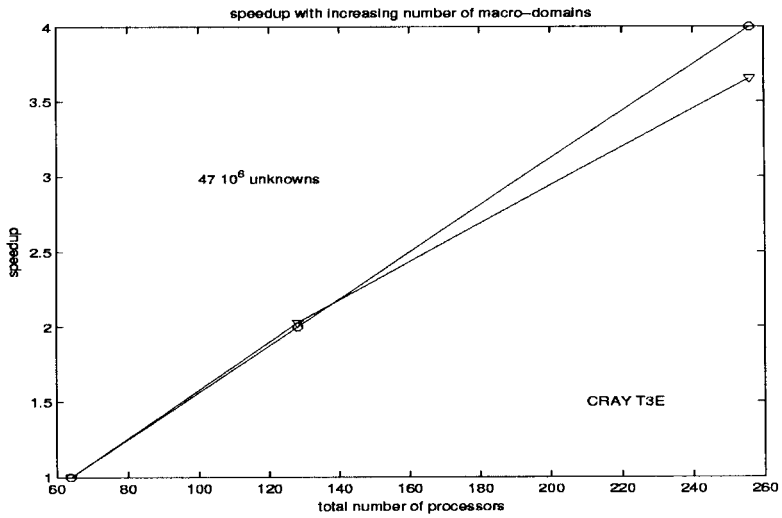

Figure 7. Speed up of the Aitken–Schwarz algorithm for the 3D Poisson problem.

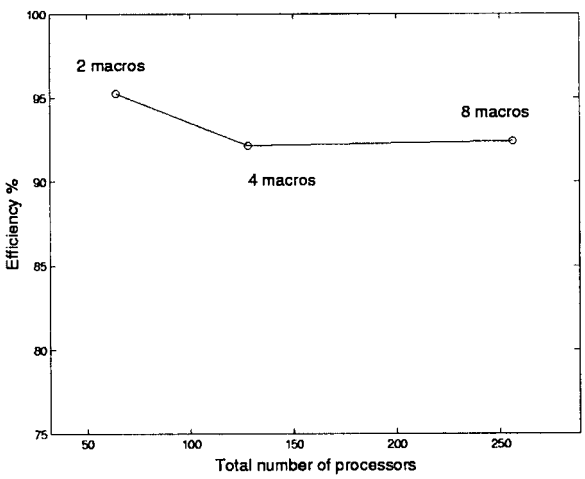

Figure 8. Efficiency of extensibility with the number of processors in x direction of the Aitken–Schwarz algorithm for the 3D Bratu problem.

is then $(p_x \times (N_x - 2) + 2) \times (p_y \times (N_y - 2) + 2) \times (p_z \times (N_z - 2) + 2)$ and the total number of processors is $p_x \times p_y \times p_z$. Figure 8 represents the behaviour of μ when the number p_x varies from 2 to 8 with $N_x = N_y = 64$, $N_z = 32$, $p_y = 4$, $P_z = 8$. It shows the extensibility efficiency of the method, the size of the computational domain increasing with the number of processors. No more than 8% of the total time is spent in the Schwarz communication and acceleration procedure.

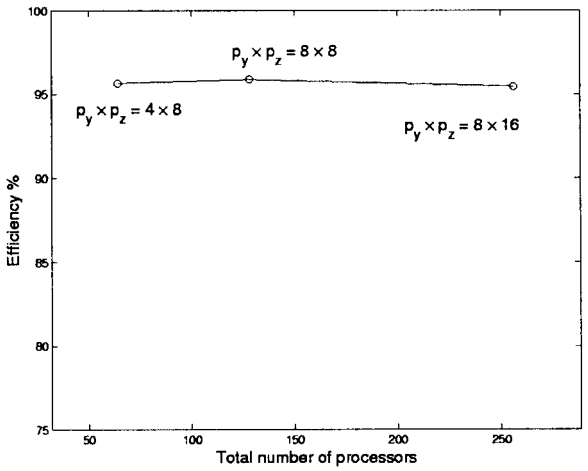

Figure 9. Efficiency of the extensibility with the number of processors in y and z directions of the Aitken–Schwarz algorithm for the 3D Bratu problem.

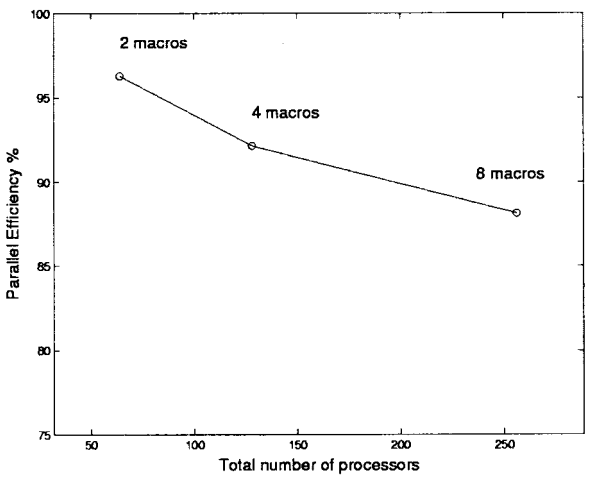

Figure 10. Extensibility of the Aitken–Schwarz algorithm for the 3D Poisson problem.

Figure 9 represents the behaviour of μ when the number p_y (respectively, p_z) varies from 4 to 8 (respectively, 8 to 16) with $p_x = 2$. It represents the perfect extensibility of the method with respect to the number of processors in the y and z direction, with 5% spent elsewhere than in the local solver.

Figure 10 represents the behaviour of μ when p_x varies from 2 to 8, with a fixed size of the computational domain. It represents the decrease of the parallel efficiency of the domain decomposition method.

We have run a similar experiment with meta-computing between large parallel systems located in different countries in order to validate our approach on 3D large-scale problems. We refer to Reference [20] for a precise description of these experiments.

5. CONCLUSION

We have analysed and tested in this paper a new family of domain decomposition solvers for linear and non-linear elliptic problems. Our method is relevant to produce a fast Poisson or Helmholtz solver on 3D Cartesian grid that keeps it numerically efficient on large parallel computers and eventually in meta-computing situations. We also have constructed algorithms that are fairly efficient to solve non-linear elliptic problems as the Bratu problem or the p-Laplacian. The future of this methodology is, however, linked to the progress of computer architectures. We believe that for many coming years, low-memory bandwidth will be the handicap of fast computers, and therefore the algorithm presented in this paper will be useful in this context.

ACKNOWLEDGEMENTS

We would like to thank the Rechenzentrum Universitat of Stuttgart and the Centre Informatique National de l'Enseignement Supérieur for their support.

REFERENCES

1. Garbey M, Vassilevsky Y. A parallel solver for unsteady incompressible 3D Navier–Stokes equation. *Parallel Computing* 2001; **27**(4):363–389.
2. Schwarz HA. *Gesammelte Mathematische Abhandlungen*, vol. 2. Springer: Berlin, 1890; 133–143. (First published in Vierteljahrsschrift der Naturforschenden Gesellschaft in Zürich, vol. 15, 1870; 272–286.)
3. Lions P.-L. On the Schwarz alternating method. I. In *First International Symposium on Domain Decomposition Methods for Partial Differential Equations*, Glowinski R, Golub GH, Meurant GA, Périaux J (eds). SIAM: Philadelphia, PA, 1988; 1–42.
4. Kuznetsov YA. Overlapping domain decomposition methods for fe-problems with elliptic singular perturbed operators. In *Fourth International Symposium on Domain Decomposition Methods for Partial Differential Equations*, Glowinski R, Kuznetsov YA, Meurant GA, Périaux J, Widlund O (eds). SIAM: Philadelphia, PA, 1991.
5. Garbey M. A schwarz alternating procedure for singular perturbation problems. *SIAM Journal on Scientific Computing* 1996; **17**(5):1175–1201.
6. Garbey M, Tromeur-Dervout D. Aitken–Schwarz method on Cartesian grids. In *Proceedings of the International Conference on Domain Decomposition Methods DD13*, Debit N, Garbey M, Hoppe R, Périaux J, Keyes D, Kuznetsov Y (eds). CIMNE: Barcelona, 2002; 53–65 (Invited Lecture).
7. Loan CFV. *Computational Frameworks for the Fast Fourier Transform*, Frontiers in Applied Mathematics. SIAM: Philadelphia, PA, 1992.
8. Golub GH, Loan CFV. *Matrix Computations* (2nd edn). Johns Hopkins University Press: Baltimore, MD, 1989.
9. Garbey M, Tromeur-Dervout D. Two level domain decomposition for multi-cluster. In *Proceedings of the International Conference on Domain Decomposition Methods DD12*, Chan HKT, Kako T, Pironneau O (eds). CIMNE: Barcelona, 2001; 325–340. (Invited lecture), http://applmath.tg.chiba-u.ac.jp/dd12/proceedings/Garbey.ps.gz.
10. Baranger J, Garbey M, Oudin-Dardun F. Recent development on Aitken–Schwarz method. In *Proceedings of the International Conference on Domain Decomposition Methods DD13*, Debit N, Garbey M, Hoppe R, Périaux J, Keyes D, Kuznetsov Y (eds). CIMNE: Barcelona, 2001; 287–294.
11. Garbey M, Tromeur-Dervout D. Operator splitting and domain decomposition for multicluster. In *Proceedings of the International Conference on Parallel CFD99*, Keyes D, Ecer A, Satofuka N, Fox P, Periaux J (eds). North-Holland: Williamsburg, 1999; 27–36. (Invited Lecture).
12. Henrici P. *Elements of Numerical Analysis*. Wiley: New York-London-Sydney, 1964.

13. Wieners C. A parallel newton multigrid method for high order finite elements and its application to numerical existence proofs for elliptic boundary value problem. *Combustion Theory and Modelling* 1996; **76**(3):175–180.
14. Selberherr S. *Analysis and Simulation of Semiconductor Devices*. Springer: Wien, New York, 1984.
15. Baranger J. Personal communication, 2000.
16. Rossi T, Toivanen J. A nonstandard cyclic reduction method, its variants and stability. *SIAM Journal on Matrix Analysis and Applications* 1999; **20**(3):628–645.
17. Rossi T, Toivanen J. A parallel fast direct solver for block tridiagonal systems with separable matrices of arbitrary dimension. *SIAM Journal on Scientific Computing* 1999; **20**(5):1778–1796.
18. Vassilevski PS. Fast algorithm for solving a linear algebraic problem with separable variables. *Comptes Rendus Academie Bulgare Sciences* 1984; **37**:305–308.
19. Kuznetsov YA, Matsokin M. On partial solution of systems of linear algebraic equations. *Soviet Journal of Numerical Analysis and Mathematical Modelling* 1989; **4**(6):453–468.
20. Barberou N, Garbey M, Hess M, Resch M, Rossi T, Toivanen J, Tromeur-Dervout D. Scalable numerical algorithms and software tools for efficient metacomputing of elliptic equations. In *Proceedings of the International Conference on Parallel CFD2001*, Wilders P, Ecer A, Satofuka N, Fox P, Periaux J (eds). Elsevier/North-Holland: Amsterdam, 2002.

