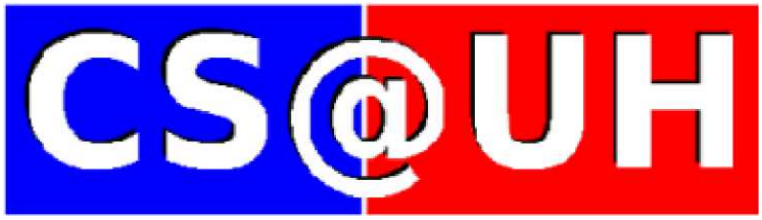

The Aitken-Like Acceleration of the Schwarz Method on Non-Uniform Cartesian Grids

J.Baranger, M. Garbey and F.Oudin-Dardun

Department of Computer Science
University of Houston
Houston, TX, 77204, USA
http://www.cs.uh.edu

Technical Report Number UH-CS-05-18

July 21, 2005

Keywords: Domain Decomposition, Aitken like acceleration, Elliptic Operators, Nonlinear Operators,Non-Uniform Cartesian Grids, Beowulf clusters

Abstract

In this paper, we present a family of domain decomposition based on an Aitken like acceleration of the Schwarz method seen as an iterative procedure with linear rate of convergence. This paper is a generalization of the method first introduced at the 12^{th} International Conference on Domain Decomposition that was restricted to regular Cartesian grids. The potential of this method to provide scalable parallel computing on a geographically broad grid of parallel computers was demonstrated for some linear and nonlinear elliptic problems discretized by finite differences on a Cartesian mesh. The main thrust of this paper is to present a generalization of the method to non-uniform Cartesian meshes. The salient feature of the method consists of accelerating the sequence of traces on the artificial interfaces generated by the Schwarz procedure using a good approximation of the main eigenvectors of the trace transfer operator. For linear separable elliptic operators, our solver is a direct solver. For nonlinear operators, we use an approximation of the eigenvectors of the Jacobian of the trace transfer operator. The acceleration is then applied to the sequence generated by the Schwarz algorithm applied directly to the nonlinear operator.

MCS-ISTIL-Université Lyon1, baranger@mcs.univ-lyon1.fr
COSC Dept. -University of Houston, garbey@cs.uh.edu, USA
MCS-ISTIL-Université Lyon1, foudin@mcs.univ-lyon1.fr

The Aitken-Like Acceleration of the Schwarz Method on Non-Uniform Cartesian Grids

J.Baranger, M. Garbey and F.Oudin-Dardun

Abstract

In this paper, we present a family of domain decomposition based on an Aitken like acceleration of the Schwarz method seen as an iterative procedure with linear rate of convergence. This paper is a generalization of the method first introduced at the 12^{th} International Conference on Domain Decomposition that was restricted to regular Cartesian grids. The potential of this method to provide scalable parallel computing on a geographically broad grid of parallel computers was demonstrated for some linear and nonlinear elliptic problems discretized by finite differences on a Cartesian mesh. The main thrust of this paper is to present a generalization of the method to non-uniform Cartesian meshes. The salient feature of the method consists of accelerating the sequence of traces on the artificial interfaces generated by the Schwarz procedure using a good approximation of the main eigenvectors of the trace transfer operator. For linear separable elliptic operators, our solver is a direct solver. For nonlinear operators, we use an approximation of the eigenvectors of the Jacobian of the trace transfer operator. The acceleration is then applied to the sequence generated by the Schwarz algorithm applied directly to the nonlinear operator.

Index Terms

Domain Decomposition, Aitken like acceleration, Elliptic Operators, Nonlinear Operators,Non-Uniform Cartesian Grids, Beowulf clusters

I. Introduction

The idea of using Aitken acceleration [14], [24], on the classical additive Schwarz Domain Decomposition (DD) method [17], [18], [19], [22], [23] has been introduced in [10] . These authors have called the corresponding method Aitken-Schwarz (AS) method. They have shown its very good numerical performances on linear and nonlinear elliptic problems discretized by a five point scheme on a rectangular Cartesian grid [13]. More recently, it was shown that this technique gives efficient meta-computing of the Poisson and Bratu problem in three space dimensions with supercomputers abroad linked by a regular internet connection [4], [9]. For an elliptic separable operator with constant coefficient, AS method is a direct solver. All these studies have been made on uniform meshes. In this case the (discrete) Fourier transform plays a crucial role (although the name of Fourier does not appear in the name of the method).

More recently a general framework of the AS method was introduced by one of the authors [8]. The technique was applied to elliptic problems with the finite volume discretization of Faille on arbitrary quadrangle cells [5]. It was shown that a compact representation of the trace of the solution generated by the Schwarz method could be used to improve drastically the convergence rate of the method via Aitken acceleration.

In this paper we propose to extend the AS method to Cartesian grids that are not necessarily uniform. Because the grid is a tensorial product of one dimensional grid with arbitrary space step, one can compute some numerical approximation of the main eigenvectors of the trace transfer operator more easily than with unstructured grids. In this paper we will present the method for finite differences as well as finite element approximations.

For linear separable elliptic operators, our solver is a direct solver. For nonlinear operators, we use an approximation of the eigenvectors of the Jacobian of the trace transfer operator. The acceleration is then applied to the sequence of traces generated by the Schwarz algorithm applied directly to the nonlinear problem. This acceleration procedure is repeated iteratively until convergence, in a scheme analogous to the Steffensen algorithm. While our method can be applied to an arbitrary number of space directions, the numerical efficiency of our method is justified in this paper by computing various linear and nonlinear test cases in two space dimensions.

MCS-ISTIL-Université Lyon1, baranger@mcs.univ-lyon1.fr
COSC Dept. -University of Houston, garbey@cs.uh.edu, USA
MCS-ISTIL-Université Lyon1, foudin@mcs.univ-lyon1.fr

It should be point out that our method requires only postprocessing of the traces generated by the Schwarz method. Our algorithm is therefore easy to implement to accelerate the convergence of an existing parallel code, and does not add a high overhead on communications. The plan of this paper will be as follows. In Sect. 2, we introduce a formalism for the trace transfer operator that is appropriate to construct the Aitken like acceleration. In Sect. 3, we compare our solver for the Helmholtz problem to other methods implemented in the general purpose library PETSc. This result motivates the generalization presented in Sect. 4 with a generalized Fourier transform for rectangular meshes with arbitrary space steps. Sect. 5 shows an application to finite element on tensorial product of one-dimensional grids. Sect. 6 gives a broader range of examples with some linear nonseparable operators and nonlinear test cases. Sect. 7 is our conclusion.

II. General Framework of the Method

The AS method is built upon three ideas:

1) The Schwarz method is an iterative method applied to a trace transfer operator acting on functions defined on the interfaces. Sparsity of the Jacobian of this operator is related to the domain decomposition (DD).
2) The discretization and choice of the interface representation may in some cases, if well chosen, increase this sparsity.
3) For an operator with a sparse matrix, simple acceleration processes can be constructed using, for example, the so-called Aitken acceleration procedure.

We are going to present the formal construction of the trace transfer operator of the Schwarz method for a DD where the connection between subdomains can be described by a one dimensional graph. We make no assumption on the PDE problem to be solved, except that all subdomain problems should be well posed.

We consider a bounded domain Ω in $\mathbb{R}^N$ with a two neighbors overlapping DD in P domains Ω_p, i.e Ω_p only intersects Ω_{p-1} and Ω_{p+1}, with obvious modifications for p=1 and P.

The non-oriented graph associated to this DD is a segment. Let us notice that the case of an annular domain with Ω_1 only intersects Ω_P and Ω_2, and Ω_P only intersects Ω_{P-1} and Ω_1 is handled with minor modifications.

The boundary Γ_p of Ω_p is decomposed into three parts: Γ_p^l (resp.Γ_p^r) included in Ω_{p-1} (resp.Ω_{p+1}) and the remaining part $\tilde{\Gamma}_p$.

Let (Π) be a boundary value problem well posed in Ω. One step of the additive Schwartz DD method with Dirichlet-Dirichlet boundary conditions is: for all p, given the Dirichlet boundary conditions l_p (resp.r_p) on Γ_p^l (resp.Γ_p^r) solve the problem (Π_p) restriction of (Π) to Ω_p with these boundary conditions and the one of (Π) on $\tilde{\Gamma}_p$; (Π_p) is assumed to be well posed.

We denote by $\bar{r}_{p-1}$ (resp.$\bar{l}_{p+1}$) the trace of the solution of (Π_p) on Γ_{p-1}^r (resp.Γ_{p+1}^l). So one step of the Schwarz method is described by one application of the trace transfer operator

$$(\bar{l}_2, \bar{r}_1, \ldots, \bar{l}_P, \bar{r}_{P-1}) = T(l_2, r_1, \ldots, l_p, r_{p-1}, \ldots, l_P, r_{P-1}), \tag{1}$$

acting on trace spaces of functions or distributions adapted to the boundary value problem. Note that if (Π) is a system, then for each p, l_p and r_p are vectors.

T has the special structure:

$$\bar{l}_2 = T_1^r(r_1), \ldots, \left\{ \begin{array}{l} \bar{r}_{p-1} = T_p^l(l_p, r_p) \\ \bar{l}_{p+1} = T_p^r(l_p, r_p) \end{array} \right\} p = 2, \cdots, P-1, \ldots, \bar{r}_{P-1} = T_P^l(l_P)$$

Here $(\bar{r}_{p-1}, \bar{l}_{p+1}) = T_p(l_p, r_p)$ is composed of a local solver of the boundary value problem (Π_p) and the trace operators on Γ_{p-1}^r and Γ_{p+1}^l. These operators can be exact or approximated.

Formally the $2(P-1)$ Jacobian matrix of T has the pentadiagonal structure:

$$\left\{\begin{array}{ccccccc} 0 & \delta_1^{rr} & 0 & 0 & & & \\ \delta_2^{ll} & 0 & 0 & \delta_2^{lr} & 0 & & \\ \delta_2^{rl} & 0 & 0 & \delta_2^{rr} & 0 & 0 & \\ 0 & 0 & \delta_3^{ll} & 0 & 0 & \delta_3^{lr} & \ddots \\ & 0 & \delta_3^{rl} & 0 & 0 & \delta_3^{rr} & \ddots \\ & & 0 & 0 & \delta_4^{ll} & 0 & \ddots \\ & & & \ddots & \ddots & \ddots & \ddots \end{array}\right\} \tag{2}$$

with $\delta_p^{lr} = \partial T_p^l / \partial r_p(l_p, r_p)$. The derivatives are assumed to exist in some sense in the traces functional spaces.

We have described the matrix of the trace transfer operator in an abstract form that is completely general. We will now specify the discretization and therefore address the problem of the generalization of the method [13] for Cartesian grids with arbitrary irregular space step in each space direction.

We introduce a discrete approximation of the traces. Each trace l_p (resp. r_p) is approximated by J numbers l_{pj} (resp. r_{pj}), $j = 1$ to J. These numbers may be point values, coefficients in a basis,...etc.

l_p and r_p are now J-vectors and $\delta_p = \begin{pmatrix} \delta_p^{ll} & \delta_p^{lr} \\ \delta_p^{rl} & \delta_p^{rr} \end{pmatrix}$ is a 2J square matrix. T is an application from $\mathbb{R}^{2J(P-1)}$ into itself with a sparse Jacobian matrix.

For some particular problems, DD and meshes, a well chosen change of unknowns $l_{pj} \rightarrow \hat{l}_{pj}$ may greatly increase the sparsity of the Jacobian of the transformed trace transfer operator $\hat{T}$. This idea, which is the core of AS method, has been introduced on a uniform mesh-using Fourier transform-in [10]. We are going to recall in the next section this algorithm and compare its efficiency to some classical fast elliptic solvers.

III. Application to the Helmholtz Problem on a Square Cartesian Grid

We consider the situation where $N = 2$ and Ω is a rectangle with a strip DD into rectangles. The left (resp. right) boundary of Ω_p is $x = x_p^l$ (resp. x_p^r). Interfaces of the DD are therefore parallel to the y direction. 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 dimensions, however the three-dimensional case is a straightforward generalization [4]. We introduce the regular discretization in the y direction $y_j = (j-1)h$, $h = \frac{1}{J}$, and central second-order finite differences (FD) of the u_{yy} derivative. Let us denote by $\hat{u}_m$ (resp. $\hat{f}_m$) the coefficient of the sine expansion of u (resp. f) and $u_j = u(x, jh)$. The Helmholtz problem decomposes into J independent semi-discretized equations corresponding to the sinus waves $sin(ky), k = 1..J$,

$$\hat{u}_{m,xx} \; - \; \mu_m \, \hat{u}_m = \hat{f}_m, \tag{3}$$

with $\mu_m = 4/h^2 \; sin^2(m\frac{h}{2}) + \lambda$.

The coefficients $\delta^{ll}, \delta^{lr}, \delta^{rl}, \delta^{rr}$ in (2) can be computed analytically using the basis functions of the two-dimensional vector space of the solutions of (3) for each wave component. We get a matrix P_m, similar to (2) for each wave component $m = 1..J$

The DD algorithm writes then

- step 1: apply one additive Schwarz iterate to the PDE problem with a two-dimensional subdomain solver of choice (i.e., multigrids, FFT etc...)
- step 2:
  - compute the sine expansion $\hat{u}^m_{k|\Gamma_p}, \; n = 0, 1$ of the traces on the artificial interface $\Gamma_p, \; p = 1..P$, for the initial boundary condition $u^0_{|\Gamma_p}$ and the solution given by one Schwarz iterate $u^1_{|\Gamma_p}$.
  - apply generalized Aitken acceleration with $n = 0$ *separately* to each wave coefficient:

$$\tilde{u}_m^{\infty} = (Id - P_m)^{-1}(\tilde{u}^1 - P\tilde{u_m}^0)$$

  in order to get $\hat{u}^{\infty}_{m|\Gamma_p}$.

- recompose the trace $u^{\infty}_{|\Gamma_p}$ in physical space.

- step 3: compute in parallel the solution in each subdomain Ω_p, with new inner BCs $u^{\infty}_{|\Gamma_p}$ and subdomain solver of choice.

We have compared the efficiency of our AS solver with a number of other elliptic solvers on a Beowulf cluster of dual 32 bit AMD processors connected by a Gigabit ethernet switch. This switch exhibits high latency compared to a Myrinet switch, for example, but is much cheaper. We have used for this comparison PETSc [2] that is an excellent general purpose software for PDE problems. PETSc consists of a variety of libraries which include many linear solvers such as Lapack, Krylov solver and algebraic multigrid solver.

In Figure 1 , we report the speedup performance of PETSc and AS on the same graphic, while on Figure 2, we give the elpased time. We choose to run PETSc using V-cycle multigrid. The preconditioner is of Richardson type to get traditional (non-Krylov accelerated) multigrid. One has two pre and two post smoothing steps of SSOR (running independently on each process) and direct LU on the coarsest grid. To be accurate, the option in the PETSc code is:

```
-dmmg_nlevels 3  mg_levels_ksp_type richardson -mg_levels_pc_type
sor -mg_levels_pc_sor_lits 2 -mg_levels_pc_sor_local_symmetric
```

This combination of options seems to give some of the best performances for PETSc.

We use the GMRES implementation of Sparskit [21] to solve the subdomain problem. This choice gives the best elapse time in the framework of the software Sparskit.

Our implementation of AS does not neglect any wave components of the interface and uses blocking broadcast and gathers for the acceleration process in Step 3 of the algorithm. This implementation is then far from optimum. We refer to [9] for a detailed description of these comparisons.

PETSc, as expected, is faster than our implementation of AS with two processors and also for three processors. However, as the number of processors increases, one can observe that the multigrid solver does not speed up well, while AS performs better. Eventually, for more than three processors, AS gives a better elapsed time than the multigrid solver. PETSc requires a better network than Gigabit ethernet to get better performances.

This is by no means a general conclusion because this test case is particularly simple. But it is rather a demonstration than the AS algorithm is tolerant for high latency network, while traditional optimum solvers are not.

We should have used PETSc as a subdomain solver, and AS for the DD method in this specific case. This is part of our ongoing software development. This result however motivates further studies to generalize the AS algorithm toward more complex situations as presented in the next section.

IV. A Generalized Fourier Transform for Rectangular Meshes with Arbitrary Space Steps

We present here an extension to nonuniform rectangular meshes in a FD context with the one dimensional DD of Sect. 3.

(Π) is a homogenous Dirichlet boundary value problem whose equation $Lu = f$ has a separable second order operator:

$$L = L_1 + L_2$$

$$L_1 = a_1\partial_{xx} + b_1\partial_x + c_1, \quad L_2 = a_2\partial_{yy} + b_2\partial_y + c_2.$$

a_1, b_1, c_1 are functions of x, and a_2, b_2, c_2 are functions of y. Our main objective now will be to rewrite the discretized problem, in such a way that we get a set of one dimensional decoupled problems on the interface to take advantage of the classical scalar acceleration technique on linear sequences of numbers [14], [24].

Since we are mainly interested in the trace of the solution on the artificial interfaces, we can first concentrate on the semi-discretization of the operator in y variable. We use an irregular mesh in y: $y_j, j = 0, \ldots, J+1$, L_2^k a discretization of L_2 on the y-mesh. $u_j(x)$ (respectively $f_j(x)$) is an approximation of $u(x, y_j)$ (respectively $f(x, y_j)$). The semi-discrete approximation (Π_p^k) of problem (Π_p) is solved on a rectangle denoted by [e,w]x[n,s] in order to simplify the notations:

$$L_1 u_j(x) + L_2^k u_j(x) = f_j(x), \quad x \in]e, w[\tag{4}$$

$$u_j(w) \text{ and } u_j(e) \text{ given} \tag{5}$$

$$u_0(x) = u_{J+1}(x) = 0. \tag{6}$$

We now show that a suitable change of unknown (the generalized Fourier transform) increases the sparsity of the Jacobian of $\hat{T}$. We set :

$$u_j(x) = \sum_{m=1}^{J} \hat{u}_m(x)\Phi_{mj}; j = 1, \cdots, J$$

where the Φ_{mj} are to be chosen and satisfy also $\Phi_{m0} = \Phi_{mJ+1} = 0$, $m = 1, \cdots, J$. The same transformation is applied to $f_j(x)$. Extending the relation between u and $\hat{u}$ to $j = 0$ and $j = J + 1$ satisfies (6). Applying the hat transform to (4-5) gives:

$$\sum_{m=1}^{J} [\Phi_{mj}(L_1\hat{u}_m(x) - \hat{f}_m(x)) + \hat{u}_m(x)L_2^k\Phi_{mj}] = 0$$

$$\sum_{m=1}^{J} \Phi_{mj}\hat{u}_{ml}(e/w) \text{ given}$$

If we introduce the eigenvalue problem:

$$L_2^k\Phi_m = \lambda_m\Phi_m, \ \Phi_{m0} = \Phi_{mJ+1} = 0 \qquad (E)$$

we obtain formally:

$$\sum_{m=1}^{J} \Phi_{mj}[(L_1 + \lambda_m)\hat{u}_m(x) - \hat{f}_m(x))] = 0 \tag{7}$$

$$\sum_{m=1}^{J} \Phi_{mj}\hat{u}_m(e/w) \text{ given} \tag{8}$$

and we have the following result:

Theorem 4.1: Assume problem (E) has J linearly independent real eigenvectors associated to real eigenvalues. Then each problem (Π_p^k) is constituted of J uncoupled continuous one dimensional linear problems $m = 1, \cdots, J$:

$$[L_1 + \lambda_m]\hat{u}_m(x) = \hat{f}_m(x) \tag{9}$$

$$\hat{u}_m(e/w) \text{ given} \tag{10}$$

The hat trace transfer operator is affine on $\mathbb{R}^{2J(P-1)}$ with a block-diagonal matrix of J blocks of the form (2) ; the m-th diagonal block corresponds to the mode Φ_m and uses the $\delta_{pm} = \begin{pmatrix} \delta_{pm}^{ll} & \delta_{pm}^{lr} \\ \delta_{pm}^{rl} & \delta_{pm}^{rr} \end{pmatrix}$ associated to $L_1 + \lambda_m$.

Proof: the eigenvectors of problem (E) being independent, the matrix Φ_{mj} is invertible and the equations (7-8) give the first part of the theorem.

We denote by e' (resp. w') the right bound of the left neighboring domain (resp. the left bound of the right neighboring domain). The local part of the trace transfer operator is

$$(u(e), u(w)) \to (u(e'), u(w'))$$

and its hat transform is

$$(\hat{u}(e), \hat{u}(w)) \to (\hat{u}(e'), \hat{u}(w')).$$

The second part of the theorem follows then from(9-10) □

We can now proceed with the full discretization of the problem. But the Aitken like acceleration will make use only of the interface representation in the appropriate eigenvector basis $\hat{u}_m$. This allows the scalar Aitken formula to be applied independently to each component to give the exact interface condition.

We introduce an irregular x-mesh $x_i,\ i = 0, \cdots, I+1$, L_1^h a discretization of L_1 on the x-mesh. The unknowns u_{ij} are approximations of $u(x_i, y_j)$. We set $f_{ij} = f(x_i, y_j)$ and $U_j = (u_{ij})_{i=0,\ldots,I+1}$. From (6), we have $U_0 = U_{J+1} = 0$. The full discrete approximation (Π_p^{hk}) of problem (Π_p) is

$$L_1^h U_j + L_2^k U_j = F_j, j = 1, \cdots, J \tag{11}$$

$$u_{oj} \text{ and } u_{I+1j} \text{ given},\ j = 1, \cdots, J \tag{12}$$

With the generalized Fourier transform:

$$u_{ij} = \sum_{l=1}^{J} \hat{u}_{il} \Phi_{lj}; j = 1, \ldots, J$$

we have the same theorem as in the case of the semi-discretization in y except that the J continuous hat problems (9-10) are replaced by the following discrete hat problem $m = 1, \ldots, J$:

$$[L_1^h + \lambda_m]\hat{U}_m = \hat{F}_m \tag{13}$$

$$\hat{U}_{0m}, \hat{U}_{I+1m} \text{ given} \tag{14}$$

Remark 4.1: The case of cylindrical domain $\Omega =]e, w[\times B$, $B \subset \mathbf{R}^{N-1}$ with boundary Γ is (in theory) similar. We look for $u_j(x)$ an approximation of $u(x, y_j)$ where the y_j are the vertexes of a mesh in B. Let *J* (resp. J_Γ) be the index set of interior points in B (resp.on Γ). L_2 is a multidimensional y-operator and L_2^k its approximation on the mesh in B. The semi-discrete problem Π_p^k is for $j \in J$

$$L_1 u_j(x) + L_2^k u_j(x) = f_j(x),\ \ x \in]e, w[\tag{15}$$

$$u_j(e) \text{ and } u_j(w) \text{ given} \tag{16}$$

$$u_j(x) = 0, x \in]e, w[, j \in J_\Gamma \tag{17}$$

The generalized Fourier transform $u_j(x) = \sum_{m \in J} \hat{u}_m(x) \Phi_{mj}$ introduces the eigenvalue problem with eigenvectors $\Phi_m = (\Phi_{mj})_{j \in J}$:

$$L_2^k \Phi_m = \lambda_m \Phi_m,\ \Phi_{m0} = \Phi_{mJ+1} = 0, \qquad (E_c)$$

We can then generalize in principle the DD method of [4] for the three dimensional case to tensorial product of one dimensional grids with arbitrary space steps.

Remark 4.2: The case of a cylindrical domain $A\times]s, n[$ is better described using the notations of section 2. The semi-discrete problem Π_p^k is for $j = 1, \ldots, J$:

$$L_1 u_j(x) + L_2^k u_j(x) = f_j(x),$$

$$u_j(x) \text{ given on } \Gamma_p^l, \Gamma_p^r, \text{ and boundary conditions of } \Pi \text{ on } \tilde{\Gamma}_p$$

$$u_0(x) = u_{J+1}(x) = 0, x \in]e, w[.$$

Theorem 4.1 may be adapted to this situation.
The case $\Omega = A \times B$ may be handled by mixing the two previous techniques.

We will now use this discretization framework to present the generalized AS algorithm for nonuniform Cartesian grids.

The Schwarz method can be considered as an iterative method for the hat transform of T which is vectorial of size $2J(P-1)$; so in general a large system.

Any acceleration process can be used [14], [24]. The AS method uses Aitken acceleration, taking advantage of the sparsity of the hat transform of T coming on one hand from the special DD, and on the other hand of the generalized Fourier transform. In fact all the modes of the hat transform are uncoupled. This allows us to use the special version of Aitken method introduced in [10] for each independent wave component problem

$$[L_1^h + \lambda_m]\hat{U}_m = \hat{F}_m,\ x \in (e, w).$$

To be more specific the left (resp. right) boundary e (resp. w) corresponds to the index i_p^l (resp.i_p^r); the right boundary of Ω_{p-1} (resp. the left boundary of Ω_{p+1}) corresponds to the index i_{p-1}^r (resp.i_{p+1}^l).

The trace transfer operator in Ω_p is

$$(l_p, r_p) \mapsto (\bar{r}_{p-1}, \bar{l}_{p+1})$$

with $l_p = \hat{U}_{i_p^l m}, r_p = \hat{U}_{i_p^r m}, \bar{r}_{p-1} = \hat{U}_{i_{p-1}^r m}, \bar{l}_{p+1} = \hat{U}_{i_{p+1}^l m}$.

First we observe that we can use the superposition principle to construct the solution without any acceleration process. We introduce $\tilde{U}_m$ solution of

$$[L_1^h + \lambda_m]\tilde{U}_m = \hat{F}_m$$

$$\tilde{U}_{i_p^l m} = \tilde{U}_{i_p^r m} = 0;$$

V_m solution of:

$$[L_1^h + \lambda_m]V_m = 0$$

$$V_{i_p^l m} = 1, V_{i_p^r m} = 0;$$

and W_m solution of:

$$[L_1^h + \lambda_m]W_m = 0$$

$$W_{i_p^l m} = 0, W_{i_p^r m} = 1$$

Then we have $\hat{U}_m = \tilde{U}_m + l_p V_m + r_p W_m$, so the local trace transfer operator in Ω_p for the mode m is:

$$\bar{r}_{p-1} = \tilde{U}_{i_{p-1}^r m} + l_p V_{i_{p-1}^r m} + r_p W_{i_{p-1}^r m}$$

$$\bar{l}_{p+1} = \tilde{U}_{i_{p+1}^l m} + l_p V_{i_{p+1}^l m} + r_p W_{i_{p+1}^l m}.$$

Note that the matrix δ_{mp} in Theorem 3.1 is:

$$\delta_{pm} = \begin{pmatrix} V_{i_{p-1}^r m} & W_{i_{p-1}^r m} \\ V_{i_{p+1}^l m} & W_{i_{p+1}^l m} \end{pmatrix}$$

The following algorithm is then an exact solver

Algorithm 1
step 1: compute the eigenelements $\lambda_m, \Phi_m, m = 1$ to J solution of problem (E).
step 2: compute $\hat{F}$.
step 3: in all subdomains and for all m solve the three one dimensional problems giving $\tilde{U}_m, V_m, W_m$. (V_m, W_m give the δ_{pm} of the matrix P_m of the affine operator $\hat{T_m}$, and $\tilde{U}_m$ the constant part c_m of this operator).
step 4: for all m solve $(I - P_m)\hat{b}_m^\infty = c_m$, where $b = (\cdots, l_p, r_{p-1}, \cdots)$ to obtain the hat transforms of the traces.
step 5: recompose the physical traces from the result of step 4.
step 6: from these traces, make one step of Schwarz method (i.e., solve the two-dimensional problems in each subdomain).

This algorithm is very similar to the DD method of Averbuch et al [1], [15] and also [12]. As described in detail in these references, this algorithm must be modified to reduce the overhead on communications between sub-domains. Further this algorithm is limited to linear operators and needs a careful and fairly technical implementation.

In the present work, we focus on a method that (i) can make use of an existing parallel implementation of the basic Schwarz method that is very simple to code, (ii) is limited to the postprocessing of the interface sequences in order to speed up the code, (iii) can be extended to several approximation theory frameworks including non regular meshes or non matching grids, and (iv) might be applied directly on nonlinear operators. The practical range of application of this method is therefore not limited to the uncoupled situation described above. However, in the particular situation of a separable linear operator, the Aitken acceleration provides a direct solver with the following algorithm:

Algorithm 2
step 1: compute the eigenelements $\lambda_m, \Phi_m, m = 1$ to J solution of problem (E).
step 2: given the traces on the interfaces, make three steps of Schwarz method.
step 3: take the generalized Fourier transform of the last four traces.
step 4: apply the one-dimension Aitken-Steffensen acceleration formula to each mode of these transformed traces.
step 5: recompose the physical traces from the result of step 4.
step 6: from these traces, make one step of Schwarz method.

We notice that step 2 and step 6 are processed by a basic additive Schwarz implementation for the elliptic solver. One can use any solver of choice for each sub-domain such as multigrids (or preconditioned Krylov) methods if a good initial guess is available, or a fast direct solver as in [20].

The main cost of each acceleration corresponds to the computation of the eigenvectors in step 1. This problem is of order equal to the size J of the interface. The QR algorithm, for example, requires $9J^3$ flops. Step 1 is a preprocessing step that can be done once and for all, if the elliptic problem should be solved many times. This is the case for the pressure equation in unsteady Incompressible Navier-Stokes simulation that uses the projection method [9]. Further step 1 can be done in asynchronous mode with distributed computing because the J problems (E) are totally independent. Step 3 to step 5 is the postprocessing procedure that is the kernel of our method. It can be coded independently of the main code for additive Schwarz.

As shown in [4], [13] this acceleration procedure might be done adaptively as a function of the eigenvalue λ_m in order to minimize the amount of global and local communications. To be more specific here, because of the evaluation cost of the eigenelements of (E) and the fact that in many cases the high modes are damped very fast by the Schwarz method itself, it is worthwhile to use only a limited number of modes. In that case steps 3 and 5 are modified and the direct and inverse hat transform uses only the $J' < J$ first modes.

For more general problems the AS method is fully iterative and uses a Steffensen iteration [14], [24] procedure on each mode of the hat transform of the traces as follows:

Algorithm 3
step 1: compute once and for all the eigenelements $\lambda_m, \Phi_m, m = 1$ to J solution of problem (E).
step 2 to step 6 are identical to the corresponding steps in *Algorithm* 2.
step7: check convergence. If necessary go to step2 for an additional cycle.

We observe that one needs to apply the one-dimensional Aitken-Steffensen acceleration of step 4 (algorithm 2 or 3) only on the generalized Fourier transform of the traces, that is to say, only on the different interfaces of the sub-domains, which limits the costs of the computation of the eigen-elements.

One advantage of our method is that it can be applied to many different situations involving block-wise relaxation method for the Schwarz iteration that may not require overlapping. It can be applied, for example, to the Dirichlet-Neumann DD method. Let us choose for example for problem (Π_p) a Neumann condition on Γ_p and a Dirichlet condition on Γ_{p-1}. Denoting by d_p and n_p the Dirichlet and Neumann traces on Γ_p the trace transfer operator is:

$$d_2 = T_1^r(n_1), \ldots, \left\{ \begin{array}{l} n_{p-1} = T_p^l(d_p, n_p) \\ d_{p+1} = T_p^r(d_p, n_p) \end{array} \right\} p = 2 \text{ to } P-1, \ldots, n_{P-1} = T_P^l(d_P)$$

with a matrix that takes the form (2). For the full discretization we approximate the normal derivative by a linear combination of some mesh values; then the trace transfer operator has an uncoupled structure. This is of course the key to get a direct solver. We refer to [10] for an application of this type with a transmission problem. It should be noticed that our technique relies only on the construction of the main eigenvectors of this operator. Our method can be extended easily to Robin boundary conditions or similar optimized interface conditions [7], and possibly boundary conditions that include relaxation terms [6].

Finally let us emphasize that the extension of this method to multidimensional DD associated to a graph that is a multidimensional grid can be done in a straightforward way with the multilevel DD of [11] that uses at each level a strip DD algorithm.

Let us now consider our DD method in the framework of Finite Element (FE) approximations of elliptic problems.

V. Application to Finite Element on Tensorial Product of One-D Grids

We still restrict ourselves to a separable operator, but in a form that is better suited to FE approximation:

$$L_1 = -\partial_x(a_1\partial_x) + b_1\partial_x + c_1, \quad L_2 = -\partial_y(a_2\partial_y) + b_2\partial_y + c_2,$$

The problem to be solved on the rectangle $R = [e, w] \times [n, s]$ is:

$$L_1\bar{u} + L_2\bar{u} = \bar{f}, \text{ in} R, \quad \bar{u}(x, s/n) = 0, \quad \bar{u}(w/e, y) = u_{w,y}/u_{e,y},$$

The change of unknown $u = \bar{u} - u_w\frac{e-x}{e-w} - u_e\frac{x-w}{e-w}$ gives the homogenous Dirichlet problem:

$$L_1u + L_2u = f, \quad u(x, s/n) = 0, \quad u(w/e, y) = 0,$$

with variational form: u in H_0^1, for all v in H_0^1

$$\int_R (a_1u_xv_x + b_1u_xv + c_1uv + a_2u_yv_y + b_2u_yv + c_2uv - fv) = 0. \tag{18}$$

We consider a semi discrete FE approximation of this variational problem. On a y-mesh we have a FE space with basis function $\varphi_q, q = 1, \cdots, J$. The unknown function is $u^k(x, y) = \sum_{q=1}^{J} u_q(x)\varphi_q(y)$ with $u_q(w/e) = 0$. Replacing u by u^k and v by $v(x)\varphi_j(y)$ in (18) we obtain the semi discrete variational problem:

$$\begin{array}{l} \sum_q \int_w^e [a_1\partial_x u_q\partial_x v + \cdots]dx \int_s^n \varphi_q\varphi_j dy... \\ +\sum_q \int_w^e u_q v dx \int_s^n [a_2\partial_y\varphi_q\partial_y\varphi_j + \cdots]dy \quad = \quad \int_R fv\varphi_j dxdy. \end{array} \tag{19}$$

We introduce the following notations:

$$\alpha^1(u, v) = \int_w^e (a_1u_xv_x + b_1u_xv + c_1uv)dx$$

$$\alpha^2(u, v) = \int_s^n (a_2u_yv_y + b_2u_yv + c_2uv)dy$$

$$f_j(x) = \int_s^n f\varphi_j dy, \beta^1(u, v) = \int_w^e uvdx, \beta^2(u, v) = \int_s^n uvdy$$

$$\beta_{jq} = \beta^2(\varphi_j, \varphi_q), \alpha_{jq} = \alpha^2(\varphi_j, \varphi_q).$$

Then the semi discrete variational problem (19) is: for all $q = 1, \cdots, J$ find u_q in $H_0^1(w, e)$ such that for all v in $H_0^1(w, e)$ and $j = 1, \cdots, J$

$$\sum_q [\beta_{jq}\alpha^1(u_q, v) + \alpha_{jq}\beta^1(u_q, v)] = \beta^1(f_j, v). \tag{20}$$

We use the generalized Fourier transform:

$$u_q(x) = \sum_{m=1}^{J} \hat{u}_m(x)\Phi_{mq}; \ q = 1, \cdots, J.$$

In order to obtain uncoupled problems we need a modified transform of the right hand side:

$$f_j = \sum_{m=1}^{J}\sum_{q=1}^{J} \beta_{jq}\Phi_{mq}\tilde{f}_m.$$

Substitution in equation (20) gives:

$$\sum_q \sum_m [\beta_{jq}\alpha^1(\hat{u}_m, v) + \alpha_{jq}\beta^1(\hat{u}_m, v) - \beta_{jq}\beta^1(\tilde{f}_m, v)]\Phi_{mq} = 0. \tag{21}$$

Choosing the Φ's as the eigenvectors of the spectral problem:

$$\sum_q \alpha_{jq}\Phi_{mq} = \lambda_m \sum_q \beta_{jq}\Phi_{mq}, m = 1, \cdots, J, \qquad (E')$$

gives to equation (21) the uncoupled form:

$$\alpha^1(\hat{u}_m, v) + \lambda_m \beta^1(\hat{u}_m, v) = \beta^1(\tilde{f}_m, v). \tag{22}$$

For the full FE approximation we introduce an x-mesh and a FE space with basis function $\theta_p,\ p = 1, \cdots, I$. We replace u_q by $\sum_p u_{pq}\theta_p$ and v by θ_i in (20). The generalized Fourier transform $u_{pq} = \sum_m \hat{u}_{pm}\Phi_{mq}$ using the eigenvectors of the spectral problem (E') gives the uncoupled discrete hat problem:

$$\forall m \in (1, .., J),\ \forall i \in (1, .., I),\ \sum_p [\alpha^1(\theta_p, \theta_i) + \lambda_m \beta^1(\theta_p, \theta_i)]\hat{u}_{pm} = \beta^1(\tilde{f}_m, \theta_i) \tag{23}$$

We are then back to the situation where we can apply the algorithm of Sect. 4, as it was the case for FD.

To illustrate this result, we consider the following numerical experiment with the homogeneous Dirichlet Poisson problem

$$-(u_{xx} + u_{yy}) = f,\ (x, y) \in \Omega_s,\ u_{\partial\Omega_s} = 0.$$

Ω_s is the square domain $(0, 1)^2$.

The function f is such that the exact solution is the polynomial

$$u_0(x, y) = 150x(x - 1)y(y - 1)(y - 1/2). \tag{24}$$

We use a Cartesian grid of Ω_s with 73×73 elements that is uniform in x, but *random* in y.

In Figure 3, we monitor the error in L_2 norm according to the number of sub-domains, and the number of modes that are accelerated. The space step in the y direction parallel to the interface is given in Figure 4. The ratio between the largest space step and smallest space step in the y direction is about 96. While the solver is in principle a direct solver, we notice that the error obtained with AS is of order 10^{-6}. A second iteration cycle of AS is needed to obtain the discrete solution with machine accuracy. We attribute this imperfect resolution that is not present with the original AS method on regular Cartesian mesh [11] to the inaccuracy of the computation of the eigenvectors that have high frequency components. As a matter of fact, if we apply the AS algorithm with an acceleration on the main half of the eigenvectors components only (i.e., $J' = \frac{J}{2}$) we obtain roughly the same level of accuracy after one cycle. It is only for the second cycle of AS that the differences between an acceleration of all components or half of them shows up.

Let us notice, however, that AS is fairly insensitive to the number of subdomains. This is one of the reasons why AS appears to be a scalable parallel linear solver [4], [9].

So far we have restricted ourselves to separable linear operators. In the next section we will address some of the difficulties in using AS to nonseparable and/or nonlinear elliptic operators.

VI. Nonseparable and Nonlinear Operators

We consider now an operator $L = L_1 + L_2$ that is *not* separable. We will focus on the following example,

$$L_1 = -\partial_{xx}u + c_1(x)u, L_2 = -\partial_{yy}u + c_2(x, y)u.$$

We introduce a full FD discretization and denote by Δ_2^h (resp.Δ_2^k) the approximation of ∂_{xx} (resp.∂_{yy}). Then an approximation of the equation $Lu = f$ is:

$$-\Delta_2^h u_{ij} + c_1(x_i)u_{ij} - \Delta_2^k u_{ij} + c_2(x_i, y_j)u_{ij} = f_{ij} \tag{25}$$

The generalized Fourier transform $u_{ij} = \sum_l \hat{u}_{il}\Phi_{lj}^i$ with a family of eigenvectors Φ_l^i indexed by i applied to this equation gives:

$$\sum_l [(-\Delta_2^h(\hat{u}_{il}\Phi_{lj}^i) + (c_1(x_i))\hat{u}_{il} - \hat{f}_{il})\Phi_{lj}^i] + \sum_l [-\Delta_2^k + c_2(x_i, y_j)]\Phi_{lj}^i\hat{u}_{ij} = 0 \tag{26}$$

To handle the second sum we introduce the i-indexed family of spectral problems:

$$[-\Delta_2^k + c_2(x_i, y_j)]\Phi^i_{mj} = \lambda^i_m \Phi^i_{mj}, m = 1, \cdots, J \qquad (E^i)$$

and choose the Φ^i_m's as the eigenvectors of (E^i). In the first sum we have (assuming for simplicity the h-mesh uniform):

$$-\Delta_2^h(\hat{u}_{il}\Phi^i_{lj}) = -1/h^2(\hat{u}_{i+1l}\Phi^{i+1}_{lj} + \hat{u}_{i-1l}\Phi^{i-1}_{lj} - 2\hat{u}_{il}\Phi^i_{lj}).$$

We multiply equation (26) by Φ^i_{mj} for a fixed m and sum over j. Using the orthogonality of the Φ^i_l for a fixed i and the notations $c^i_{lm} = \sum_j \Phi^{i-1}_{lj}\Phi^i_{mj}$ and $d^i_{lm} = \sum_j \Phi^{i+1}_{lj}\Phi^i_{mj}$, we obtain the coupled problem:

$$\begin{aligned} & -1/h^2(c^i_{mm}\hat{u}_{i-1m} + d^i_{mm}\hat{u}_{i+1m} - 2\hat{u}_{im}) \\ +(c_1(x_i) + \lambda^i_m)\hat{u}_{im} + \sum_{l\neq m} & -1/h^2(c^i_{lm}\hat{u}_{i-1l} + d^i_{lm}\hat{u}_{i+1l}) \quad = \quad \hat{f}_{im} \end{aligned} \qquad (27)$$

This shows that the hat unknowns $\hat{U}_m = (\hat{u}_{im})_i$ are coupled by tridiagonal matrices C_{lm} with a sub-diagonal of c^i_{lm} , a null diagonal, and an upper-diagonal of d^i_{lm}.

This coupling depends on how different are the Φ^i's. For example if there are only two different families of eigenvectors : Φ^1 for $i \leq \bar{i}$ and Φ^2 for $i \geq \bar{i}+1$ then the matrix C_{lm} has only two nonzero elements $c^{\bar{i}+1}_{lm}$ and $d^{\bar{i}}_{lm}$. Moreover the diagonal m-block has $d^{\bar{i}}_{mm}$ in position $(\bar{i}, \bar{i}+1)$ and $c^{\bar{i}+1}_{mm}$ in position $(\bar{i}+1, \bar{i})$.

To illustrate this situation, let us consider a problem with non separable operator, on the same Cartesian grid

$$-\Delta u(x,y) + c(x,y).u(x,y) = f(x,y), (x,y) \in \Omega_s,\ u_{\partial\Omega_s} = 0.$$

The rhs f is such that the exact solution is u_0 from (24).

We use two sub-domains, with an overlap from one to five meshes ; the function $c(.,.)$ depends only on the y-variable on each mid-part of the domain (here: $c(x,y) = 10y$ on the first mid-part of the domain, and $c(x,y) = 1$ on the second).

We use algorithm 3 to accelerate the Schwarz method, this time with the resolution of problem (E_i) in each sub-domain. We have then to solve one problem (E_i) per interface of the overlap (in our case: two problems (E_i)).

We observe that the error after only one acceleration is almost of order 10^{-4}, in comparison of the error of the Schwarz method, which is of order 10^{-2}. Note that the error after each acceleration decreases according to the size of the overlap (see Figure 5).

Further the same argument may be applied to more general second order operator

$$L_1 = -a_1(x)\partial_{xx}u + b_1(x)\partial_x u + c_1(x)u,\ L_2 = -a_2(x,y)\partial_{xx}u + b_2(x,y)\partial_x u + c_2(x,y)u,$$

if the approximation L^h_1 has coefficients depending on x but not on y.

We would like further to show numerically, that the nature of the coupling in (27) is fundamental. For flow in porous media, it is necessary to use fast solver for the operator $-div\,(a(x,y)\nabla u(x,y))$, with disparate scales for the size of a between geologic layers.

The simplest case corresponds to parallel geologic layer that can be aligned to the grid of discretization. To illustrate this situation we define the problem :

$$-div\,(a(x,y)\nabla u(x,y)) = f(x,y),\ (x,y) \in \Omega_s,\ u_{\partial\Omega_s} = 0,$$

with $a(.,.)$ as follows

$$a(x,y) = a_0 + (1-a_0)(1 + tanh((x - \frac{1}{2})/\epsilon))/2,\ a_0 = 10^1,\ \epsilon = 10^{-2}.$$

The function $a(x,y)$ exhibits a jump of order 10, across an interface aligned with the y direction in the middle of the domain. As shown in Figure 6, the AS method exhibits fast convergence with a minimum overlap.

In contrast, the convergence of AS becomes very poor if the function $a(.,.)$ is chosen with a jump not parallel to the interface. Figure 7 gives such evidence with

$$a(x,y) = a_0 + (1-a_0)(1 + tanh((x - (3h*y + 1/2 - h))/\epsilon))/2.$$

Because of the construction of the function $a(.,.)$, the problem becomes "strongly" nonseparable, i.e., the tridiagonal matrices C_{lm} cannot be neglected any longer.

The method described in this section is not cost effective in the general case. As a matter of fact the number of eigenvector problems to be solved is prohibitively large. We conclude this section with an illustration of the performance of the AS method when we use a priori an analytical approximation of the eigenvectors of the trace transfer operator.

The first example is a Poisson problem solved on an overset mesh (see Figure 8). The domain of computation is $\Omega_r \setminus D$, where Ω_r is the rectangle $(0, 2\pi) \times (0, \pi)$ and D is the disc of center (π, π) and radius $\pi/12$. We use a multiplicative Schwarz algorithm between two overlapping subdomains whose boundary fits respectively $\partial\Omega_r$ and ∂D. The mesh of the subdomain with the boundary $\partial\Omega_r$ is a subset of a regular Cartesian mesh denoted Ω_r^h. The mesh of the second subdomain denoted C^h is a regular mesh in polar coordinates (ρ, θ). The overlap between both subdomains in radial direction ρ is bounded by $2\ h$. The transmission conditions of Dirichlet type between the two subdomains are obtained by using the standard second order bilinear interpolation. Because the interpolation satisfies a maximum principle, we have a linear convergence of the multiplicative Schwarz algorithm. We accelerate the trace of the solution on the artificial interface ∂C^h by using $\exp(k2\pi i\theta),\ k \in Z$ as an approximation of the eigenvectors of the trace transfer operator. Figure 9 shows the decay of the error in maximum norm when one iterates on the cycle of AS accelerations. In this example the exact solution is $u(x, y) = \sin(2x) + \cos(2y)$. The Cartesian mesh is a subset of the regular grid 70×54. The annular subdomain has a regular mesh in polar coordinate (ρ, θ) with 20×48 grid points. This algorithm reaches the level of accuracy of the discrete overset solution after 5 iterations of the AS method. We have checked that the convergence speed is similar for larger overlaps. Let us notice, however, that the convergence rate may deteriorate when the disc boundary ∂D approaches the boundary of the rectangle $\partial\Omega$.

Our second and last example is for the weakly non linear problem

$$\Delta u + \lambda \exp(u) = f,\ (x, y) \in \Omega_s, u_{\partial\Omega_s} = 0. \tag{28}$$

This Bratu problem [25] is solved on a Cartesian grid using the approximation of the eigenvectors of the trace transfer operator corresponding to Δ^h, that is $\sin k\pi y,\ k \in Z$. We use a strip DD as in Section 3, and apply a Newton algorithm to solve each subdomain. The AS algorithm is therefore applied to the nonlinear problem directly. The AS cycle is repeated until convergence to the discrete solution. An alternative solution would be to use AS on the linearized operator itself, inside a global Newton loop [4].

We analyze the numerical efficiency of our iterative solver by counting the number of linear solves in each subdomain to reach convergence with an error in maximum norm that is less than a tolerance number tol. The interesting result is that the number of subdomain solves is fairly insensitive to the space step in y direction. The sensitivity of the number of iterations to reach convergence to the number of subdomains decays as the overlap get larger. Overall we have a very moderate growth of the number of iterations as the number of subdomains grows. This important property makes the AS method numerically relevant to achieve scalable performance on a parallel system. Figure 10 reports on the number of subdomain solves with respectively a 11×80 and 81×80 discretization grids, $\lambda = 6$ and $tol = 10^{-5}$. The stop criterium for the Newton algorithm is one order of magnitude smaller than tol. The overlap can be one or three mesh points in the x direction. This result is an average of twenty runs starting with a random positive initial condition of maximum norm 10^{-1}. The standard deviation is of order one to two iterations. We may have larger deviation if tol is close to the level of accuracy obtained after a given AS cycle. We have an average of 11 AS cycle to reach a 10^{-5} discrete error in maximum norm. The more cost effective solution corresponds to the larger number of subdomains that is used in parallel computation. Let us now summarize the conclusions of this paper.

VII. CONCLUSION

We have shown in this paper how to extend to general Cartesian mesh with arbitrary space steps in each direction, the so-called AS method first presented in [10] for FD discretization of elliptic operators on *regular* Cartesian grids. Results were presented for FD as well as FE approximations. This paper demonstrates that the AS method is not inherently dependent on the Fourier method, but rather on the possibilities to split the elliptic operator using a family of dominant eigenvectors. The trace-transfer operator can then be well approximated by a set of uncoupled

one dimensional operators. Therefore, the Aitken acceleration applied to the coefficient of the traces generated by the Schwarz method in this decomposition has high numerical efficiency.

This acceleration technique is not a linear technique by nature, and can be applied to nonlinear elliptic problems as well [13]. One may consider using this technique to build preconditioner to Krylov methods [16]. We preferably pursue the development of this method with an objective that to speed up existing CFD code with minimum changes in the code and allow efficient distributed computing with slow networks [4], [9] that are characteristic of affordable Beowulf clusters.

Acknowledgement : This work was supported in part by the NSF Grant ACI-0305405. We would like to thanks B.Smith and the PETSc team of Argonne Nat. Lab for their assistance in using the PETSc library.

REFERENCES

[1] A.Averbuch, M. Israeli and L.Vozovoi, *On Fast Direct Elliptic Solver by Modified Fourier Method*, Numer. Algorithms, Vol 15, pp 287-313, 1997.

[2] S.Balay, W.D.Gropp, L.C.McInnes, and B.F.Smith, *PETSc Users Manual*, Tech Rep. ANL-95/11, Revision 2.1.5, Argonne Nat. Lab. 2003.

[3] J. Baranger, M. Garbey and F.Oudin-Dardun, *Recent Development on Aitken-Schwarz Method*, 13th Int. Conf. on Domain Decomposition Methods DD13, Domain Decomposition Methods in Science and Engineering, CIMNE, Barcelona, N.Debit et Al edt, pp289-296, 2002.

[4] N. Barberou, M.Garbey, M.Hess, M. Resch, T. Rossi, J.Toivanen and D.Tromeur Dervout, *On the Efficient Meta-computing of linear and nonlinear elliptic problems*, Journal of Parallel and Distributed Computing- special issue on grid computing, Vol63, Issue 5, pp 564-577, 2003.

[5] I.Faille, *A Control Volume Method to Solve an Elliptic Equation on Two-dimensional Irregular Meshing*, Comp.Meth.Appl.Mech.Engrg.,100, pp.275-290, 1992.

[6] D. Funaro, A. Quarteroni, P. Zanolli, *An Iterative Procedure with Interface Relaxation for Domain Decomposition Methods*, SIAM J. Numer. Anal. 25, no. 6, pp 1213-1236, 1988.

[7] M. Gander, L. Halpern, F. Nataf, *Optimized Schwarz Methods*, Domain decomposition methods in sciences and engineering (Chiba, 1999), DDM.org, Augsburg, 1527 (electronic), 2001.

[8] M.Garbey, *Acceleration of the Schwarz method for elliptic problem*, to appear in SIAM J. of Scientific Computing.

[9] M.Garbey, B.Hadri and W.Shyy, *Fast Elliptic Solver for Incompressible Navier Stokes Flow and Heat Transfer Problems on the Grid*, , 43rd Aerospace Sciences Meeting and Exhibit Conference, Reno January 2005, AIAA-2005-1386, 2005.

[10] M. Garbey and D. Tromeur-Dervout, *Two level Domain Decomposition for Multiclusters*, Domain Decomposition in Sciences and Engineering, T. Chan & Al editors, published by DDM.org, pp325-339, 2001.

[11] M. Garbey and D. Tromeur-Dervout, *Aitken-Schwarz Algorithm on Cartesian Grid*, 13th Int. Conf. on Domain Decomposition Methods DD13, Domain Decomposition Methods in Science and Engineering, CIMNE, Barcelona, N.Debit et Al edt, pp53-66, 2002.

[12] M.Garbey and D.Tromeur Dervout, *Parallel Algorithms with Local Fourier Basis*, J. Comput. Phys., Vol 137, No 2, pp 575-599, 2001.

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

[14] P. Henrici, *Elements of Numerical Analysis*, John Wiley & Sons Inc, New York-London-Sydney, 1964.

[15] M.Israeli, E.Braverman and A.Averbuch, *A Hierarchical Domain Decomposition Method with Low Communication Overhead*, 13th Int. Conf. on Domain Decomposition Methods DD13, Domain Decomposition Methods in Science and Engineering, CIMNE, Barcelona, N.Debit et Al edt, pp395-404, 2002.

[16] D. E. Keyes, *How Scalable is Domain Decomposition in Practice?*, Proceedings of the 11th Intl. Conf. on Domain Decomposition Methods (C.-H. Lai, et al, eds.), pp. 286-297, 1998.

[17] Yu.A.Kuznetsov, *Overlapping Domain Decomposition Method for FE-problems with elliptic singular perturbed operators*, Fourth International Symposium on Domain ecomposition Method for PDEs (Moscow, 1990) SIAM Phladelphia, PA, pp223-241, 1991.

[18] P.L.Lions, *On the Schwarz Alternating Method I*, First International Symposium on Domain Decomposition Methods for Partial Differential Equations, Roland Glowinski and Gene H. Golub and Gérard A. Meurant and Jacques Périaux edt. SIAM, Philadelphia, PA, pp 1-42, 1988.

[19] P.L.Lions, *On the Schwarz Alternating Method II*, Second International Symposium on Domain Decomposition Methods for Partial Differential Equations, Tony Chan and Roland Glowinski and Jacques Périaux and Olof Widlund edt. SIAM, Philadelphia, PA, pp 47-70, 1989.

[20] T.Rossi and J.Toivanen, *A Parallel Fast Direct Solver for the Discrete Solver for Block Tridiagonal Systems with Separable Matrices of Arbitrary Dimension*. SIAM J.Sci.Comp., 20, 1778-1793, 1999.

[21] Y.Saad, *Iterative Methods for Sparse Linear Systems*, SIAM, 2nd ed., 2003.

[22] H.A.Schwarz, *Gesammelte Mathematische Abhandlungen*,2, 2nd ed.,, Springer Berlin, pp.133-143, 1980. *Vierteljahrsschrift der Naturforschenden Gesellschaf*, Zurich, Vol15, 1st ed. pp.272-286, 1870.

[23] B. Smith, P. Bjorstad and W.Gropp, *Domain Decomposition, Parallel Multilevel Methods for Elliptic Partial Differential Equations*, Cambridge University Press, 1996.

[24] J. Stoer and R. Burlish, *Introduction to Numerical Analysis*, TAM 12 Springer, 1980.

[25] C.Wieners, *A Parallel Newton Multigrid for High Order Finite Elements and its Application to Numerical Existence Proofs for Elliptic Boundary Value Problem*, Combustion Theory Modelling, Vol 76(3), pp 175-180, 1996.

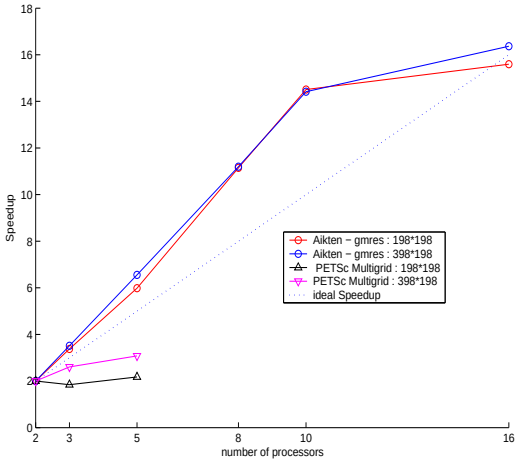

Fig. 1. Speedup between Aikten solved with GMRES and PETSc multigrid.

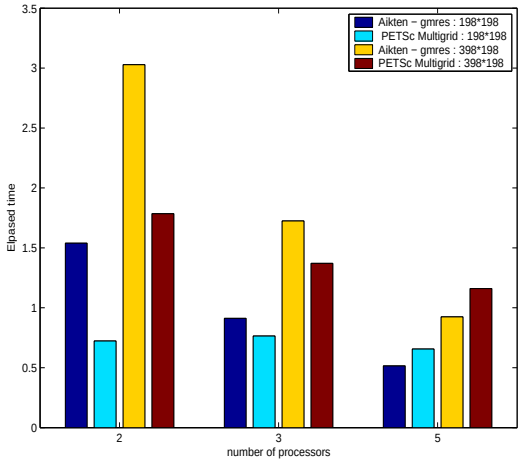

Fig. 2. Elapsed time for Aikten solved with GMRES in each subdomain and PETSc multigrid.

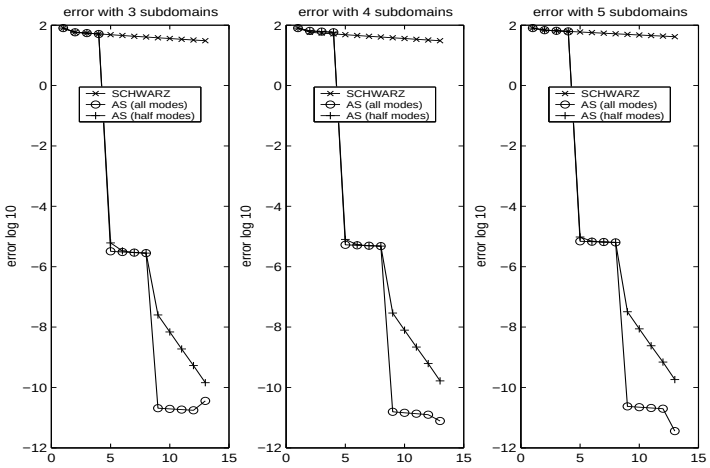

Fig. 3. Error in L_2 norm with, from left to right 3, 4 and 5 subdomains

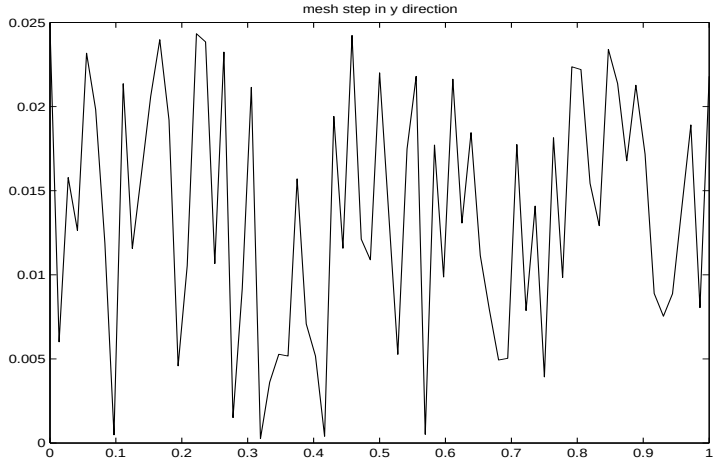

Fig. 4. Non uniform Cartesian mesh for the Poisson problem

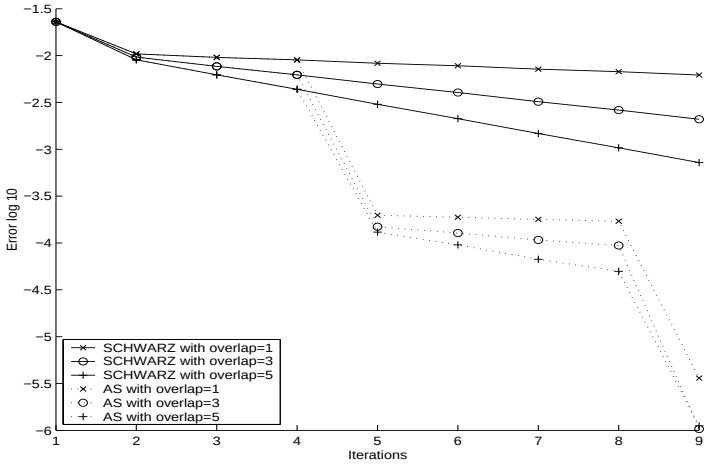

Fig. 5. Nonseparable operator : error with d size of overlap

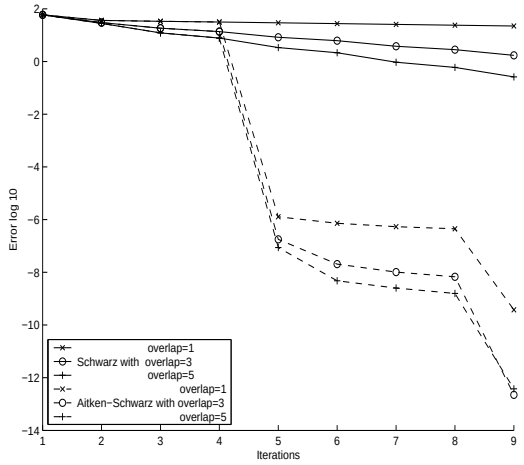

Fig. 6. $a(.,.)$ has a steep gradient parallel to the interface

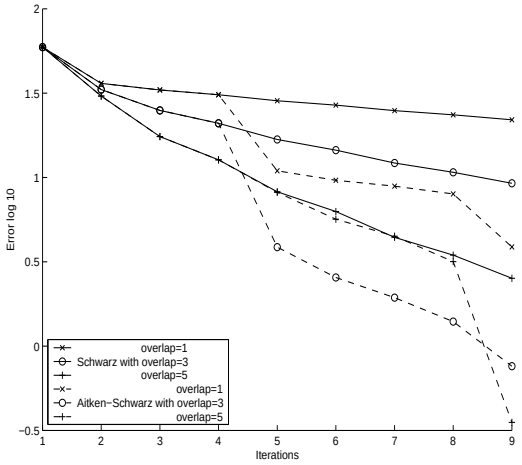

Fig. 7. $a(.,.)$ has a steep gradient that crosses the interface

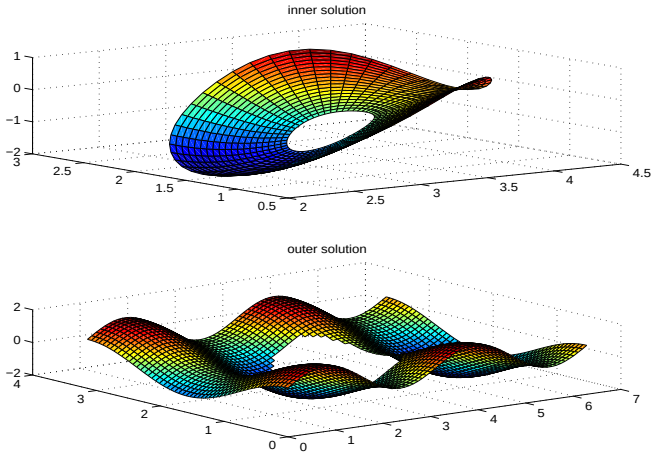

Fig. 8. Solution components on the overset mesh

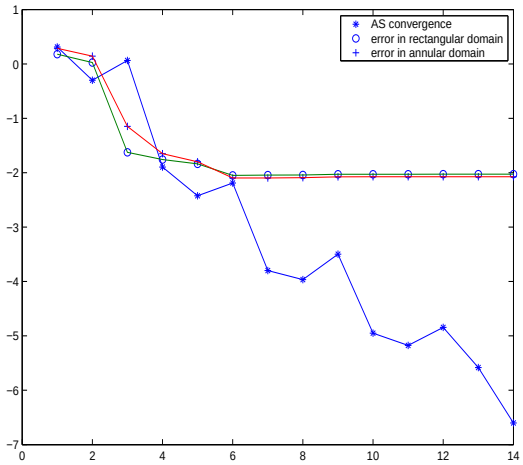

Fig. 9. Convergence history with the overset mesh.

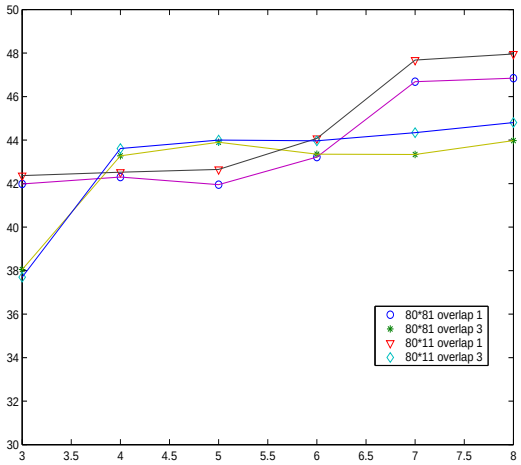

Fig. 10. Number of block solves for the Bratu problem.