A Direct Solver for the Heat Equation with Domain Decomposition in Space and Time

Marc Garbey

University of Houston, Computer Science (`http://www.cs.uh.edu/~garbey/`)

Summary. In this paper we generalize the Aitken-like acceleration method of the additive Schwarz algorithm for elliptic problems to the additive Schwarz waveform relaxation for the heat equation. The domain decomposition is in space and time. The standard Schwarz waveform relaxation algorithm has a linear rate of convergence and low numerical efficiency. This algorithm is, however, friendly to cache use and scales with the memory in parallel environments. We show that our new acceleration procedure of the waveform acceleration algorithm results in a fast direct solver.

1 Introduction

Currently, standard processors are becoming multi-cores and there is a strong incentive to make use of all these parallel resources while avoiding conflict in memory access. We also have an overwhelming abundance of parallel computers available when using grids. The Additive Schwarz (AS) method for elliptic problems or the Additive Schwarz Waveform Relaxation (ASWR) method for parabolic problems can be implemented easily in distributed computing environments and have very simple and systematic communication schemes. This algorithm is friendly to memory cache use and scales with the memory in parallel environments. ASWR in particular minimizes the number of messages sent in a parallel implementation and is very insensitive to delays due to a high latency network. The main drawback of the method is that it is one or several orders of magnitude slower than modern solvers such as multigrids. In the meantime, multigrids have poor parallel efficiency with high latency networks.

There have been two main classes of methods to speed up AS and ASWR. One is to introduce a coarse grid preconditioner. But a coarse grid operator reduces drastically the parallel efficiency on a slow network. A second option is to optimize the transmission conditions. This general avenue of work has been followed with success by numerous workers - see for example [3, 8, 9, 10] and their references. We have introduced in [7] a different and somehow complementary approach that consists of accelerating the sequence of trace on the interface generated by the AS method. The advantage of our postprocessing algorithm, besides its simplicity, is that it has quasi-optimum arithmetic complexity for the Poisson equation discretized on Cartesian grid while offering unique parallel efficiency on the grid. This is the only example, to our knowledge, of a numerically efficient Poisson solver that performs well on a grid of computers [2]. Our method offers also a general framework to speed up elliptic and non-linear elliptic solvers in a broad variety of conditions [1, 2, 6, 7].

Our main objective in this paper is to present an extension of this technique to the heat equation with Domain Decomposition (DD) in space *and* time. A generalization to Parabolic operators and its application to grid computing will be reported elsewhere [5].

2 Aitken-Schwarz Method for Linear Operators in One Space Dimension

The basic Aitken-Additive-Schwarz (AAS) method for linear elliptic problems can be found for example in [7]. Let us describe our AASWR algorithm for a domain decomposition in space *and* time with the following Initial Boundary Value Problem (IBVP):

$$\frac{\partial u}{\partial t} \;=\; L[u] \;+\; f(x,t),\; (x,t) \in \Omega = (0,1) \times (0,T), \tag{1}$$

$$u(x,0) \;=\; u_o(x),\; x \in (0,1), \tag{2}$$

$$u(0,t) = a(t),\; u(1,t) = b(t),\; t \in (0,T), \tag{3}$$

L is a second order linear elliptic operator. We assume that L coefficients are time independent and that the problem is well posed and has a unique solution.

We introduce the following discretization in space and time

$$0{=}x_0 < x_1 < ... < x_{N-1} < x_N{=}1,\; h_j{=}x_j - x_{j-1}, t_k{=}k\; dt,\; k{=}0 \dots M,\; dt{=}\frac{T}{M}.$$

Let us denote by X the column vector $X = (x_1, \dots, x_{N-1})^t$. A first order Euler implicit scheme in time writes

$$\frac{U^{k+1} - U^k}{dt} \;=\; D\, U^{k+1} \;+\; f(X, t^{k+1}),\; k = 0, .., M-1, \tag{4}$$

$$U^0 \;=\; u_o(X), U_0^{k+1} = a(t^{k+1}),\; U_N^{k+1} = b(t^{k+1}),\; k = 0, .., M-1, \tag{5}$$

where U^k is the column vector $U^k = (U_1^k, \dots, U_{N-1}^k)^t$. We also introduce the notation U_j for the row vector $U_j = (U_j^1, \dots, U_j^M)$.

D is a square matrix that comes from a finite difference or a finite element approximation for example. We do not need to specify this approximation. Our purpose is to compute efficiently the numerical solution of the discrete problem (4)-(5). At each time step one solves the linear system

$$(Id \;-\; dt\; D)U^{k+1} = F(U^k), \tag{6}$$

where Id is the matrix of the identity operator.

We assume that the matrix $A \;=\; Id \;-\; dt\; D$ of the linear system (6) is regular.

Introducing the matrices $U = (U^1, ..., U^M)$ and $F = (F(U^1), ..., F(U^M))$, we have

$$A\,U \ = \ F, \; U_0 = (a(t^1), \dots, a(t^M)), \; U_N = (b(t^1), ..., b(t^M)). \tag{7}$$

Let $\Omega_i = (y_i^l, y_i^r)$, $i = 1..q$, be a partition of Ω with

$$x_0 = y_1^l < y_2^l < y_1^r < y_3^l < y_2^r, \dots, y_q^l < y_{q-1}^r < y_q^r = x_N.$$

One iteration of the ASWR algorithm writes

```
$for\ i = 1..q, do$
    $A_i\, V_i^{n+1} \ = \ F_i,\ in\ \Omega_i \times (0,T),$
    $V_i^{n+1}(y_i^l) = V_{i-1}^n(y_i^l),\ V_i^{n+1}(y_i^r) = V_{i+1}^n(y_i^r),$
$enddo$
```

where A_i is the appropriate sub-block of A corresponding to the discretization of the IBVP problem in $\Omega_i \times (0,T)$. This algorithm generates a sequence of vectors $W^{k_s} = (V_2^{l,k_s}, V_1^{r,k_s}, V_3^{l,k_s}, V_2^{r,k_s}, \dots, V_q^{l,k_s})$ corresponding to the boundary values on the set

$$\mathcal{S} = (y_2^l, y_1^r, y_3^l, y_2^r, \dots, y_q^l, y_{q-1}^r) \times (t^1, ..., t^M)$$

of the V_i for each iterate k.

The proof of convergence of the additive Schwarz waveform relaxation on the continuous problem (1) with the heat equation given in [4] is based on the maximum principle. The convergence of the ASWR algorithm at the discrete level follows from a discrete maximum principle as well and apply for example to the classical three points finite difference scheme with the heat equation problem. Because the parabolic problem (1) is linear, the trace transfer operator

$$W^{k_s+1} - W^\infty \rightarrow W^{k_s} - W^\infty$$

is linear. Its matrix P has the following pentadiagonal structure:

$$\left| \begin{array}{ccccccccc} 0 & P_1^r & 0 & 0 & .... & & & & \\ P_2^{l,l} & 0 & 0 & P_2^{l,r} & ... & & & & \\ P_2^{r,l} & 0 & 0 & P_2^{r,r} & ... & & & & \\ & & & & & & & & \\ & & & & ... & P_{q-1}^{l,l} & 0 & 0 & P_{q-1}^{l,r} \\ & & & & ... & P_{q-1}^{r,l} & 0 & 0 & P_{q-1}^{r,r} \\ & & & & ... & 0 & 0 & P_q^l & 0 \end{array} \right| .$$

The block $P_i^{l,l}, P_i^{l,r}, P_i^{r,l}, P_i^{r,r}$ are square matrices of size $(M-1)^2$. If the matrix P is known and the matrix $Id - P$ is regular, one step of the ASWR provides enough information to reconstruct the exact interface values by solving the linear system

$$(Id - P) W^\infty \ = \ W^1 - P\, W^0. \tag{8}$$

We can then define Algorithm (I):

Step 1: compute the first iterate of ASWR.

Step 2: solve the linear problem (8).

Step 3: compute the second iterate using the exact boundary value W^∞.

We observe that this algorithm is a direct solver provided that $Id - P$ is regular, no matter the overlap, or the fact that ASWR converges or not. This method is a generalization of the Aitken-Schwarz algorithm described in [7] for the case of linear elliptic operators. We call *algorithm (I)* the Aitken-Additive Schwarz waveform relaxation algorithm. We have the following result [5].

Theorem 1. *If the ASWR algorithm converges, then AASWR is a direct solver.*

The construction of P is done using the following basis of functions

$$\delta_j^k = 1, \text{ if } j = k, \ 0 \text{ otherwise}, \ j, k \in \{1, .., M\}$$

to represent the trace of the solution on the interfaces

$$y_i^{l/r} \ \times \ \{t_1, ..., t_M\}, \ i = 1..q.$$

Let us consider the family of subproblems in $\Omega_i \times (0, T)$,

$$\frac{V_{i,j}^{k+1} - V_{i,j}^k}{dt} \ = \ D_i[V_{i,j}^{k+1}], \ k = 0, \ldots, M-1, \tag{9}$$

$$V_{i,j}^0 \ = \ 0, V_{i,j}^{k+1}(y_i^l) \ = \ 0, \ V_{i,j}^{k+1}(y_i^r) \ = \ \delta_j^{k+1}, \ k = 0, \ldots, M-1. \tag{10}$$

Let $V_{i,j}$ denote the matrix that is the solution of the discrete problem (9)-(10). The j column vector of $P^{r,r}$, respectively $P^{r,l}$, is the trace of $V_{i,j}$ on y_{i+1}^l, respectively y_{i-1}^r. $P_i^{r,r}$ and $P_i^{r,l}$ are consequently lower triangular matrices.

We notice that all $V_{i,j}$ are obtained from $V_{i,1}$ by a translation in time, i.e.,

$$V_{i,j}(X_i, t) \ = \ V_{i,1}(X_i, t - t_{j-1}), \ t \in \{t_j, \ldots, t_M\}, \tag{11}$$

and

$$V_{i,j}(X_i, t) \ = \ 0, \ t \in \{t_0, t_{j-1}\}. \tag{12}$$

The first column vector of $P^{r,r}$, respectively $P^{r,l}$, is the trace of $V_{i,1}$ on y_{i+1}^l, respectively y_{i-1}^r. From (11) we see that all columns of $P_i^{r,r}$, respectively $P_i^{r,l}$, are obtained from the first column of matrix $P_i^{r,r}$, respectively $P_i^{r,l}$, with no additional computation. To conclude, the construction of the matrix P of the trace transfer operator is achieved if one computes once and for all the solution of the two following sub-problems in $\Omega_i \times (0, T)$,

$$\frac{V_{i,j}^{k+1} - V_{i,j}^k}{dt} \ = \ D_i[V_{i,j}^{k+1}], \ k = 0, \ldots, M-1, \tag{13}$$

$$V_{i,j}^0 \ = \ 0, \ V_{i,j}^{k+1}(y_i^l) \ = \ \delta_1^{k+1}, \ V_{i,j}^{k+1}(y_i^r) \ = \ 0, \ k = 0, \ldots, M-1, \tag{14}$$

and

$$\frac{V_{i,j}^{k+1} - V_{i,j}^k}{dt} \ = \ D_i[V_{i,j}^{k+1}], \ k = 0, \ldots, M-1, \tag{15}$$

$$V_{i,j}^0 \ = \ 0, \ V_{i,j}^{k+1}(y_i^l) \ = \ 0, \ V_{i,j}^{k+1}(y_i^r) \ = \ \delta_1^{k+1}, \ k = 0, \ldots, M-1. \tag{16}$$

Remark 1. All sub-problems listed above needed for the construction of the trace transfer operator matrix can be solved with embarrassing parallelism.

We are going now to illustrate the method with the classical finite difference approximation for the one dimensional heat equation. The domain of computation is $(0, 1) \times (0, T)$. The grid has constant space step h and time step $dt = h$. We keep the number of grid points per sub-domain fixed with $N_b = 20$. Further the overlap is kept minimum, that is a one mesh interval. The Standard Method (SM) applies a direct tridiagonal solver to integrate each time step. The LU decomposition of the tridiagonal system can be computed once, since the same linear system is solved at every time step. The arithmetic complexity of the SM is then $n_1 = C_1\ N\ M$, where C_1 is an integer. $C_1 = 5$ for Gaussian elimination. The arithmetic complexity of one iterate of the ASWR algorithm is $n_q\ =\ C_1\ M\ (N + q - 1)$ which is asymptotically equivalent to n_1.

All subdomains correspond to the same finite difference operator. Consequently, the construction of the matrix P requires to solve one sub-domain problem (13)-(14) or (15)-(16). The arithmetic complexity of the construction of P is then $C_1\ M\ \frac{N+q-1}{q}$ and can be neglected against n_q. The acceleration step requires to solve the sparse linear system (8) uses asymptotically $n_{interface} = C_2 M[(q-1)^2 + O(q)]$ floating point operations (flops). $n_{interface}$ is small compare to n_q as long as $q << \sqrt{N}$.

Overall the number of flops for the AASWR procedure is about twice the number of flops for the standard SM with no DD. However modern computer architectures do not perform linearly with the number of flops. To illustrate this concept, we have performed the computation with both algorithm SM and AASWR on a PC running Matlab with a Pentium 4 2.66GHz. This PC has 1GB of main memory. With moderate number of time steps and large problem size, the advantage of the AASWR algorithms over the SM is clear. Figure 1 provides some comparison between both algorithm with ten time steps, i.e $M = 10$, $N_b = 20$ and a number of subdomains that varies from 2 to 20. The elapsed time is given in seconds and averages the measurement provided by one hundred runs. We remind here that the size of the problems grows linearly with the number of domains according to $N\ =\ N_b\ +\ (q-1)\ (N_b - 1)$. Overall the construction of P and the acceleration step has negligible elapse time. In AASWR the elapse time grows linearly with the number of subdomains. AASWR performs better than SM for $q > 6$. We believe that the cache size is responsible for the two peaks in the curve giving the performance of the SM. On the contrary the AASWR seems to be insensitive to the cache size for the dimension of the sub-domain that has been chosen here.

Figure 2 shows that the condition number of the matrix $(Id - P)$ used in the acceleration step grows linearly with the number of subdomains, which is proportional to the problem size in space N. However from our numerical experiments we have concluded that the acceleration procedure does not seems to impact significantly the accuracy of our exact solver.

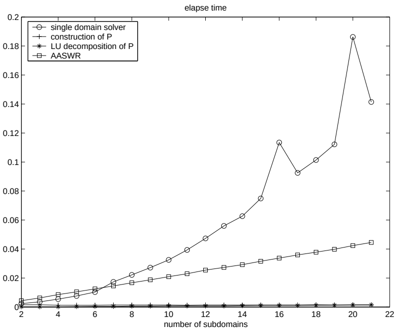

Fig. 1. Convergence of ASWR and AASWR for the heat equation.

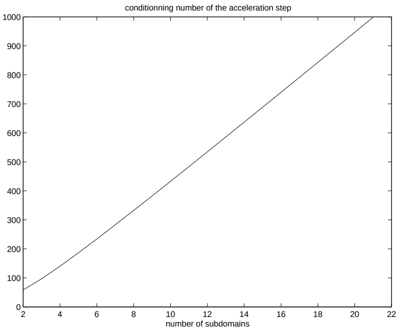

Fig. 2. Condition number of the linear system (8).

Most of the results obtained in this section can be extended to multi-dimensional parabolic problems provided L is separable or a weak perturbation of a separable operator [5].

3 Aitken-Schwarz Method for Linear Operators in the Multidimensional Case

To simplify the notations we will restrict ourselves to two space dimensions. We further assume that the domain Ω is a square discretized by a rectangular Cartesian grid with arbitrary space steps in each direction. Let us consider the IBVP:

$$\frac{\partial u}{\partial t} \;=\; L[u] \;+\; f(x,y,t),\ (x,y,t) \in \Omega = (0,1)^2 \times (0,T), \tag{17}$$

$$u(x,y,0) \;=\; u_o(x,y),\ (x,y) \in (0,1)^2, \tag{18}$$

$$u(0,y,t) = a(y,t),\ u(1,y,t) = b(y,t),\ y \in (0,1),\ t \in (0,T), \tag{19}$$

$$u(x,0,t) = c(x,t),\ u(x,1,t) = d(x,t),\ x \in (0,1),\ t \in (0,T), \tag{20}$$

where L is a second order linear elliptic operator. We assume that the problem is well posed and has a unique solution. Using an appropriate shift in space we can restrict ourselves to homogeneous Dirichlet boundary conditions.

The domain $\Omega = (0,1)^2$ is decomposed into q overlapping strips $\Omega_i = (y_i^l, y_i^r) \times (0,1)$.

We first present the general algorithm when L is a separable linear operator and refer to the theoretical framework established in [1] for elliptic operator:

$$L = L_1 + L_2,\ L_1 = e_1 \partial_{xx} + f_1 \partial_x + g_1,\ \ L_2 = e_2 \partial_{yy} + f_2 \partial_y + g_2.$$

e_1, f_1, g_1 are functions of x only, and e_2, f_2, g_2 are functions of y only. We write the discretized problem as follows

$$\frac{U^{k+1} - U^k}{dt} \;=\; D_{xx}[U^{k+1}] \;+\; D_{yy}[U^{k+1}] \;+\; f(X,Y,t^{k+1}),\ k = 0,\ldots,M-1, \tag{21}$$

with appropriate boundary conditions corresponding to (18)-(20).

Our main objective is to rewrite the discretized problem in such a way that we can reuse the results of Section 2 that is for the one space dimension case. Let us assume that D_{yy} has a family of $(N_y - 1)$ independent eigenvectors $\Phi_j,\ j = 1,..,N_y$ in $\mathbb{R}^{N_y - 1}$ with corresponding eigenvalues μ_j.

The Φ_j are implicitly the numerical approximation in $(0,1)$ of the solutions of the following continuous eigenvector problems:

$$L_2[v(y)] \;=\; \mu\, v(y),\ v(0) = v(1) = 0. \tag{22}$$

Let us introduce the decompositions

$$U^k(x,y,t) = \sum_{j=1}^{N_y-1} \Lambda_j^k(x,t)\Phi_j(Y), \qquad u_o(x,y) = \sum_{j=1}^{N_y-1} \lambda_j^k(x)\Phi_j(y),$$
$$f(x,y,t^k) = \sum_{j=1}^{N_y-1} f_j^k(x,t^k)\Phi_j(y), \qquad a(y,t^k) = \sum_{j=1}^{N_y-1} a_j(t^k)\Phi_j(y),$$
$$b(y,t^k) = \sum_{j=1}^{N_y-1} b_j(t^k)\Phi_j(y).$$

The discrete solution of (21) satisfies the following set of $(Ny-1)$ uncoupled problems

$$\frac{\Lambda_j^{k+1} - \Lambda_j^k}{dt} = D_{xx}[\Lambda_j^{k+1}] + \mu_j \, \Lambda^{k+1} + f_j(X,t^{k+1}), \; k = 0,\dots,M-1, (23)$$
$$\Lambda_j^0 = \lambda_j(X), \; \Lambda^{k+1}(x_0) = a_j(t^{k+1}), \; \Lambda^{k+1}(x_{N_x}) = b_j(t^{k+1}), \; k = 0,\dots,M-1. (24)$$

The trace transfer operator can be decomposed into (N_y-1) independent trace transfer operators

$$W_j^{k_s} - W_j^{\infty} \rightarrow W_j^{k_s+1} - W_j^{\infty},$$

that apply to each component of the trace of the solution expanded in the eigenvector basis $E = \{\Phi_j, \; j = 1,\dots,(N_y-1)\}$. Let Q_j be the matrix of this linear operator. The matrix P has now a (N_y-1) diagonal block structure, where each block is the matrix Q_j. The acceleration procedure of *Algorithm (I) Step 2* writes now

• Expand the trace of the solution in the eigenvector basis E and solve component wise

$$(Id - Q_j)W_j^{\infty} = W_j^1 - Q_j \, W_j^0, \; \forall j \in \{1,\dots,(Ny-1)\}. \qquad (25)$$

Assemble the boundary condition $W^{\infty} = \sum_{j=1,\dots,Ny-1} W_j^{\infty} \, \Phi_j$.

Let us emphasize that the sub-domain problems in $\Omega_j \times (0,T)$ can be integrated by any existing efficient numerical solver. It is only the acceleration step 2 that requires the decomposition of the *trace* of the solution into the eigenvector basis E. Because all eigenvector components of the solution are independents, we have then as in the one dimension space case:

Theorem 2. *If the ASWR algorithm converges, then AASWR is a direct solver.*

The construction of the Q_j can be done exactly as in the one space dimension case and can be computed with embarrassing parallelism.

This algorithm applies to the standard heat equation problem discretized in space on a five point stencil with central finite differences on a regular Cartesian mesh. Following the same steps as in Section 2.4, one can show that AASWR requires roughly two times as many floating point operations. But as stated before the AASWR algorithm is a parallel algorithm fairly tolerant to high latency networks. We have verified also that AASWR performs better than SM on a scalar processor with small number of time steps and large problem size.

We have verified also that the accuracy of our AASWR solver is satisfactory for three dimensional problem with singular source terms.

Remark 2. Our result can be easily generalized to tensorial products of a one dimensional grid with adaptive space stepping. The key hypothesis is the separability of the discrete operator $D_{xx} + D_{yy}$ on the tensorial product of grid. Because h_y is not a constant, the eigenvectors Φ_j are not known analytically and should be computed numerically as in [1].

Details of the parallel implementation of our method that are specific to space *and* time decomposition are reported in [5].

4 Conclusion

In this paper we have shown how to generalize the Aitken-like acceleration method of the additive Schwarz algorithm for elliptic problems to the additive Schwarz waveform relaxation for the heat equation. This new DD algorithm is in space *and* time. Since the concept of our acceleration technique is general and might be applied in principle to any block-wise relaxation scheme, we expect that it can be combined with some optimized transmission conditions for the same PDE problem. A further step in the development of our methodology would be to consider unstructured meshes, and approximate the trace transfer operator with for example, the coarse grid interface approximation presented in [6].

Acknowledgement. This work was supported in part by NSF Grant ACI-0305405.

References

[1] J. Baranger, M. Garbey, and F. Oudin-Dardun. On Aitken like acceleration of Schwarz domain decomposition method using generalized Fourier. *Domain Decomposition Methods in Science and Engineering*, pages 341 – 348, 2003.

[2] 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. *J. Parallel Distrib. Comput. (special issue on grid computing)*, 63:564 – 577, 2003.

[3] B. Després. Décomposition de domaine et problème de Helmholtz. *C. R. Acad. Sci. Paris Sér. I Math.*, 311(6):313–316, 1990.

[4] M. Gander and H. Zhao. Overlapping Schwarz wavefrom relaxation for the heat equation in n-dimensions,. *BIT*, 40(4):001 – 004, 2000.

[5] M. Garbey. Acceleration of a Schwarz waveform relaxation method for parabolic problems. Preprint UH-CS-06-11. Submitted.

[6] M. Garbey. Acceleration of the Schwarz method for elliptic problem. *SIAM J. Sci. Comput.*, 26(6):1871 – 1893, 2005.

[7] M. Garbey and D. T. Dervout. On some Aitken-like acceleration of the Schwarz method. *Internat. J. Numer. Methods Fluids*, 40(12):1493 – 1513, 2002.

[8] C. Japhet, F. Nataf, and F. Rogier. The optimized order 2 method. Application to convection-diffusion problems. *Future Generation Computer Systems FUTURE*, 18, 2001.

[9] V. Martin. An optimized Schwarz waveform relaxation method for unsteady convection-diffusion equation. *Appl. Numer. Math.*, 52(4):401 – 428, 2005.

[10] F. Nataf and F. Rogier. Factorization of the convection-diffusion operator and a (possibly) non overlapping Schwarz method. *M3AS*, 5(1):67 – 93, 1995.