Contemporary Mathematics
Volume 157, 1994

ADDITIVE SCHWARZ METHODS AND ACCELERATION WITH VARIABLE WEIGHTS

TSI-MIN SHIH CHIN-BO LIEM TAO LU

ABSTRACT. In section 1, we use Lions' framework to prove the convergence of a synchronous domain decomposition method for solving Dirichlet problems of linear uniformly elliptic equations, and to prove, under a more servere condition, the convergence will be geometrical. In section 2, we show that for the corresponding finite element solution the rate of convergence for the cases with generous overlap and minimal overlap are of $O((1+\ln H/h)^{-1})$ and of $O((1+ H/h)^{-1})$ respectively. In section 3, we prove that acceleration with variable weights will converge faster.

1. A domain decomposition method for overlapping subdomains

Consider the following Dirichlet problem

$$\begin{cases} Lu = f, & \text{in } \Omega \\ u = g, & \text{on } \partial\Omega \end{cases} \tag{1}$$

where $\Omega \subset R^N$ is a bounded open set,

$$Lu = -\sum_{i,j=1}^{N} \frac{\partial}{\partial x_i}(A_{ij}(x)\frac{\partial u}{\partial x_j}) + B(x)u,$$

1991 Mathematics Subject Classification. 65N06, 65N30, 65N55, 65N99, 65Y05, 65Y99.

The detailed version of this paper will be submitted for publication elsewhere.

Support by Hong Kong Polytechnic Research Sub-committee Grant 341/072.

$A_{ij}(x)$ satisfy the uniformly positive definite condition and $B(x) \geq 0$.

Let $\Omega = \bigcup_{i=1}^{m} \Omega_i$, here Ω_i, $i = 1, \ldots, m$, are overlapping opensets.

Algorithm 1 (Continuous case):

1° Choose an initial $u_0 \in H^1_g(\Omega)$.

2° Solve in parallel for $i = 1, 2, \ldots, m$,

$$(2) \qquad \begin{cases} Lu^i_n = f, & \text{in } \Omega_i \\ u^i_n = u_n, & \text{on } \partial\Omega_i \end{cases}$$

here $u^i_n - u_n \in H^1_0(\Omega_i)$ and set $u^i_n = u_n$ in $\Omega\backslash\bar{\Omega}_i$.

3° $u_{n+1} = \frac{1}{m} \sum_{i=1}^{m} u^i_n$. Set $n + 1 \Rightarrow n$ and go to 2°.

In order to prove the convergence, we make use of the Lions' interpretation of the Schwarz alternating method [1, 2]. Consider the bilinear form

$$a(u,v) = \int_\Omega \Big(\sum_{i,j=1}^{N} A_{ij} \frac{\partial u}{\partial x_i} \frac{\partial v}{\partial x_j} + B\, u\, v\Big)dx.$$

From the assumptions, $a(u,v)$ is uniformly elliptic in $H^1_0(\Omega)$, i.e., there exists a constant $\nu > 0$, such that

$$(3) \qquad \|u\|^2_a \triangleq a(u,u) \geq \nu \|u\|^2_{H^1_0(\Omega)}, \ \forall\, u \in H^1_0(\Omega).$$

Now the subproblem (2) implies for $i = 1, 2, \cdots, m$,

$$(4) \qquad a(u^i_n - u_n, v) = a(u - u_n, v), \ \forall\, v \in H^1_0(\Omega_i),$$

and

$$(5) \qquad u^i_n - u_n = P_i(u - u_n) .$$

Here $P_i : H^1_0(\Omega) \rightarrow H^1_0(\Omega_i)$ are the projection operators with respect to the energy norm $\|u\|^2_a$. Using (5), we deduce that

$$(6) \qquad u - u^i_n = (I - P_i)(u - u_n), \ i = 1, 2, \ldots, m,$$

and

(7) $u - u_{n+1} = (I - \frac{1}{m}\sum_1^m P_i)(u - u_n) = (I - \frac{1}{m}\sum_1^m P_i)^{n+1}(u - u_0)$.

Theorem 1.

(a). If

(8) $$H_0^1(\Omega) = \overline{H_0^1(\Omega_1) + H_0^1(\Omega_2) + \dots + H_0^1(\Omega_m)},$$

then u_n will converge to u, i.e., $\|u - u_n\|_a \to 0$, as $n \to \infty$.

(b). If

(9) $$H_0^1(\Omega) = H_0^1(\Omega_1) + H_0^1(\Omega_2) + \dots + H_0^1(\Omega_m),$$

then u_n will converge to u geometrically.

Remark 1. In step 3°, instead of setting $u_{n+1} = \frac{1}{m}\sum_{i=1}^m u_n$, we may set $u_{n+1} = \sum_{i=1}^m \omega_i u_n^i + (1-\omega)u_n$, where $\omega_i > 0$, $i = 1, 2, \dots, m$, and $\omega = \sum_{i=1}^m \omega_i < 2$. Evidently,

$$u - u_{n+1} = (I - \sum_{i=1}^m \omega_i P_i)(u - u_n),$$

and we still have $\|u - u_n\|_a \to 0$, as $n \to \infty$.

2. Finite element solutions with an estimate of the rate of convergence

Let $\Omega \subset R^2$ be a polygon. Decomposite Ω into m non-overlapping triangles Ω_i, $i = 1, \dots, m$. Denote H_i = diam Ω_i, $H = \max_i H_i$ and $\Omega^H = \{\Omega_i\}_{i=1}^m$.

Subdivide every Ω_i step by step and at last we have the refined triangulation Ω^h. For each i, construct a polygon $\Omega_i' \supset \Omega_i$. The vertices of Ω_i' are also the nodes of Ω^h.

2.1. $\{\Omega_i'\}_{i=1}^m$ has generous overlap. Assuming that H_i is of order H and that

(10) $$\rho(\partial\Omega_i'\setminus\partial\Omega, \partial\Omega_i\setminus\partial\Omega) \ge CH_i, \quad i = 1, 2, \dots, m,$$

here C is a constant independent of i.

Algorithm 2 (Finite element approximation with generous overlap):

1° Choose an initial u_0 satisfying $u_0 - g^I \in V^h$.

2° Solve the following m + 1 subproblems in parallel:

For i = 1, 2, ..., m, find u_n^i such that $u_n^i - u_n \in V_i^h$ and satisfies

(11) $$a(u_n^i , v^h) = (f, v^h), \ \forall v^h \in V_i^h .$$

3° Find u_n^0 such that $u_n^0 - R^I u^n \in V_0^h$ and satisfies

(12) $$a(u_n^0 ,v^h) = (f, v^h), \ \forall v^h \in V_0^h ;$$

here g^I is a linear interpolation of g on Ω^h and R^I is the linear interpolation operator which restricts u_n on Ω^H .

4° Set $u_{n+1} = \sum_{i=0}^{m} \omega_i u_n^i$, where $\omega_i > 0$ and $\sum_{i=0}^{m} \omega_i = 1.$

The following theorem can be proved by applying a theorem from Dryja and Widlund [3].

Theorem 2 . u_n converges to u^h and the rate of convergence is of $0\left(\left(1 + \ln \frac{H}{h}\right)^{-1}\right)$. Here the rate of convergence is defined to be $\tilde{\rho} \equiv -\ln \|I - \sum \omega_i P_i\|$.

2.2. $\{\Omega'_i\}_{i=1}^m$ **has minimal overlap.** Instead of (10), we obtain Ω'_i by adding to Ω_i the refined triangles next to Ω_i, therefore the width of the overlapping area will be of order h.

Algorithm 3 (Finite element approximation with minimal overlap):

Step 1° and 2° are the same as Algorithm 2.

Step 3° Set $u_{n+1} = \sum_{i=1}^{m} \omega_i u_n^i$, where $\omega_i > 0$ and $\sum_{i=1}^{m} \omega_i = 1.$

By the Theorem 3 of Dryja and Widlund [8], we can prove the following

Theorem 3. u_n converges to u^h and the rate of convergence is of $0\left(\left(1 + \frac{H}{h}\right)^{-1}\right)$.

3. Acceleration with variable weights

In the previous algorithms, the weights do not vary with the position. But in [4], we introduced an algorithm with weights varying with the position. Here we prove that this algorithm is better than Algorithm 1.

Suppose that $\Omega = \bigcup_{i=1}^{m} \Omega_i$, here $\{\Omega_i\}_{i=1}^{m}$ are overlapping sets. Let $\pi_m = \bigcap_{j=1}^{m} \Omega_j$. For $k = 1, 2, \cdots, m-1$, define π'_k be the set of ℓ multiple points ($\ell \geq k$), i.e., $Q \in \pi'$ if there exist $\ell \geq k$ subsets $\Omega_{i_1}, \Omega_{i_2}, \cdots, \Omega_{i_\ell}$ such that $Q \in \bigcap_{j=1}^{\ell} \Omega_{i_j}$. Now define $\pi_k = \pi'_k \backslash \mathrm{cl}(\bigcup_{j=k+1}^{m} \pi_j)$, π_k are open sets consist exactly of k- multiple points.

The algorithm given in [4] is as follows.

Algorithm 4 (Acceleration with variable weights):

Steps 1° and 2° are the same as Algorithm 1.

Step 3°. If $Q \in \pi_k$ and $Q \in \bigcap_{j=1}^{k} \Omega_{i_j}$, then set

$$\bar{u}_{n+1}(Q) = \frac{1}{k} \sum_{j=1}^{k} u_n^{i_j}(Q) .$$

Theorem 4. Assume that $H_0^1(\Omega) = H_0^1(\Omega_1) + \ldots + H_0^1(\Omega_m)$. Algorithm 4 is better than Algorithm 1 with respect to the energy norm $\|u\|_a^2 = a(u,u)$.

Remark 2. From Lions' Theorem 4 [2], if u_0 is a subsolution or supersolution, then it is easy to prove that $\bar{e}_n(Q)$ approaches zero faster than $e_n(Q)$ does.

The algorithms are tested by examples and we find that all the results fit the theorems very well.

REFERENCES

1. P.L. Lions: On the Schwarz Alternating Method I. SIAM Proceedings of the First International Symposium on Domain Decomposition Methods for Partial Differential Equations. Philadelphia, 1988, PP. 1-42.
2. P.L. Lions, On the Schwarz Alternating Method II: Stochastic Interpretation and Order Properties. 'Domain Decomposition Methods', SIAM Proceedings of the Second International Symposium on Domain Decomposition Methods, Philadelphia, 1989, PP. 47-70.
3. M. Dryja: An Additive Schwarz Algorithm for Two- and Three-Dimensional Finite Element Elliptic Problems. 'Domain Decomposition Methods', SIAM Proceedings of the Second International Symposium on Domain Decomposition Methods, Philadelpha, 1989, PP. 168-172.
4. T. Lu, T.M. Shih and C.B. Liem: Parallel Algorithms for Solving Partial Differential Equations. 'Domain Decomposition Methods', SIAM Proceedings of the Second International Symposium on Domain Decomposition Methods, Philadelphia, 1989, PP. 71-80.
5. G.I. Marchuk: Method of Numerical Mathematics, 2nd ed., Springer-Verlag, 1982.
6. L.A. Hageman and D.A. Young, Applied Iterative Methods, Acad. Press, 1981.
7. T. Lu, T.M. Shih and C.B. Liem, Domain Decomposition Methods - New Numerical Techniques for Solving Partial Differential Equations (in Chinese), Science Press, Beijing, China, 1992.
8. M. Dryja and O. Widlund: Domain Decomposition Algorithms with Small Overlap, presented at the Sixth International Symposium on Domain Decomposition Methods in Science and Engineering, Como, Italy, 15-19 June, 1992.

Department of Applied Mathematics, Hong Kong Polytechnic, Kowloon, Hong Kong (T.M. Shih and C.B. Liem).

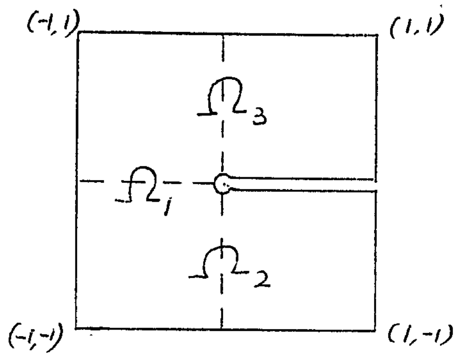
E-mail address: RAHOSHIH@HKPCC.HKP.HK (T.M. Shih and C.B. Liem)

Institute of Mathematical Science, Academia Sinica, Chengdu Branch, Chengdu, China (T. Lu).

Example 1

$$\begin{cases} -\Delta u = 0, & \Omega, \\ u = \theta = \tan^{-1}(y/x), & \partial\Omega, \end{cases}$$

Exact $u = \theta$.

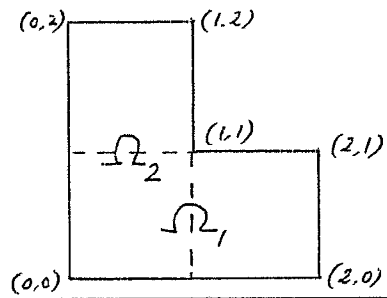


h = grid size	n = #ITER		$\lVert u_{n+1}-u_n \rVert_\infty$		CPU(sec)		E_{n+1}/E_n		ROC≈-ln(E_{n+1}/E_n)		ROC*(1+ln(H/h))	
	Algo I	Algo III	Algo I	Algo III	Algo I	Algo III	Algo I	Algo III	Algo I	Algo III	Algo I	Algo III
1/4	40	10	9.9E-6	9.5E-6	7.9	2.0	.743	.215		1.54		4.38
1/5	43	11	7.6E-6	9.1E-6	24.8	6.8	.755	.249		1.39		4.27
1/6	44	12	9.2E-6	6.2E-6	70.5	18.6	.764	.275		1.29		4.26
1/7	46	13	8.0E-6	4.4E-6	189	53.7	.771	.284		1.26		4.29
1/8	47	13	8.7E-6	8.3E-6	369	112	.777	.291		1.23		4.35

Example 2

$$\begin{cases} -\Delta u + u = 1, & \Omega, \\ u = 1, & \partial\Omega, \end{cases}$$

Exact $u = 1$.

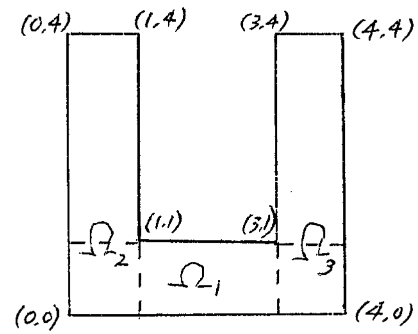


h = grid size	n = #ITER		$\lVert u_n - u \rVert_\infty$		CPU(sec)		E_{n+1}/E_n		ROC≈-ln(E_{n+1}/E_n)		ROC*(1+ln(H/h))	
	Algo I	Algo III	Algo I	Algo III	Algo I	Algo III	Algo I	Algo III	Algo I	Algo III	Algo I	Algo III
1/4	27	12	9.2E-9	2.3E-9	1.67	1.21	.500	.148		1.91		7.58
1/6	28	13	5.0E-9	1.6E-9	9.11	8.09	.500	.192		1.65		7.22
1/8	28	14	5.4E-9	2.0E-9	56.9	55.6	.500	.220		1.51		7.04
1/10	28	14	5.8E-9	5.9E-8	196	217	.500	.240		1.43		6.99

Example 3

$$\begin{cases} -\nabla\left(\left(1+\frac{x}{2}+\frac{y}{3}\right)\nabla u\right) = f, & \Omega, \\ u = g, & \partial\Omega, \end{cases}$$

Exact u = sin x sin y .

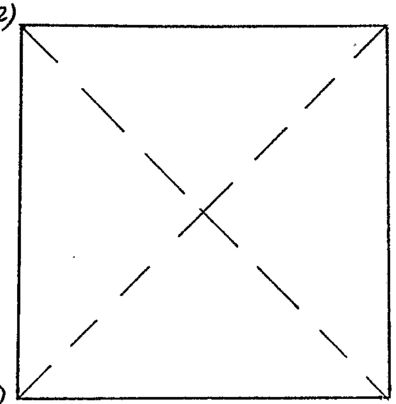


h = grid size	n = #ITER		$\|u_{n+1}-u_n\|_\infty$		CPU(sec)		E_{n+1}/E_n		ROC=-ln(E_{n+1}/E_n)		ROC*(1+ln(H/h))	
	Algo I	Algo III	Algo I	Algo III	Algo I	Algo III	Algo I	Algo III	Algo I	Algo III	Algo I	Algo III
1/4	52	12	8.6E-9	2.7E-9	17.0	4.15	.721	.162		1.82		9.65
1/5	54	13	8.3E-9	2.4E-9	66.2	18.5	.729	.190		1.66		9.17
1/6	55	13	9.7E-9	7.4E-9	206	60.0	.736	.207		1.58		9.02

Example 4

$$\begin{cases} -\Delta u = 4(x+y)-2(x^2+y^2), & \Omega, \\ u = 0, & \partial\Omega, \end{cases}$$

Exact $u = xy(x-2)(y-2)$.



h = grid size	n = #ITER		$\lVert u_n - u \rVert_\infty$		CPU(sec)		E_{n+1}/E_n		ROC≈-ln(E_{n+1}/E_n)		ROC*(H/h)	
	Algo II	Algo III	Algo II	Algo III	Algo II	Algo III	Algo II	Algo III	Algo II	Algo III	Algo II	Algo III
1/6	100	51	1.0E-5	6.4E-11	260	127	.895	.626	.111	.468	1.33	5.62
1/9	100	39	4.6E-4	8.5E-6	1127	506	.928	.734	.0747	.309	1.34	5.56
1/12	100	42	3.2E-3	8.7E-5	3307	1360	.945	.794	.0566	.231	1.36	5.54
1/15	28	22	2.7E-1	2.4E-2	2261	1695	.956	.832	.045	.184	1.35	5.52

Example 5

$$\begin{cases} -\Delta u = 2\pi^2 \,\mathrm{Sin}(\pi x)\mathrm{Sin}(\pi y)\,, & \Omega\,, \\ u = 0\,, & \partial\Omega\,, \end{cases}$$

Exact $u = \mathrm{Sin}(\pi x)\mathrm{Sin}(\pi y)$.

h = grid size	n = #ITER		$\lvert u_n - u \rvert_\infty$		CPU(sec)		E_{n+1}/E_n		ROC≈-ln(E_{n+1}/E_n)		ROC*(H/h)	
	Algo II	Algo III	Algo II	Algo III	Algo II	Algo III	Algo II	Algo III	Algo II	Algo III	Algo II	Algo III
1/6	100	50	9.7E-6	9.7E-11	257	128	.895	.626	.111	.468	.666	2.81
1/9	100	46	4.4E-4	9.4E-6	71106	524	.928	.734	.0747	.309	.672	2.78
1/12	100	62	3.1E-3	8.3E-7	3356	2051	.945	.794	.0566	.231	.679	2.77
1/15	30	22	2.4E-1	2.3E-2	2418	1664	.956	.832	.045	.184	.675	2.76