Available online at www.sciencedirect.com
SCIENCE DIRECT®
C. R. Acad. Sci. Paris, Ser. I 339 (2004) 145–150
COMPTES RENDUS MATHEMATIQUE

Numerical Analysis

Simultaneous directions parallel methods for elliptic and parabolic systems

José R. Galo [a], Isidoro I. Albarreal [b], M. Carmen Calzada [a], José Luis Cruz [a], Enrique Fernández-Cara [b], Mercedes Marín [a]

[a] *Dpto. Informática y Análisis Numérico, Universidad de Córdoba, Campus de Rabanales, Ed. C2-3, 14071 Córdoba, Spain*
[b] *Dpto. E.D.A.N., Universidad de Sevilla, Aptdo. 1160, 41080 Sevilla, Spain*

Received 20 April 2004; accepted 10 May 2004

Available online 17 June 2004

Presented by Roland Glowinski

Abstract

Adapting some previous ideas by Lu, Neittaanmäki and Tai, we use simultaneous directions methods to solve linear second-order elliptic and parabolic equations. The task is reduced to solving a (large) family of independent one-dimensional linear systems. We indicate some convergence results. We also present some numerical experiments. ***To cite this article: J.R. Galo et al., C. R. Acad. Sci. Paris, Ser. I 339 (2004).***

Résumé

Méthodes parallèles de directions simultanées pour des équations elliptiques et paraboliques. On utilise des méthodes parallèles de type directions simultanées pour résoudre des équations linéaires elliptiques et paraboliques du second ordre. On adapte quelques idées de Lu, Neittaanmäki and Tai, ce qui conduit à une (grande) famille de systèmes différentiels ordinaires qui sont indépendants. Quelques résultats de convergence et stabilité sont indiqués. On présente aussi quelques expériences numériques. ***Pour citer cet article : J.R. Galo et al., C. R. Acad. Sci. Paris, Ser. I 339 (2004).***

Version française abrégée

On considère d'abord le problème elliptique linéaire

$$\begin{cases} -\beta\Delta u + \alpha u = f & \text{dans } \Omega, \\ u = g & \text{sur } \partial\Omega, \end{cases} \tag{1}$$

où $\Omega \subset \mathbf{R}^d$ est un domaine borné régulier, $\beta > 0$, $\alpha \geqslant 0$ et les fonctions f et g sont données.

E-mail addresses: ma1gasaj@uco.es (J.R. Galo), iignacio@us.es (I.I. Albarreal), ma1canam@uco.es (M.C. Calzada), jlcruz@uco.es (J.L. Cruz), cara@us.es (E. Fernández-Cara), merche@uco.es (M. Marín).

1631-073X/$ – see front matter
doi:10.1016/j.crma.2004.05.007

Dans [5], Lu et al. ont proposé un schéma parallèle pour ce problème. On part de la décomposition (4), on introduit les paramètres $\tau, \omega > 0$, et on fait le calcul de U^{m+1} à partir de U^m comme suit : D'abord, on calcule les $U^{m+1,n}$, $1 \leqslant n \leqslant d$, en résolvant les problèmes 1D (5) (complétés par des conditions au bord appropriées) ; ensuite, on prend U^{m+1} comme dans (6).

On trouve donc une méthode où les étapes intermédiaires (5) peuvent être résolues en parallèle. On dit qu'il s'agit d'une méthode SDI.

Supposons que le problème (1) a été approché au sens habituel des différences finies centrées dans le domaine discrétisé Ω_h. Au lieu des opérateurs A_n on a maintenant des matrices carrées de valeurs propres $\lambda_{k,n}$ avec $0 \leqslant \mu \leqslant \lambda_{k,n} \leqslant \nu$ pour tout n et k. Introduisons les fonctions ψ and $\gamma_{\mu,\nu}$ données par (7). Alors on a (8).

Théorème 0.1. *L'algorithme* (5), (6) *satisfait les propriétés suivantes* :

(i) *Pour chaque $\tau > 0$, il existe $\omega_\tau > 0$ tel que la méthode converge si $0 < \omega < \omega_\tau$.*
(ii) *Pour chaque $\omega > 0$, il existe $\tau_\omega > 0$ tel que la méthode converge si $0 < \tau < \tau_\omega$.*

De plus, l'algorithme converge pour $(\omega, \tau) \in (0, 2d/c_{\mu,\nu}] \times (0, +\infty)$, où $c_{\mu,\nu}$ est défini dans (8).

Pour la démonstration de ce résultat, quelques extensions et une analyse des *meilleures valeurs* des paramètres ω et τ, voir [2] et [3].

On considère maintenant le système parabolique linéaire

$$\begin{cases} \dfrac{\partial u}{\partial t} - \beta \Delta u + \alpha u = f & \text{dans } \Omega \times (0, T), \\ u = g & \text{sur } \partial\Omega \times (0, T), \\ u(x, 0) = u_0(x) & \text{dans } \Omega, \end{cases} \tag{2}$$

où f, g et u_0 sont à nouveau données.

Pour la résolution numérique de (2), on introduit d'abord une approximation en espace (encore au sens des différences finies centrées). Ceci conduit au problème de Cauchy différentiel ordinaire (11). Dans une deuxième étape, on discrétise en temps. Ainsi, on introduit une partition uniforme de pas $k = \Delta t$ de l'intervalle $[0, T]$ et les $2d + 1$ quantités δ_n pour $1 \leqslant n \leqslant d$ et ω_n pour $0 \leqslant n \leqslant d$. L'algorithme est maintenant le suivant :

(a) On prend $U^0 = u_{0h}$.
(b) Pour $m \geqslant 0$ et U^m donnés, on calcule les $U^{m+1,n}$ pour $1 \leqslant n \leqslant d$ en résolvant les problèmes 1D (12) où $f^m = f_h((m + \frac{1}{2})k)$ et, finalement, on prend U^{m+1} comme dans (13).

Encore une fois, les systèmes (12) peuvent être résolus en parallèle et on trouve une méthode SDI.

Théorème 0.2. *On suppose que les δ_n et ω_n sont $\geqslant 0$.*

(1) *Si $k \leqslant 2/d^2\nu$, l'algorithme* (12), (13) *est stable pour toutes les valeurs des δ_n. Sinon, il est stable si $\delta_n \geqslant \eta_1$ avec $\eta_1(k, \mu, \nu) > 0$.*
(2) *Si les $\delta_n \geqslant c_{\mu,\nu}/2$, l'algorithme est stable pour tout $k > 0$. Sinon, il est stable si $k \in (0, \eta_2)$ avec $\eta_2(\mu, \nu) > 0$.*

Théorème 0.3. *La méthode* (12), (13) *converge avec erreur globale* $O(h^2 + k)$ *si les paramètres δ_n et ω_n satisfont $\omega_0 + \sum_{n=1}^{d} \omega_n = 1$ et $\sum_{n=1}^{d} \omega_n \delta_n = 1$ et les conditions de stabilité du Théorème* 0.2. *D'autre part, l'erreur est* $O(h^2 + k^2)$ *si on a aussi $\omega_n \delta_n^2 = 1/2$ pour tout $n = 1, \ldots, d$.*

Pour les démonstrations de ces résultats et quelques extensions, voir [2] et [4]. Pour quelques résultats numériques, voir [2–4] et la version anglaise de cette Note.

1. The simultaneous directions method for linear elliptic systems

Let us first consider the following linear partial differential problem of the elliptic kind:

$$\begin{cases} -\beta\Delta u + \alpha u = f(x) & \text{in } \Omega, \\ u = g(x) & \text{on } \partial\Omega. \end{cases} \tag{3}$$

Here and in the sequel, $\Omega \subset \mathbf{R}^d$ is a bounded regular domain, $\beta > 0$, $\alpha \geqslant 0$ and the functions f and g are given. In [5], Lu et al. proposed the following parallel fractional scheme for the solution of (3). Starting from the decomposition

$$A = A_1 + \cdots + A_d, \qquad A_n = \frac{\alpha}{d} I - \beta \frac{\partial^2}{\partial x_n^2} \tag{4}$$

and introducing the parameters $\omega, \tau > 0$, the passage from U^m to U^{m+1} is made as follows: we first compute $U^{m+1,n}$, $1 \leqslant n \leqslant d$, by solving numerically the 1D problems

$$(I + \tau A_n)U^{m+1,n} = \left(I - \tau \sum_{k=1,\, k\neq n}^{d} A_k \right) U^m + \tau b, \quad 1 \leqslant n \leqslant d \tag{5}$$

(completed with appropriate boundary conditions); then, set

$$U^{m+1} = \frac{\omega}{d} \sum_{n=1}^{d} U^{m+1,n} + (1-\omega)U^m. \tag{6}$$

This is called a *simultaneous directions implicit* (SDI) method. Notice that we get here, at least, two levels of parallelization: the first one corresponds to the d simultaneous steps (5); the second level corresponds to the simultaneous solution, in each fractional substep, of a set of discrete 1D problems.

In order to clarify the behavior of (5), (6), it seems interesting to carry out a detailed analysis of the influence of the values of the *evolution parameter* τ and the *coordination parameter* ω, similar to the one made by Marchuck in [6] for standard (sequential) fractional step methods.

To fix ideas, it will be assumed in the sequel that we have chosen a mesh of size $h > 0$ and we have replaced the partial derivatives in Ω by centered finite differences in the discrete domain Ω_h in the usual way. Consequently, we will be actually dealing with squared matrices A and A_n with eigenvalues $\lambda_{k,n}$ satisfying $0 \leqslant \mu \leqslant \lambda_{k,n} \leqslant \nu$ for all n and k. Let ψ and $\gamma_{\mu,\nu}$ be given by

$$\psi(x) = \left(\sum_{i=1}^{d} x_i \right) \left(\sum_{i=1}^{d} \frac{1}{1+x_i} \right), \qquad \gamma_{\mu,\nu}(\tau) = \max_{[\mu,\nu]^d} \psi(\tau x). \tag{7}$$

Then $\gamma_{\mu,\nu} : \mathbf{R}_+ \mapsto \mathbf{R}$ is strictly positive, continuous and strictly increasing and satisfies

$$\lim_{\tau\to 0} \gamma_{\mu,\nu}(\tau) = 0, \qquad \lim_{\tau\to+\infty} \gamma_{\mu,\nu}(\tau) = c_{\mu,\nu} > d^2. \tag{8}$$

Theorem 1.1. *The algorithm* (5), (6) *satisfies the following*:

(i) *For any $\tau > 0$, there exists $\omega_\tau > 0$ such that the method is convergent whenever $0 < \omega < \omega_\tau$.*
(ii) *For any $\omega > 0$, there exists $\tau_\omega > 0$ such that the method is convergent whenever $0 < \tau < \tau_\omega$.*

Furthermore, the algorithm converges for any $(\omega, \tau) \in (0, 2d/c_{\mu,\nu}] \times (0, +\infty)$.

Remark 1. For usual *alternating directions implicit* (ADI) methods, it is possible to determine a family of parameters that is independent of the problem, that is to say of μ and ν, for which convergence is ensured. This is not possible here (it seems natural that we pay some cost for parallelization).

An analysis of the values of ω and τ that make the previous parallel fractional step method converge faster is carried out in [2] and [3]. In this reference it is shown that under the assumption

$$\frac{4d}{h^2}\frac{\beta}{\alpha} \leqslant \frac{1+\sqrt{5}}{2} \tag{9}$$

the asymptotic convergence speed of the method (5), (6) applied to (3) is approximately 0.75. This means that three iterations are needed in order to reduce the initial error by a factor $1/10$. Although (9) can look like very restrictive, it is verified in some interesting situations. For example, for the application of the parallelization algorithm to the solution of time-dependent problems, the most interesting situations are those for which β is small (β is essentially the diffusion coefficient) and α is large (α is essentially the inverse of the time discretization step). For more details, see [1] and [2].

2. The method for linear parabolic problems

We will now consider the linear parabolic system

$$\begin{cases} \dfrac{\partial u}{\partial t} - \beta\Delta u + \alpha u = f & \text{in } \Omega \times (0,T), \\ u = g & \text{on } \partial\Omega \times (0,T), \\ u(x,0) = u_0(x) & \text{in } \Omega, \end{cases} \tag{10}$$

where f, g and u_0 are again given functions.

For the numerical solution of (10), we first approximate in the spatial variable $x = (x_1, \ldots, x_d)$, for instance introducing again a finite difference approximation. This leads to a Cauchy problem for an ordinary differential system in $\Omega_h \times (0,T)$:

$$\begin{cases} \dfrac{\mathrm{d}u_h}{\mathrm{d}t} + Au_h = f_h(t) & \text{in } (0,T), \\ u_h(0) = u_{0h}. \end{cases} \tag{11}$$

In a second step, problem (11) is discretized in the time variable t. Thus, let us introduce a uniform partition, with step $k = \Delta t$, of the time interval $[0,T]$ and the $2d+1$ positive numbers δ_n for $1 \leqslant n \leqslant d$ and ω_n for $0 \leqslant n \leqslant d$ (the evolution and coordination parameters in this case). The algorithm is now the following:

(a) Take $U^0 = u_{0h}$.
(b) For given $m \geqslant 0$ and U^m (the approximation of $u_h(mk)$), compute $U^{m+1,n}$ for $1 \leqslant n \leqslant d$ by solving the 1D problems

$$(I + k\delta_n A_n)U^{m+1,n} = \left(I - k\delta_n \sum_{j=1,\, j\neq n}^{d} A_j\right)U^m + k\delta_n f^m, \tag{12}$$

where $f^m = f_h((m+\frac{1}{2})k)$ and finally set

$$U^{m+1} = \sum_{n=1}^{d} \omega_n U^{m+1,n} + \omega_0 U^m. \tag{13}$$

At each time step, the systems (12) are mutually independent and consequently can be solved in parallel. Accordingly, this is again a SDI method.

Theorem 2.1. *Assume that all the parameters δ_n and ω_n are non-negative. Then the SDI method* (12), (13) *satisfies the following*:

(1) *If $k \leqslant 2/d^2\nu$ the method is stable for any δ_n. Otherwise, it is stable whenever $\delta_n \geqslant \eta_1$ for some $\eta_1(k, \mu, \nu) > 0$.*
(2) *If all $\delta_n \geqslant c_{\mu,\nu}/2$ the method is stable for any positive k. Otherwise, the method is stable for $k \in (0, \eta_2)$, for some $\eta_2(\mu, \nu) > 0$.*

Theorem 2.2. *The SDI method* (12), (13) *is convergent, with global error $O(h^2 + k)$, whenever the evolution and coordination parameters satisfy $\omega_0 + \sum_{n=1}^{d} \omega_n = 1$, and $\sum_{n=1}^{d} \omega_n \delta_n = 1$ and the stability conditions of Theorem* 2.1 *hold. Furthermore, it converges with global error $O(h^2 + k^2)$ if one also has $\omega_n \delta_n^2 = 1/2$ for all $n = 1, \ldots, d$.*

For the proofs of these results and some extensions, see [2] and [4].

3. Some numerical results

In this section, we illustrate the behavior of the previous SDI methods. We will consider a 3D problem of the kind (3) with $\Omega = (0, 1) \times (0, 1) \times (0, 1)$, $\alpha = 0$, $\beta = 1$ and $f(x) \equiv 4$. The exact solution is

$$u(x) = \sinh(\pi x_1)\sin(\pi x_2) + \sinh(\pi x_1)\sin(\pi x_3) + x_1(1 - x_1) + x_3(1 - x_3).$$

Of course, the imposed boundary conditions are those satisfied by u on $\partial\Omega$. The algorithm can be applied to arbitrary domains but, here, we have considered a very simple geometrical situation in order to analyze the speed-up and efficiency.

In Fig. 1, we represent at the logarithmic scale the ℓ_∞-norm of the exact error obtained at each iteration for $\tau = 0.1$, $h = 0.01$ and different values of ω. The use of the optimal ω (close to $2/3$) leads to an improvement of the convergence speed.

The algorithm has been implemented in a SGI Origin 2000 computer with 8 processors, using the parallel computing model of OpenMP. In order to measure its performance, we have introduced the *speed-up* S_P and the *efficiency* η_P, respectively defined by

$$S_P = \frac{\text{Resolution time with 1 processor}}{\text{Resolution time with } P \text{ processors}}, \qquad \eta_P = \frac{S_P}{P}.$$

We have obtained results for different meshes and $P = 1, 2, 3, 6$ and 8. In Figs. 2(a) and 2(b), the corresponding speed gains S_P and efficiencies η_P are shown.

We see that, for coarse grids, parallelization does not improve the gain in speed. On the contrary, when the number of nodes is high, the gain in speed and the efficiency increase, and we obtain a (reasonable) efficiency of

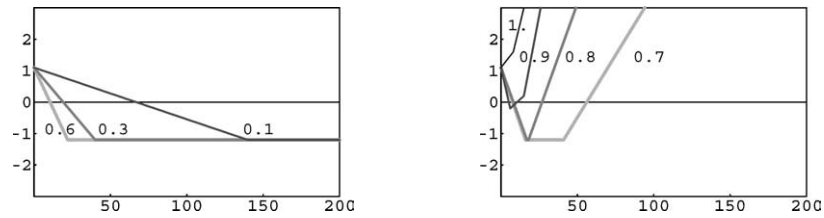
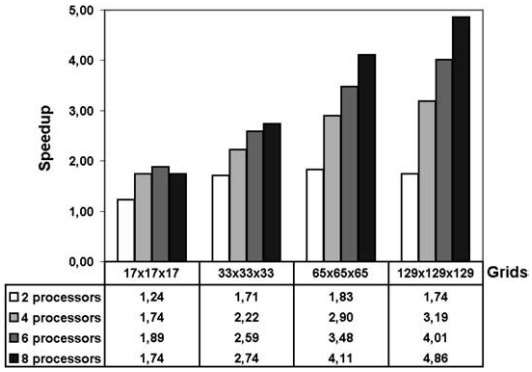
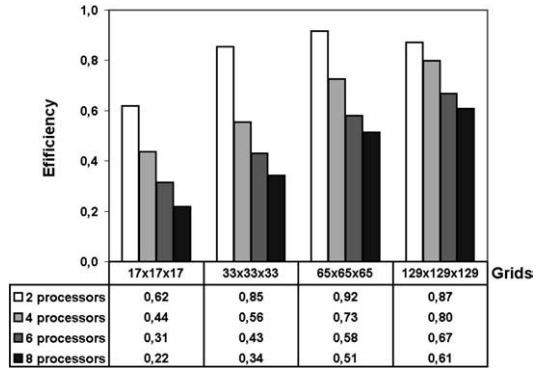


Fig. 1. The influence of ω for $\tau = 0.1$. We represent $\log|U^m - u|_\infty$ vs. the number of iterations.



(a)



(b)

Fig. 2. The speed-up and the efficiency. (a) The speed-up S_P; (b) The efficiency η_P.

0.6 for 8 processors. The results are similar for other tests in non-rectangular domains. For more details and other numerical results, see [2–4].

Acknowledgements

This work has been partially supported by D.G.E.S. (Spain), Grant BFM2003–06446.

References

[1] I. Albarreal, M.C. Calzada, J.L. Cruz, E. Fernández-Cara, J.R. Galo, M. Marín, Convergence analysis and error estimates for a parallel algorithm for solving the Navier–Stokes equations, Numer. Math. 93 (2002) 201–221.
[2] J.R. Galo, PhD Thesis, Univ. de Sevilla, 2002.
[3] J.R. Galo, I. Albarreal, M.C. Calzada, J.L. Cruz, E. Fernández-Cara, M. Marín, Convergence and optimization of the parallel method of simultaneous directions for the solution of elliptic problems, in press.
[4] J.R. Galo, I. Albarreal, M.C. Calzada, J.L. Cruz, E. Fernández-Cara, M. Marín, Stability and convergence of a parallel fractional step method for the solution of linear parabolic problems, in press.
[5] T. Lu, P. Neittaanmäki, X.C. Tai, A parallel splitting-up method for partial differential equations and its applications to Navier–Stokes equations, RAIRO–Math. Model. Num. 26 (6) (1992) 673–708.
[6] G.I. Marchuk, Splitting and alternating direction methods, in: Handbook of Numerical Analysis, I, North-Holland, 1990, pp. 197–462.