A 2/66

Appl. Math. Lett. Vol. 4, No. 2, pp. 25–29, 1991
Printed in Great Britain.

0893-9659/91 $3.00 + 0.00

A Parallel Splitting Up Method and Its Application To Navier-Stokes Equations

[1]T. Lu, [2]P. Neittaanmäki and [2,3]X-C. Tai

[1]Institute of Mathematical Science, Academia Sincica
[2]University of Jyväskylä, Department of Mathematics
[3]Institute of Systems Science, Academia Sinica

(Received June 1990)

Abstract. A parallel splitting-up method (or the so called alternating-direction method) is proposed in this paper. The method not only reduces the original linear and nonlinear problems into a series of one dimensional linear problems, but also enables us to compute all these one dimensional linear problems by parallel processors. Applications of the method to linear parabolic problem, steady state and nonsteady state Navier-Stokes problems are given.

1. INTRODUCTION

The classical splitting-up method [1] is efficient for solving physical problems. However it is not suitable for parallel computing, as the computing of the present fractional step (or so called intermediate step) always needs the value of the previous fractional step. Here we propose a splitting-up method, for which the computing of the fractional steps are independent of each other, and consequently they can be computed by parallel processors.

We also apply the proposed method to steady state and nonsteady state Navier-Stokes problems. By the method we can reduce the Navier-Stokes problem into a series of independent one dimensional linearized problems; moreover, we are able to solve the Navier-Stokes equation by using only the usual one dimensional finite element space (not the zero-divergent FE space as usual) and to use parallel processors in solving all these one dimensional linear problems.

2. PARALLEL SPLITTING UP METHOD FOR THE EVOLUTION EQUATIONS

In this section, we will consider the following linear evolution equation:

$$\begin{cases} \dfrac{\partial\phi}{\partial}t + A\phi = f & \text{in } Q_T = \Omega\times[0,T] \\ \phi(0) = \phi_0. \end{cases} \tag{2.1}$$

Here Ω is a bounded domain in $\mathbb{R}^m$, A can be both a linear operator or a matrix. We assume that A is time independent and that A and f can be split as:

$$\begin{aligned} A &= A_1 + A_2 + \cdots + A_m \\ f &= f_1 + f_2 + \cdots + f_m. \end{aligned} \tag{2.2}$$

For example, if $A\phi = \operatorname{div}(a\operatorname{grad}\phi)$, then we can split A as:

$$A = \sum_{i=1}^{m} A_i, \qquad \text{with } A_i\phi = D_i(aD_i\phi).$$

Supported by the Academy of Finland and TEKES, Finsoft III- project

In the following, we propose a splitting-up method, for which the computing of the fractional steps are independent of each other. Consequently they can be computed by parallel processors. Also, because of this, it may avoid the complicated boundary correction procedure [1,§3] in order to achieve the higher order convergence.

The classical splitting-up method can be regarded as a perturbation of the Crank-Nicolson scheme. But it seems the scheme proposed here cannot be regarded as a perturbation of Crank-Nicolson scheme.

ALGORITHM 2.1. *(Parallel splitting up method for evolution equations)*

Step 1. Choose step size $\tau > 0$. If ϕ^j is already computed, we parallelly compute $\phi^{j+\frac{i}{2m}}$ as

$$\frac{\phi^{j+\frac{i}{2m}} - \phi^j}{m\tau} + A_i\phi^{j+\frac{i}{2m}} = f_i^j. \tag{2.3}$$

Here $f_i^j = f_i((j+\frac{1}{2})\tau)$.

Step 2. Set $\phi^{j+1} = \frac{1}{m}\sum_{i=1}^{m}\phi^{j+\frac{i}{2m}}$.

Step 3. If $T = (j+1)\tau$ stop, otherwise go to Step 1.

If $A_i, i = 1, \dots, m$, are given as in the example, then as in [5], (2.3) can again be solved by parallel processors for a series of one dimensional problems.

THEOREM 2.1. *If A is the infinitesimal generator of a C_0-semigroup and $A_i \geq 0, i = 1, 2, \dots, m$ then for any $\tau > 0$ Algorithm 2.1 is stable and the error is:*

$$e^j = \phi(j\tau) - \phi^{j+1} = o(\tau). \tag{2.4}$$

We can achieve, as well, second order convergence:

ALGORITHM 2.2. *(Second order parallel splitting up method)*

Step 1. Assume ϕ^j is known. Compute $\phi^{j+\frac{i}{2m}}$ parallelly as:

$$(I + \frac{m}{2}\tau A_i)\phi^{j+\frac{i}{2m}} = \left(I - \sum_{k=1,k\neq i}^{m}\frac{m}{2}\tau A_k\right)\phi^j + \frac{m}{2}\tau f^j \quad i = 1, \dots, m. \tag{2.5}$$

Step 2. Set

$$\phi^{j+1} = \frac{2}{m^2}\left[\left(\frac{m^2}{2} - m\right)\phi^j + \sum_{i=1}^{m}\phi^{j+\frac{i}{2m}}\right]. \tag{2.6}$$

Similarly, if $A_i, i = 1, 2, \dots, m$, are given as in the example, (2.5) can be solved by parallel processors for a series of one dimensional problems as in [5].

THEOREM 2.2. *If $\tau > 0$ is small enough, then Algorithm 2.2 is stable and is globally second order convergent.*

REMARK 2.1. *We assumed that A is time independent which is not essential to the problem. When A is related to t, Algorthms 2.1-2.2 are still valid. Then we compute ϕ^{j+1} by taking $A((j+1/2)\tau)$ as A.*

3. PARALLEL SPLITTING UP METHOD FOR THE NAVIER-STOKES EQUATION

3.1. Steady state Navier-Stokes problems.

It is well known (see [7], for example) that the m-dimensional steady state incompressible fluid flow can be described by the following Navier-Stokes equation:

$$\begin{cases} -\nu\Delta u + \sum_{i=1}^{m} u_i D_i u + \operatorname{grad} p = f, & \text{in } \Omega \\ \operatorname{div} u = 0, & \text{in } \Omega \\ u = 0, & \text{on } \partial\Omega. \end{cases} \tag{3.1}$$

Here the domain $\Omega \subset \mathbb{R}^m$ (m = 2 or 3) is bounded and open, $u = (u_1, u_2, \dots, u_m)$ is the velocity of the fluid which is a m-dimensional vector function, p is the pressure which is a scalar function, $D_i = \frac{\partial}{\partial x_i}$. Let $\mathbf{L}^2(\Omega)$ be a vector Hilbert space with inner product and norm as: $(u, v) = \sum_{i=1}^m \int_\Omega u_i v_i dx, |u|^2 = (u, u)$. Let $\mathbf{H}_0^1(\Omega)$ be the closure of $\mathbf{C}_0^\infty(\Omega) = C_0^\infty(\Omega) \times \dots \times C_0^\infty(\Omega)$ under the inner product $((u, v)) = \sum_{i=1}^m (D_i u, D_i v)$, and we also denote the norm as $\|u\|^2 = ((u, u))$. Here and also later $|\cdot|, \|\cdot\|$ and $(\cdot,\cdot), ((\cdot,\cdot))$ will be used to denote the norms and inner products respectively as we defined here. Set $\tilde{\mathbf{V}} = \{u \in \mathbf{C}_0^\infty(\Omega) \,|\text{div } u = 0\}$. $\mathbf{H}$ is the closure of $\tilde{\mathbf{V}}$ in the sense of $\mathbf{L}^2(\Omega)$ norm. $\mathbf{V}$ is the closure of $\tilde{\mathbf{V}}$ in the sense of $\mathbf{H}_0^1(\Omega)$ norm ([2,7]).

The Navier-Stokes equation (3.1) has the following variational formulation: find $u \in \mathbf{V}$ such that

$$\nu((u, v)) + b(u, u, v) = (f, v) \quad \forall v \in \mathbf{V}, \tag{3.2}$$

where the trilinear form $b(\cdot,\cdot,\cdot)$ is defined as:

$$b(u, w, v) = \sum_{i=1}^m \int_\Omega u_i \frac{\partial w}{\partial x_i} v dx \quad \forall u, v, w \in \mathbf{H}_0^1(\Omega).$$

We can easily see that there exists a constant which is only related to m, such that

$$|b(u, v, w)| \le C(m)\|u\| \, \|v\| \, \|w\|.$$

Here we will use the splitting up method to solve (3.1) in such a way that when we are going to use the finite element method to solve (3.1), we do not need to construct a finite element space for $\mathbf{V}$. Moreover, in the proposed method the computing can also be done by parallel processors.

The application of the splitting-up method to Navier-Stokes problems was originally introduced by Teman [6]. In [7], more information about this application can be found. In [6,7], the splitting up method was applied to solve the nonstationary Navier-Stokes problem. For steady state problems, it seems that no results are available yet. In this section, we will propose the splitting up method for the steady state Navier- Stokes problem (3.1). The Algorithm 3.1 given below is a two-step iteration scheme: in the first step, it solves a nonlinear elliptic system, and in the second step, it solves a Poisson equation. In Algorithm 3.2, we split the problem into a series of independent 1-D problems.

ALGORITHM 3.1. *(Nonlinear splitting up method for Navier-Stokes equation)*

Step 1. Choose an initial function $u^{1/2} \in \mathbf{H}_0^1(\Omega)$, *parameter* $\tau > 0$ *and the error tolerance* $\epsilon_0 > 0$.

Step 2. If we have $u^{j-1/2}, j \ge 0$, *then we solve the following Poisson equation:*

$$\begin{cases} \Delta p^j = \text{div } u^{j-1/2}, & \text{in } \Omega \\ \frac{\partial p^j}{\partial n} = 0, & \text{on } \partial\Omega. \end{cases} \tag{3.3}$$

Here $\frac{\partial p^j}{\partial n}$ *denotes the outer normal derivative to* $\partial\Omega$.

Step 3. Let $u^j = u^{j-1/2} - \text{grad } p^j$.

If $|u^j - u^{j-1}| \le \epsilon_0$ *then stop, otherwise go to Step 4.*

Step 4. Solve the following nonlinear elliptic system: find $u^{j+1/2} \in \mathbf{H}_0^1(\Omega)$ *such that*

$$(u^{j+1/2} - u^j, v) + \tau\nu((u^{j+1/2}, v)) + \tau\hat{b}(u^{j+1/2}, u^{j+1/2}, v) = \tau(f, v) \quad \forall v \in \mathbf{H}_0^1(\Omega), \tag{3.4}$$

and go to Step 2. Here $\hat{b}(u, v, w) = \frac{1}{2}(b(u, v, w) - b(u, w, v)), \quad u, v, w \in \mathbf{H}_0^1(\Omega)$.

THEOREM 3.1. *If* $m \le 4$ *and* ν *is large enough or* f *"small" enough such that* $\nu^2 > C(m)\|f\|_{H^{-1}(\Omega)}$. *Then as* $\tau \to 0$ *and* $n\tau \to \infty$, $\|u^n - u\|_{\mathbf{L}^2(\Omega)} \to 0$.

ALGORITHM 3.2. *(A parallel linear splitting up method for Navier-Stokes equation)*
Step 1, Step 2 and Step 3 are the same as Algorithm 3.1.
Step 4. Solve parallelly $u^{j+i/q} \in \mathbf{H}_0^1(\Omega), q = 2(m+1)$ *such that*

$$(u^{j+i/q} - u^j, v) + \tau\nu((u^{j+i/q}, v))_i + \tau b_i(u^j, j^{j+i/q}, v) = \tau(f_i, v) \quad \forall v \in \mathbf{H}_0^1(\Omega), \tag{3.5}$$

$$i = 1, \dots, m, \text{ where } ((u,v))_i = (D_i u, D_i v) \text{ and}$$

$$b_i(u, w, v) = \frac{1}{2}\int_\Omega u_i \frac{\partial w}{\partial x_i} v dx - \frac{1}{2}\int_\Omega u_i w \frac{\partial v}{\partial x_i} dx.$$

Step 5. Set $u^{j+1/2} = \frac{1}{m}\sum_{i=1}^m u^{j+i/q}$ *and go to Step 2.*

For every i, (3.5) can be solved parallelly by the 1-D method as in [5], especially by a one dimensional finite element method. In order to get the convergence of Algorithm 3.2, we need the following assumption:

(A1) Suppose that $||u||_{0,\infty}^2 + ||u||_{1,\infty}^2 \le \nu^2\gamma_0^2$.

Here $\gamma_0 = \min_{1 \le i \le n}(1, \gamma_i)$ and γ_i is the Sobolev constant such that $|D_i v| \ge \gamma_i |v|^2 \quad \forall v \in \mathbf{H}_0^1(\Omega)$.

In correspondence with (**A1**), we also assume that u satisfies the needed regularity properties of (**A1**).

THEOREM 3.2. *Under condition* (**A1**) *Algorithm 3.2 is convergent in the same sense as in Theorem 3.1.*

REMARK 3.1. *In condition* (**A1**), *we need* $u \in \mathbf{W}_0^{1,\infty}(\Omega) \cap \mathbf{H}^2(\Omega) \cap \mathbf{H}_0^1(\Omega)$ *and* $||u||_{0,\infty}^2 + ||u||_{1,\infty} \le \nu^2\gamma_0^2$. *As u is also related to ν we should show that these conditions are not "empty conditions." It is easy to show that if Ω is smooth, f regular and small enough or ν big enough then condition* (**A1**) *can be satisfied (see [4,7]).*

3.2. The evolution Navier-Stokes equation.

In this section, we consider the evolution Navier-Stokes equation:

$$\begin{cases} \dfrac{\partial u}{\partial t} - v\Delta u + \displaystyle\sum_{i=1}^m u_i D_i u + \operatorname{grad} p = f & \text{in } Q_T = \Omega \times [0, T] \\ \operatorname{div} u = 0 & \text{in } Q_T \\ u = 0 & \text{on } \partial\Omega \times [0, T] \\ u(x, 0) = u_0(x) & \text{in } \Omega. \end{cases} \tag{3.6}$$

We will use the same notations as in Chapter 3.1. We will also restrict our attention to only two and three dimensional problems, i.e., $m = 2$ or 3.

The application of the splitting up method to evolution Navier-Stokes problems can be found in Temam [6,7]. In the following, we will use the parallel splitting-up method for the Navier-Stokes equation. It reduces the Navier-Stokes equation into a series of one dimensional linearized problems, and so enables us to solve the Navier-Stokes equation by only using the usual one dimensional finite element spaces. All the one dimensional problems are independent of each other and can be computed by parallel processors.

ALGORITHM 3.3. *(Linearized one dimensional splitting up method)*
Step 1. Choose $\tau > 0$. *Set* $u^0 = u_0$ *and split* $f = f_1 + \cdots + f_m$.
Step 2. If u^j *is known then find* $u^{j+i/q} \in \mathbf{H}_0^1(\Omega)(q = 2(m+1))$ *such that*

$$(u^{j+i/q} - u^j, v) + m\tau\nu((u^{j+i/q}, v))_i + m\tau b_i(u^j, u^{j+i/q}, v)$$

$$= m\tau(f_i((j+1)\tau), v) \quad \forall v \in \mathbf{H}_0^1(\Omega), \quad i = 1, 2, \dots, m. \tag{3.7}$$

Step 3. Set $u^{j+1/2} = \frac{1}{m}\sum_{i=1}^{m} u^{j+i/q}$.
Step 4. Solve p^j :

$$\begin{cases} \Delta p^j = \operatorname{div} u^{j-1/2}, & \text{in } \Omega \\ \frac{\partial p^j}{\partial n} = 0, & \text{on } \partial\Omega. \end{cases} \tag{3.8}$$

Step 5. Set $u^{j+1} = u^{j+1/2} - \operatorname{grad} p^{j+1}$. *If* $(j+1)\tau = T$ *stop, otherwise go to Step 2.*

Similar to Algorithm 3.2, (3.7) is a series of independent one dimensional problems. We can use a one dimensional finite element method as in [5] to solve all these one dimensional problems parallelly.

THEOREM 3.3. *If* $u(t)$ *is regular and* (**A**1) *is valid for every* $t \in [0, T]$, *then Algorithm 3.3 is convergent and*

$$\|u^n - u(n\tau)\|^2_{L^2(\Omega)} \le C\tau,$$

where the constant C *is independent of* τ.

For the proofs of above presented theorems we refer to [4].

REMARK 3.2. *In paper [3] by Lions and Temam, the same parallel splitting idea, as proposed in Algorithm 2.1, was discovered and used in connection with the optimization problems (with applications to variational inequalities).*

REFERENCES

1. A. R. Goulary, Splitting method for time dependent partial differential equations, In *The State of the Art in Numerical Analysis* (Proc. Conf. Univ. York, Heslington, 1976), pp. 757-796, Academic Press, London, (1977).
2. M. Křížek and P. Neittaanmäki, Finite element approximation to variational problems with application, *Pitman Monographs in Pure and Applied Mathematics* **50**, Longman (1990).
3. J.L. Lions and R. Temam, Eclaterment et decentralisation en calcul des variations, Springer Verlag, In *Proc. of Symposium on Optimization, Lecture Notes in Mathematics*, pp. 196-217, (1970).
4. T. Lu, P. Neitaanmäki and X-C. Tai, A parallel splitting up method for partial diferential equations and its applications to Navier-Stokes equation, Research Report 8 , University of Jyväskylä, Department of Mathematics, (1989).
5. X-C. Tai and P. Neittaanmäki, A parallel finite element splitting up method for parabolic problems, *Numerical methods for partial differential equations* (to appear 1990).
6. R. Temam, Sur lápproximation de la solution des équations de Navier-Stokes par la méthode de pas fractionnaires (I), *Arch. Rational Mech. Anal.* **32**, 135-153 (1969).
7. R. Temam, *Navier-Stokes Equations*, North-Holland, Amsterdam, (1977).

[1]Permanent address: Institute of Mathematical Science, Academia Sinica, Chengdu 610015, P.R. China
[2]University of Jyväskylä, Department of Mathematics, 40100 Jyväskylä, Finland
[3]Permanent address: Institute of Systems Science, Academia Sinica, Beijing 100080, P.R. China