A Parareal in Time Semi-implicit Approximation of the Navier-Stokes Equations

Paul F. Fischer,[1] Frédéric Hecht,[2] Yvon Maday[2]

[1] Argonne National Laboratory, Mathematics and Computer Science Division
[2] Université Pierre et Marie Curie, Laboratoire Jacques Louis Lions, Boite courrier 187 75252 Paris cedex 05, France.

Summary. The "parareal in time" algorithm introduced in Lions et al. [2001] enables parallel computation using a decomposition of the interval of time integration. In this paper, we adapt this algorithm to solve the challenging Navier-Stokes problem. The coarse solver, based on a larger timestep, may also involve a coarser discretization in space. This helps to preserve stability and provides for more significant savings.

1 Introduction

The "parareal in time" algorithm was introduced in Lions et al. [2001] to allow parallel computations based on a decomposition of the interval of time integration. This algorithm, which can be interpreted as a predictor-corrector scheme (see Bal and Maday [2002] and Baffico et al. [2002]), involves a prediction step based on a coarse approximation and propagation of the phenomenon and a correction step computed in parallel and based on a fine approximation. Significant speedups are observed (see, in particular, Bal [2003] on this aspect). A combination of the parareal in time algorithm with more conventional domain decomposition approaches was presented in Maday and Turinici [2003] and exploits both space and time concurrency.

Many applications of the method have already been performed, but this paper is the first that targets the challenging Navier-Stokes problem. The coarse solver is based on a large timestep but also on a coarse discretization in space, which further reduces serial overhead.

2 The Basic Algorithm on a Simple Equation

Consider the following time dependent problem:

$$\frac{\partial y}{\partial t} + \mathcal{A}y = 0, \quad y(T_0) = y_0,$$

where, for the sake of simplicity, $\mathcal{A}$ *does not depend on time.* We introduce the propagator $\mathcal{E}$ such that $y(\tau) = \mathcal{E}_\tau(y_0)$. Let $T_n = n\Delta T$, $n = 0, ..., N$ be instants at which we wish to consider snapshots of the solution. Then $y(T_n) = \mathcal{E}_{T_n}(y^0) = \mathcal{E}_{\Delta T}(y(T_{n-1}))$, from the semigroup property of $\mathcal{E}$.

In most cases $\mathcal{E}$ is not realizable and can only be approximated; for instance, we can introduce a fine and precise propagator $\mathcal{F}$ defined by an Euler scheme, either implicit or explicit. Similar to the continuous solution, we have the approximations $y(T_n) \simeq \lambda_n = \mathcal{F}_{T_n}(y_0) = \mathcal{F}_{\Delta T}(\lambda_{n-1})$. Clearly, the approximation process is sequential.

The parareal algorithm assumes we are given another propagator denoted as $\mathcal{G}$. It is cheaper (and consequently less accurate) than $\mathcal{F}$. One can think of $\mathcal{F}$ as based on an Euler scheme with a very small timestep δt and $\mathcal{G}$ as based on an Euler scheme with the larger timestep ΔT. We present and implement here another possibility, as proposed in Lions et al. [2001], in which $\mathcal{G}$ is based on a coarse approximation in space as well.

The iterative process $\lambda_n^{k+1} = \mathcal{G}_{\Delta T}(\lambda_{n-1}^{k+1}) + \mathcal{F}_{\Delta T}(\lambda_{n-1}^k) - \mathcal{G}_{\Delta T}(\lambda_{n-1}^k)$ provides a converging sequence toward λ_n. Our interest in this predictor-corrector scheme lies in the fact that after iteration k and before iteration $k+1$ starts, we can compute in parallel the corrections $\mathcal{F}_{\Delta T}(\lambda_{n-1}^k) - \mathcal{G}_{\Delta T}(\lambda_{n-1}^k)$ for all n; thus the only sequential part of the algorithm is the evaluation of the coarse operator.

3 The Parareal in Time Algorithm for Navier-Stokes

We apply the parareal scheme to the incompressible Navier-Stokes equations,

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \frac{1}{Re}\nabla^2 \mathbf{u} \text{ in } \Omega, \quad \nabla \cdot \mathbf{u} = 0 \text{ in } \Omega, \tag{1}$$

with prescribed boundary and initial conditions for the velocity, $\mathbf{u}$. Here, p is the pressure, Re the Reynolds number and Ω is a regular domain of $\mathbb{R}^d$.

The temporal discretization is based on the high-order operator-splitting methods developed in Maday et al. [1990] that generalize the characteristics method of Pironneau [1982]. The left-hand side of (1) is recast as a material derivative, which is discretized by using a stable rth-order backward difference formula (BDFr):

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = \frac{D\mathbf{u}}{Dt} \approx \frac{1}{\Delta t}\left(\beta_0 \mathbf{u}_{n+1} - \sum_{i=1}^{r} \beta_i \overline{\mathbf{u}}_{n+1-i}\right). \tag{2}$$

For BDF1, $(\beta_0, \beta_1) = (1, 1)$, and for BDF2, $(\beta_0, \beta_1, \beta_2) = (\frac{3}{2}, \frac{4}{2}, \frac{-1}{2})$. The values $\overline{\mathbf{u}}_{n+1-i}$ represent the values of $\mathbf{u}$ at the foot of the characteristic associated with each gridpoint and are computed by solving the linear convective subproblem, $(\overline{\mathbf{u}}_j)_t + \mathbf{u} \cdot \nabla \overline{\mathbf{u}}_j = 0$, $t \in (t_{n+1-i}, t_{n+1}]$, with initial condition $\overline{\mathbf{u}}_j(t_{n+1-i}) = \mathbf{u}_{n+1-i}$ for $j = n+1-i$, $i = 1, \ldots, r$. This leads to a linear

symmetric Stokes problem to be solved implicitly at each step and permits timestep sizes, Δt, corresponding to convective CFL numbers greater than unity, thus significantly reducing the number of Stokes solves.

3.1 Finite Element Approximation in Space

The finite element method is based on an compatible choice of spaces for the velocity and the pressure: the Taylor-Hood method. The time discretization is a characteristics method of order 1 for the convection and implicit for the Stokes operator. The algebraic system resulting from the discretization is solved through a Cahouet-Chabart algorithm. The problem solved corresponds to a 2-D flow past a cylinder of diameter 2 centered at the origin. The upstream boundary is located at a distance 5 from the center of the cylinder and the downstream boundary at a distance 10.

A special consideration when using the parareal scheme in conjunction with a *semi-implicit* timestepping scheme is that the step size be small enough to ensure stability. This is a particular concern for the coarse solver, $\mathcal{G}$, where one wants to choose a large timestep in order to minimize the serial overhead. Fortunately, the characteristics scheme allows this, provided that the subintegration timestep satisfies the governing stability criterion.

3.2 Spectral Approximation in Space

The spectral element method (SEM) for the Stokes problem is also based on a compatible choice of spaces: it is the $\mathbb{P}_M \times \mathbb{P}_{M-2}$ discretization introduced in Maday and Patera [1989]. The discretization spaces are $X^M := \left\{ \mathbf{v} \in H_0^1(\Omega)^d, \mathbf{v}_{|\Omega^e} \in \mathbb{P}_M(\Omega^e)^d, e = 1, ..., E \right\}$ for the velocity and $Y^M := \{q \in L^2(\Omega), q_{|\Omega^e} \in \mathbb{P}_{M-2}(\Omega^e), e = 1, ..., E\}$, for the pressure. Here, $\mathbb{P}_M(\Omega^e), e = 1, ..., E$ is the space of polynomials of partial degree $\leq M$ on each of the E nonoverlapping elements, Ω^e, whose union composes Ω. At present, we restrict our attention to cases where Ω is a rectangular domain comprising a tensor-product array of $E = E_x \times E_y$ elements allowing the use of the fast diagonalization method. Details of the SEM formulation and implementation can be found in Fischer [1997].

To implement the parareal scheme, we require a solver of the form $\underline{\mathbf{u}}_{n+1} = \mathcal{F}_{\Delta T}(\underline{\mathbf{u}}_n)$. That is, given an initial value $\underline{\mathbf{u}}_n$, the solver propagates the solution over a time interval $(T_n, T_{n+1}]$ to produce a result $\underline{\mathbf{u}}_{n+1}$. We thus need to "restart" the computation for each application of $\mathcal{F}$ and $\mathcal{G}$: we use Richardson extrapolation and combine two steps of size $\Delta t/2$ with one of size Δt to yield an $O(\Delta t^3)$ local error at the start of each $\mathcal{F}$ ($\Delta t = \delta t$) or $\mathcal{G}$ ($\Delta t = \Delta T$) substep.

4 Further Reduction of the Cost of the Coarse Operator

In nondimensional time units, the parareal single- and P-processor solution times and parallel speedup are, respectively,

$$\tau_1 = \frac{T}{\delta t}, \;\; \tau_P = K\left(\frac{1}{P}\frac{T}{\delta t} + \alpha\frac{T}{\Delta T}\right), \;\; S_P := \frac{\tau_1}{P\tau_P} = \frac{P}{K\left(1 + \alpha P\frac{\delta t}{\Delta T}\right)}, \;\; (3)$$

The P-processor estimates in (3) neglect communication overhead and simply reflect the extra work resulting from additional iterations (K) and the serial coarse propagation ($\alpha T/\Delta T$). The factor α reflects the relative per-step cost of $\mathcal{F}_{\Delta T}$ and $\mathcal{G}_{\Delta T}$. To achieve reasonable scalability we need K and $\alpha P\frac{\delta t}{\Delta T}$ to be order unity.

Here, we propose to use propagators based not only on different timesteps, but also on different spectral degrees.

4.1 The Finite Element Context

The reduction is obtained by using a fine grid defined by dividing each coarse triangle into four triangles. The resulting number of vertices is equal to 1021 in the coarse mesh (H) and 3994 in the fine mesh (h). The coarse operator $\mathcal{G}_{\Delta T}$ is based on ΔT and the coarse grid H. The proposed parareal in time scheme is then

$$\mathbf{U}_{n+1}^{k+1} = \Pi_H^h \mathcal{G}_{\Delta T}(\Pi_h^H \mathbf{U}_n^{k+1}) + \mathcal{F}_{\Delta T}(\mathbf{U}_n^k) - \Pi_H^h \mathcal{G}_{\Delta T}(\Pi_h^H \mathbf{U}_n^k), \qquad (4)$$

The operator that allows one to go from the coarse mesh to the fine one (denoted as Π_H^h) and reciprocally (i.e., Π_h^H) can be either the interpolation operator I_h (resp. I_H) or the L^2 projection on discretely divergence free functions Π_h (resp. Π_H).

4.2 The Spectral Context

We assume that we have a coarse operator $\mathcal{G}_{\Delta T}$ based on ΔT and a spectral degree $\tilde{M}$ together with a fine solver $\mathcal{F}_{\Delta T}$ based on δt and a spectral degree $M > \tilde{M}$. The proposed parareal in time scheme is then

$$\mathbf{U}_{n+1}^{k+1} = \Pi_{\tilde{M}}^M \mathcal{G}_{\Delta T}(\Pi_M^{\tilde{M}} \mathbf{U}_n^{k+1}) + \mathcal{F}_{\Delta T}(\mathbf{U}_n^k) - \Pi_{\tilde{M}}^M \mathcal{G}_{\Delta T}(\Pi_M^{\tilde{M}} \mathbf{U}_n^k), \qquad (5)$$

where $\Pi_{\tilde{M}}^M$ is the L^2 prolongation operator from $X^{\tilde{M}}$ onto X^M and $\Pi_M^{\tilde{M}}$ is the L^2 projection operator from X^M onto $X^{\tilde{M}}$.
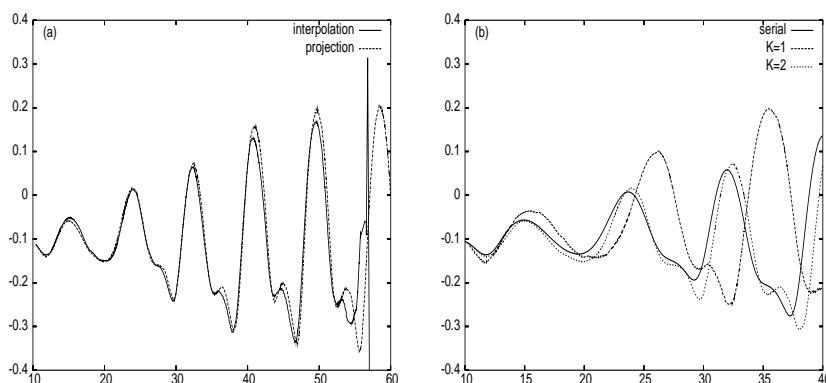
Fig. 1. Time history of the vertical component of the velocity at point $(1.5, 0)$ for the parareal algorithm with coarse operator based on coarse spatial and temporal discretization. (a) Comparison of interpolation and L^2 projection operators. (b) Comparison $K = 1$, $K = 2$, and standard serial algorithms.

5 Results

5.1 The Finite Element Context

We have run the cylinder simulation at $Re = 200$, starting with an initial condition made of a flow computed by running the coarse simulation over a time equal to 10. We have used a fine timestep equal to 0.02 and a coarse timestep equal to 0.2. The simulation has been run over successive time intervals of size 2 corresponding to $P = 10$. Over these intervals, we have run the parareal scheme with small numbers of iterations because of the small size of the intervals.

We have first compared the two types of operators to go from one mesh onto the other one. The simple interpolation operator appears to be unstable: the simulation blows up, for example, with the use of the strategy three coarse sweeps alternated with two fine sweeps, corresponding to K=2. On the contrary, the use of an L^2-type projection operator on the discrete divergence-free functions is stable throughout these simulations. These results are illustrated in Fig. 1a, where we plot the time history of the vertical component of the velocity at a point situated on the axis of the flow at a distance 1.5 downstream of the center of the cylinder. To check the accuracy of the method, we have computed the solution corresponding to the fine timestep as a reference and compared the parareal scheme with 2 coarse + 1 fine (K=1) and 3 coarse + 2 fine (K=2) to the reference solution. The results are plotted in Fig. 1b, and the solution for K=2 is quite good. Note that the plot representing the history at this point is much discriminating because of the complexity of the
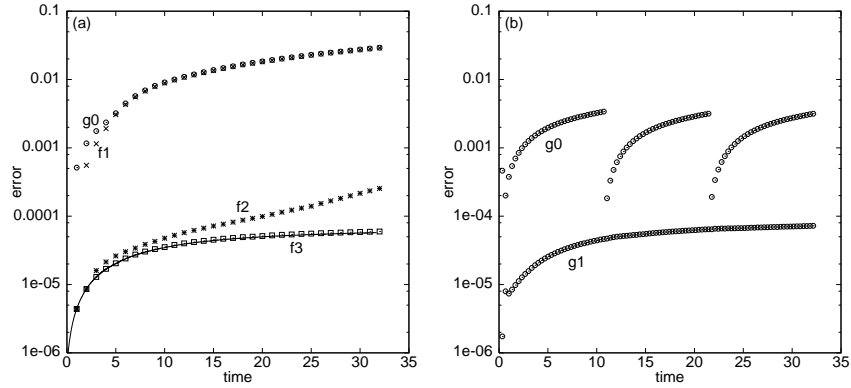
Fig. 2. Relative error in the v-component of the solution, versus time, for the SEM solution of the Orr-Sommerfeld test problem with $E = 15$, $M = 15$, and $\delta t = .005$. (a) single partition with $K = 3$, (b) tripartition with $K = 1$.

flow. For longer times, we have checked that the period of the flow and the time to establish the periodic flow are very well captured by the parareal scheme for this situation.

These are only preliminary runs, from which we conclude that the projection operator performs better than the interpolation one. We will seek to optimize the timestep choice in order to be able to get an accurate solution with a lower number of iterations. Let us note that these simulations have been done *effectively* in parallel by using the code Freefem.

5.2 The Spectral Context

We have applied the parareal/SEM algorithm to the Orr-Sommerfeld problem studied in Fischer [1997]. The computational domain is $\Omega = [0, 2\pi] \times [-1, 1]$, with periodic boundary conditions in x and homogeneous Dirichlet conditions on $y = \pm 1$. The growth of a small-amplitude (10^{-5}) Tollmien-Schlichting wave, superimposed on plane Poiseuille channel flow at $Re = 7500$, is monitored and compared with linear theory over the interval $t \in [0, 32]$, which corresponds to ≈ 1.25 periods of the traveling wave solution.

Figure 2a shows the relative L^∞ (maximum pointwise) error in the y-component of velocity versus time for an $E=15$ element discretization using BDF2 with $M=15$, $\Delta t_{\mathcal{F}} = \delta t = .005$ for $\mathcal{F}_{\Delta T}$, and $\tilde{M} = 15$, $\Delta t_{\mathcal{G}} = .333 = \Delta T/3$ for $\mathcal{G}_{\Delta T}$. The solid line shows the discretization error for the standard serial algorithm (BDF2,$\Delta t = .005$). The point plots show the error in the solution for the first coarse sweep (g0) and for the first three fine sweeps (f1, f2, and f3). The number of coarse and fine substeps per iteration is 32, corresponding to a $P = 32$ processor simulation. For this problem, the scheme
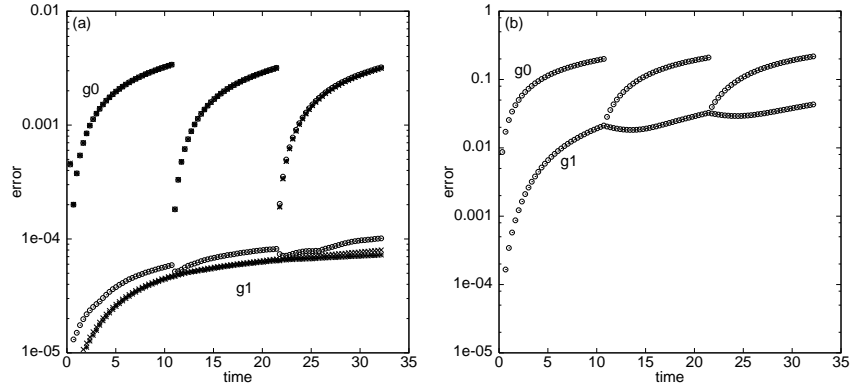
Fig. 3. Error histories for configuration as in Fig. 2b: (a) using varying fine/coarse approximation orders $(M,\tilde{M})$; $\circ = (13,13)$, $\times = (15,13)$, $* = (15,15)$; and (b) $(M,\tilde{M})$ $= (15,15)$ without Richardson extrapolation.

converges in $K = 3$ iterations. Each coarse-grid step, $\mathcal{G}_{\Delta T}(\underline{\mathbf{u}}^n)$, is computed by using three steps of size $\Delta t_{\mathcal{G}} = 1/3$, plus two additional steps for the Richardson extrapolation, corresponding to $\alpha = 5$. Based on (3), the estimated speedup is $S_{32} \approx 6$.

To reduce ΔT and, hence, K, we also consider applying the parareal scheme to *subintervals* of $[0, T]$, where the initial condition on each interval is taken to be the $K = 1$ solution from the preceding interval. The scheme requires an initial $\mathcal{G}_{\Delta T}$ sweep ($k = 0$), followed by a single $\mathcal{F}_{\Delta T}$ and $\mathcal{G}_{\Delta T}$ correction ($k = 1 =: K$). The errors after the $\mathcal{G}_{\Delta T}$ sweeps are shown in Fig. 2b. Here, three subintervals are used, with $\delta t = .005$, and $\Delta T = \Delta t_G = 67\delta t$. Given that there are two coarse sweeps per interval, each with a cost of three Navier-Stokes solves (because of the Richardson extrapolation), we have $\alpha = 6$, corresponding to a speedup of $S_{32} = 8.3$.

We next consider the coarse approximation in space for $\mathcal{G}_{\Delta T}$, given by (5). Here, one must be careful that the temporal errors do not dominate the spatial errors. Otherwise, perceived benefits from reducing $\tilde{M}$ could equally well be gained through reductions in both M and $\tilde{M}$. We verify that this is not the case by plotting in Fig. 3a the errors for the tripartition algorithm of Fig. 2b using discretization pairs $(M, \tilde{M})$=(15,15), (15,13), and (13,13). The error for the $(15, 13)$ pairing is almost the same as for the $(15, 15)$ case. The coarse-grid solve cost, however, is significantly reduced. For the SEM in two dimensions, this cost scales as $\tilde{M}^3$, so we may expect $\alpha \approx (13/15)^3 6$, which implies $S_{32} = 11.2$ for the three-step Richardson scheme.

The Richardson iteration was chosen for programming convenience. Other approaches with lower cost that also have an $O(\Delta t^3)$ local truncation error could be used for the initial coarse step. For example, one could employ

a semi-implicit scheme combining Crank-Nicolson and second-order Runge-Kutta that would require only a single set of system solves, thus effectively reducing α threefold. The corresponding speedup for $\alpha \approx 2(13/15)^3$ would be $S_{32} = 19.7$. Note that simply dropping the Richardson extrapolation in favor of BDF1 has disastrous consequences, as illustrated by the error behavior in Fig. 3b.

References

L. Baffico, S. Bernard, Y. Maday, G. Turinici, and G. Zérah. Parallel in time molecular dynamics simulations. *Phys. Rev. E.*, 66, 2002.

G. Bal. Parallelization in time of (stochastic) ordinary differential equations. Math. Meth. Anal. Num. (submitted), 2003.

G. Bal and Y. Maday. A "parareal" time discretization for non-linear PDE's with application to the pricing of an American put. In *Recent developments in domain decomposition methods (Zürich, 2001)*, volume 23 of *Lect. Notes Comput. Sci. Eng.*, pages 189–202. Springer, Berlin, 2002.

P. Fischer. An overlapping schwarz method for spectral element solution of the incompressible navier-stokes equations. *J. Comp. Phys.*, 133:84–101, 1997.

J.-L. Lions, Y. Maday, and G. Turinici. A parareal in time discretization of pde's. *C.R. Acad. Sci. Paris, Serie I*, 332:661–668, 2001. URL `http://www.elsevier.nl/gej-ng/10/37/18/47/27/35/article.pdf`.

Y. Maday and A. T. Patera. Spectral element methods for the Navier-Stokes equations. In A. Noor and J. Oden, editors, *State of the Art Surveys in Computational Mechanics*, New York, 1989. ASME.

Y. Maday, A. T. Patera, and E. M. Rønquist. An operator-integration-factor splitting method for time-dependent problems: application to incompressible fluid flow. *J. Sci. Comput.*, 5(4):263–292, 1990.

Y. Maday and G. Turinici. The parareal in time iterative solver : a further direction to parallel implementation. In *Fifteenth International Conference on Domain Decomposition Methods*, Berlin, 2003. Springer, Lecture Notes in Computational Science and Engineering.

O. Pironneau. On the transport-diffusion algorithm and its applications to the Navier-Stokes equations. *J. Num. Math.*, 38(3):309–332, Mar. 1982.