Parallel Spectral Element Solutions of Eddy-Promoter Channel Flow

Paul F. Fischer
Anthony T. Patera

CRPC-TR90053
March, 1990

Center for Research on Parallel Computation
Rice University
P.O. Box 1892
Houston, TX 77251-1892

Parallel Spectral Element Solutions of Eddy-Promoter Channel Flow

Paul F. Fischer
Center for Research on Parallel Computation
California Institute of Technology
Pasadena, CA 91125 USA

Anthony T. Patera
Department of Mechanical Engineering
Massachussetts Institute of Technology
Cambridge, MA 02139 USA

1 Introduction

We consider solution of the two-dimensional Navier-Stokes equations in the periodic domain of Fig. 1:

$$\begin{aligned} \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u} &= -\nabla p + \tfrac{1}{Re}\nabla^2\mathbf{u} \; + \; f(t)\hat{\mathbf{e}}_1 \qquad && \mathit{in}\;\Omega \\ \nabla.\mathbf{u} &= 0 && \mathit{in}\;\Omega, \end{aligned} \tag{1}$$

subject to,

$$\begin{aligned} \mathbf{u} &= 0 \qquad \mathit{on}\;\partial\Omega_1,\;\partial\Omega_2,\;\mathit{and}\;\partial\Omega_3 \quad ; \\ \mathbf{u}(0,y,t) &= \mathbf{u}(L,y,t) \quad ; \\ p(0,y,t) &= p(L,y,t) \quad . \end{aligned} \tag{2}$$

Equation (1) is nondimensionalized with respect to channel half-height, h, and the centerline velocity which would arise in laminar plane Poiseuille flow in the absence of the cylinder for the same given flow rate, resulting in a Reynolds number defined as:

$$Re \equiv \frac{\frac{3}{2}\bar{V}h}{\nu} \quad , \tag{3}$$

where

$$\bar{V} \equiv \frac{1}{2h}\int_0^{2h} u_1(x,y,t)\,dy \quad . \tag{4}$$

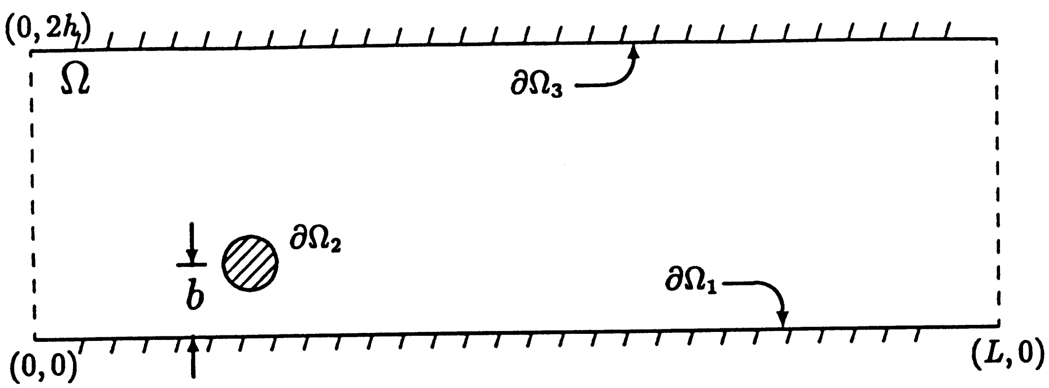


Figure 1: Eddy-promoter flow geometry: $L = 6.666h$, $b = 0.5h$, cyl. dia. $= 0.4h$.

The forcing term, $f(t)\hat{\mathbf{e}}_1$, corresponds to a mean pressure gradient driving the flow in the $\hat{\mathbf{e}}_1$ $(+x)$ direction. The functional form of $f(t)$ is chosen to enforce $\bar{V}$ to be constant with respect to time.

2 Numerical Formulation

Temporal discretization of (1) follows the fractional step method of [1,2], which leads to a series of elliptic problems to be solved at each time step by conjugate-gradient iteration. The method has been developed in such a way that it is readily implemented on distributed- or shared-memory parallel computers; we discuss these considerations in Section 4.

The fractional-step formulation is comprised of three computational steps. Beginning with explicit treatment of the nonlinear terms, compute $\hat{\mathbf{u}}$:

$$\frac{\hat{\mathbf{u}} - \mathbf{u}^n}{\Delta t} = \mathbf{C}^n + f^{n+1}\hat{\mathbf{e}}_1 \quad . \tag{5}$$

Here, $\mathbf{C}^n$ is taken to be an explicit representation of the nonlinear convective terms, in this case given by a 3rd order Adams-Bashforth discretization:

$$\mathbf{C}^n = \frac{23}{12}\mathbf{u}^n \cdot \nabla \mathbf{u}^n - \frac{16}{12}\mathbf{u}^{n-1} \cdot \nabla \mathbf{u}^{n-1} + \frac{5}{12}\mathbf{u}^{n-2} \cdot \nabla \mathbf{u}^{n-2} \quad . \tag{6}$$

Next, compute the pressure correction:

$$\begin{aligned} \frac{\hat{\hat{\mathbf{u}}} - \hat{\mathbf{u}}}{\Delta t} &= -\nabla p && \mathit{in}\ \Omega \\ \nabla^2 p &= \tfrac{1}{\Delta t} \nabla \cdot \hat{\mathbf{u}} && \mathit{in}\ \Omega \\ \nabla p \cdot \hat{\mathbf{n}} &= \tfrac{1}{\Delta t} \hat{\mathbf{u}} \cdot \hat{\mathbf{n}} && \mathit{on}\ \partial\Omega. \end{aligned} \tag{7}$$

Finally, compute the viscous correction:

$$\frac{\mathbf{u}^{n+1} - \hat{\mathbf{u}}}{\Delta t} = \frac{1}{Re}\nabla^2 \mathbf{u}^{n+1} \quad . \tag{8}$$

The method is first order accurate in time [3]. The most numerically intensive steps are the (iterative) elliptic solves, with the Neumann problem for pressure (7) having significantly slower convergence than the Helmholtz equations (8) for the viscous terms.

To ensure that the constant mass flux condition is satisfied at t^{n+1}, we recognize that, as formulated in (5-8), $\mathbf{u}^{n+1}$ is linear in f^{n+1}. Therefore, we can decompose (5-8) into two sub-problems, one for which $f^{n+1} = 0$, and one for which $\mathbf{u}^n, \mathbf{C}^n = 0$. Thus, we first solve (5-8) for:

$$\mathbf{u}^* = \mathbf{u}^*(\mathbf{u}^n = 0, \mathbf{C}^n = 0, f^{n+1} = 1) \quad . \tag{9}$$

Second, we compute:

$$\tilde{\mathbf{u}}^{n+1} = \tilde{\mathbf{u}}^{n+1}(\mathbf{u}^n, \mathbf{C}^n, f^{n+1} = 0) \quad . \tag{10}$$

Finally, we update $\mathbf{u}^{n+1}$ as:

$$\mathbf{u}^{n+1} = \tilde{\mathbf{u}}^{n+1} + \alpha \mathbf{u}^* \quad , \tag{11}$$

where α is a scalar such that the desired mean flow, $\bar{V}$, is obtained at t^{n+1}. Note that the advantage of the splitting (9-11) is that (9) is independent of time (though not independent of Δt), and needs only be solved for $\mathbf{u}^*$ once in a preprocessing step, which can then be stored.

In all cases, we take as our initial condition:

$$\begin{aligned} u_1(x,y,t=0) &= y(2-y) \\ u_2(x,y,t=0) &= 0 \quad . \end{aligned} \tag{12}$$

This initial condition is not divergence free due to the presence of the cylinder, but it offers the advantage of avoiding thin shear layers near the walls during startup which would necessitate additional spatial resolution.

Spatial discretization is based upon the spectral element method [4,5] which is a high-order weighted residual technique in which the domain is broken up into relatively few, macro- (spectral) elements, and the geometry, data, and solution within each element are approximated by tensor-product polynomial basis functions. A typical decomposition of the eddy-promoter geometry is shown in Fig. 2. C^0 continuity

is imposed across element interfaces and convergence is achieved through increased (polynomial) order of approximation within each element.

The above temporal discretization (5-8) leads to a series of elliptic sub-problems to be solved at each time step which are cast in the weak form:

Find $u \in \mathcal{H}_0^1(\Omega)$ *such that*

$$\int_\Omega \nabla\phi\nabla u + \lambda^2\phi u \, d\Omega \;=\; \int_\Omega \phi f \, d\Omega \quad \forall\phi \in \mathcal{H}_0^1(\Omega), \tag{13}$$

where the space $\mathcal{H}_0^1$ is the space of all functions which have a square integrable first derivative and are zero on the boundary.

The spectral element method proceeds by subdividing the domain, Ω, into K elements, Ω^k, which are mapped to the square $(x,y)\big|_{\Omega^k} \rightarrow (r,s) \in [-1,1]^2$. The solution, data, and test functions are expressed as tensor-product polynomials in (r,s) of degree $\leq N$ with respect to each variable. Thus, the discrete represenation of u takes the form:

$$u(x,y)\Big|_{\Omega^k} = \sum_{p=0}^{N}\sum_{q=0}^{N} u_{pq}^k h_p(r) h_q(s) \quad , \tag{14}$$

The Legendre spectral element method in $\mathbf{R}^2$ employs Lagrangian interpolant bases $h_i(\xi)$ satisfying $h_i(\xi_j) = \delta_{ij}$, where the grid points ξ_j are the Gauss-Lobatto-Legendre points. The polynomial coefficients $u_{pq}^k = u^k(r_p, s_q)$ are therefore the grid values of u in element k. Gauss-Lobatto quadrature assures accurate approximation to the integrals in (13). Further details of the spatial discretization and formulation of the elliptic problems can be found in [5].

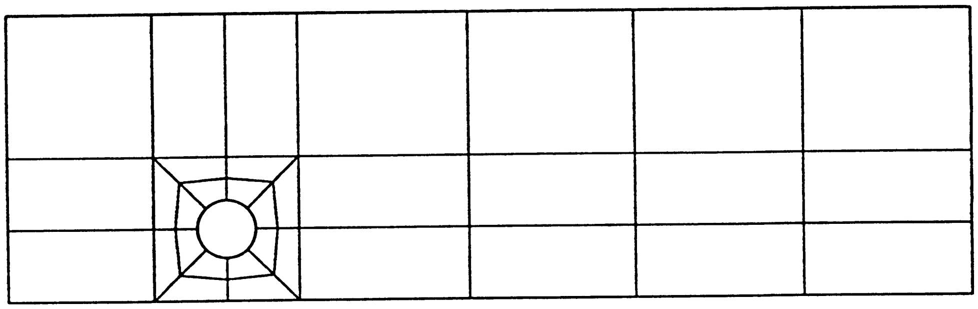


Figure 2: Spectral element mesh ($K = 33$) for the eddy-promoter geometry of Fig. 1

3 Iterative Solvers

The above discretization leads to a linear system of equations of the form:

$$\mathbf{A}\mathbf{u} = \mathbf{B}\mathbf{f} \quad , \tag{15}$$

where A is taken to be the discrete Laplacian (for the case $\lambda = 0$), **B** the diagonal mass matrix, **u** and **f** the (global) discrete representation of the solution and data respectively. Due to memory, operation count, and parallelization considerations, iterative methods are used to solve the system (15). Such methods are dependent upon repeated evaluation of matrix-vector products of the form $\mathbf{r} = \mathbf{A}\mathbf{u}$, where **u** and **r** are intermediate vectors associated with successive iterations.

For spectral element problems in higher space dimensions $\mathbf{R}^d$ the linear operators have large bandwidth and, if formed explicitly, are non-sparse with $O(KN^{2d})$ entries. The subsequent operation count and memory requirements can be significantly reduced if the matrix-vector product **Au** is evaluated element by element, using a factored form in which the discrete derivatives associated with the gradient operators in (13) are applied in a sequential fashion. A typical term in the elemental matrix-vector product $A^k u^k$ for the case of $d = 2$ is:

$$\sum_{p=0}^{N} \rho_{pj} D_{pi} (\sum_{q=0}^{N} D_{pq} u^k_{qj}) \qquad \forall i,j \in \{0,...,N\}^2 \tag{16}$$

where ρ_{ij} is the quadrature weight associated with the point (r_i, s_j), and D_{ij} is the derivative operator,

$$D_{ij} \equiv \left. \frac{dh_j(r)}{dr} \right|_{r=r_i} \quad . \tag{17}$$

The residual evaluation is completed via direct stiffness summation wherein intermediate residual values at nodes shared by multiple elements are summed and redistributed to the elemental data structures. The factored evaluation of **Au** requires only $O(KN^d)$ storage and $O(KN^{d+1})$ operations for general isoparametric discretizations.

Our current implementation employs standard Jacobi- (diagonal) preconditioned conjugate gradient iteration [6]. The preconditioned **A** system has condition $O(K_1^2 N^2)$ implying an iteration count of $O(K_1 N)$, where K_1 is the number of elements in a single spatial direction. The majority of the computational effort is associated with evaluation of **Au**, as all other terms have an operation count of $O(KN^d)$ or less. In addition, two operations require communication of information between elements, namely, direct stiffness summation and inner-product evaluations of the form $\mathbf{r}^t\mathbf{r}$. While these steps require only $O(KN^{d-1})$ and $O(KN^d)$ operations, respectively, they

represent the leading order *communication* terms in the parallel implementation discussed below.

4 Parallel Implementation

The spectral element discretizations, bases, and iterative solvers of the previous sections are constructed so as to admit a native, geometry-based parallelism, in which each spectral element (or group of spectral elements) is mapped to a separate processor/memory, with the individual processor/memory units being linked by a relatively sparse communications network. This conceptual architecture is naturally suited to the spectral element discretization in that it provides for tight, structured coupling within the dense elemental constructs, while simultaneously maintaining generality and concurrency at the level of the unstructured macro-element skeleton. The locally-structured/globally-unstructured spectral element parallel paradigm is closely related to the concept of domain-decomposition by substructured finite elements.

Our methods are implemented in an essentially machine-independent fashion. First, we construct a spectral element code in a standard high-level language in which each spectral element is treated as a "virtual parallel processor". In particular, each spectral element is treated as a separate entity, and all data structures and operations are defined and evaluated at the elemental level. The data and code are descended to M processors, each operating asynchronously. The only procedures which require communication are, by construction, the direct stiffness summation associated with residual evaluation, and vector reduction, which are relegated to special subroutines to effect data transfer. Processor synchronization is imposed at each iteration by the communication steps.

The residual calculation $\mathbf{Au}$ is the most complex operation in our parallel Navier-Stokes algorithm. Because of the variational formulation, the required action to update the residual along element interfaces is to first compute an intermediate residual vector, $A^k u^k$, within each element, and then sum correspondant edge values between elements. The parallelism in this procedure is quite evident; computation of local residuals requires $O(KN^{d+1})$ operations, while the communication between elements (processors) requires less than $O(KN^{d-1})$ words to be transmitted, resulting in a favorable computation to communication ratio of $O(N^2)$.

The conjugate-gradient iteration requires evaluation of two inner products which also require communication. These can be evaluated at a communication cost proportional to $\log_2 M$ on most topologies and any wormhole-connected communication system. The impact of this term is dependent on the work local to the processors, the ratio of computation to communication speed of the particular hardware, and

the number of processors. Typically it is not significant, but it can be a source of performance degradation for small problems when M is large, and ultimately limits the speedup possible for a particular problem [7].

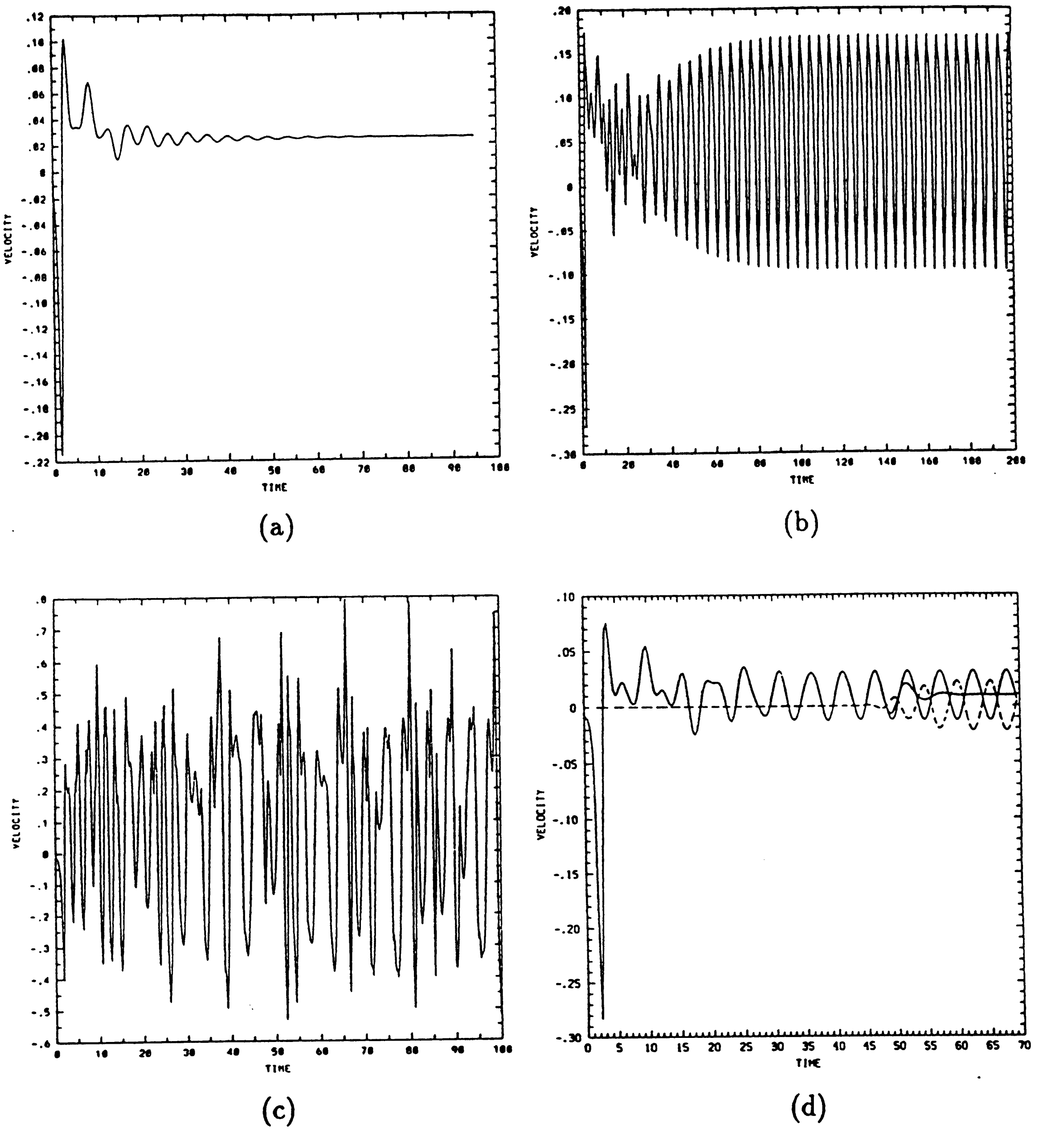
We have implemented our algorithms on the Intel iPSC/2-VX hypercube which is typical of the class of architectures for which the parallel spectral element method is well suited. The iPSC is a distributed memory, message passing, parallel processor consisting of $M = 2^D$ independent processor/memories, or nodes, arranged on a D-dimensional hypercube communication network. The iPSC/2-VX is an upgraded version of Intel's original iPSC hypercube which incorporates improved "worm-hole" message routing allowing data to be transferred between non-nearest-neighbor processors with minimal degradation in data transfer rate, as well as a twenty-fold reduction in message transfer times, resulting in a single word transfer rate of $\Delta(1) \simeq 300\mu$sec and an asymptotic rate of $\Delta(\infty) \simeq 1.4\mu$sec / word. The nodes are based upon Intel 80386/80387 processor/coprocessors with a floating point execution rate of roughly 0.1 MFLOPS, coupled with attached vector processors which achieve 3-4 MFLOPS on standard vector operations and 10-12 MFLOPS on matrix-matrix products. Typical performance for the spectral element code is 2-3 MFLOPS per node, including communication overhead.

5 Results

We have considered three classes of problems: *(i)* two-dimensional periodic geometry (e.g. Fig. 1), *(ii)* two-dimensional flow past 9 successive cylinders with parabolic velocity profile at inflow and standard Neumann boundary conditions at outflow, *(iii)* three-dimensional extensions of Fig.1 with and without endwalls. We briefly discuss the results of the class *(i)* and *(ii)* problems, and present computing times for all three classes.

Numerous simulations have been carried out for the two-dimensional, streamwise periodic flows, ranging from $Re = 0$ to $Re = 600$ [8]. In the range $Re > 0$ to $Re = Re_c$, the initially transient flow field evolves to a steady state. For subcritical Reynolds numbers close to Re_c the predominant part of the transient takes the form of an exponentially decaying sine wave, as seen for example in Fig. 3a, which is a time history of the transverse component of velocity at a point 3 cylinder diameters downstream of the center of the cylinder for $Re = 100$. As the Reynolds number approaches Re_c, the decay rate decreases; by extrapolating to zero decay rate, we estimate that $Re_c = 136$. We believe this number to be good to within 5%. However, non-splitting calculations should be undertaken before passing final judgement.

Above Re_c, the flow no longer settles to a steady state. Rather, it transitions



(a)

(b)

(c)

(d)

Figure 3: Time history traces of transverse velocity component at a point three diameters downstream of the cylinder center: (a) Re=100, (b) Re=200, (c) Re=600. Fig. (d) is the comparison between the periodic and inflow-outflow calculation for $Re = 140$ at a point 4.166 diameters downstream (of the 6th cylinder for the inflow-outflow case). The dashed curve represents the difference between the two traces; the convective nature of the instability can be observed by the sudden termination of the disturbance for the inflow-outflow configuration at a time $t_c \simeq 50$.

a steady periodic state for Re near Re_c, as evidenced by the time history trace of Fig. 3b for $Re = 200$. A fit of the saturation amplitude versus Reynolds number for several Reynolds numbers slightly greater than Re_c, showed the saturation amplitude is to vary as $(Re - Re_c)^{.59}$ [9]. Fig. 4 shows a typical streamline pattern at this Reynolds number, indicating the presence of a travelling wave moving from left to right, with wavelength $\lambda = L/2$. Over the range $Re = 100$ to $Re = 200$, the Strouhal number was found to be nearly constant, with a value of $St = \frac{2fh}{3\bar{V}} = 0.188$.

At $Re = 600$, the flow ceases to exhibit the single mode behavior found at Reynolds numbers near Re_c; Fig. 3c shows the erratic behavior of the velocity signal at the point of the previous figures.

Experimental results have suggested that the observed two-dimensional instability is of a convective nature. That is, the disturbance will grow (to a saturated state) in a frame of reference moving with the flow, but will not be self-sustaining at a fixed location in a laboratory setup, if there is not sufficient "noise" in the inlet profile. To examine this possibility, a second set of simulations have been carried out in which the domain is no longer periodic, but of fixed length, $L_9 = 9.5L$, with an array of nine successive cylinders to provide a periodic geometric disturbance to the otherwise plane Poiseuille channel flow, as depicted in Fig. 5. A parabolic inlet profile was specified, with the usual Neumann outflow boundary conditions on velocity at the exit [9].

With initial conditions identical to those for the periodic case, time history traces behind each of the cylinders was found to be identical (as shown in Fig. 3d) to the periodic results up to a time $t_c \simeq l/\bar{V}$, at which point the signal would fall off

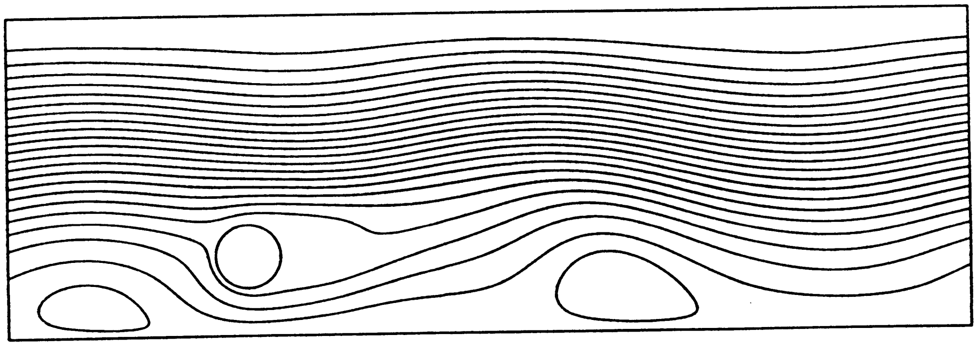


Figure 4: Streamlines for the supercritical Reynolds number $Re = 200$ reveal the travelling wave nature of the instability, with wavelength $\lambda = L/2$.

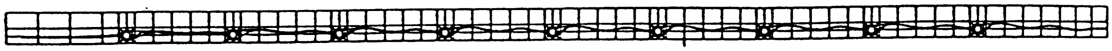


Figure 5: Multi-cylinder geometry ($K = 315$) and resultant streamlines for $Re = 160$.

rapidly to a constant value. Here, l is the distance downstream of the first cylinder. It is, in principle, possible to increase l to the point where t_c will be greater than the time required to reach a saturated periodic state. Thus, one can conclude that the periodic simulations discussed above provide relevant analysis for this physically realizable situation. Furthermore, in the presence of noise, experimental results [10] suggest that the periodic solution appears relatively quickly downstream.

6 Timings

We present in Table 1 the run time required for several of the test cases considered. All problems were computed on the Intel iPSC/2-VX, save case (a), which was solved on a DECStation 3100. Case (a) is a steady Stokes flow calculation which is discretized using consistent spaces for the velocity and pressure and which is solved using the Uzawa algorithm [5]. Case (e) is the nine cylinder calculation. For this problem, the CPU time per time-step, per processor, is roughly 15 times that required for the corresponding single cylinder case, (c). This is consistent with the increased number of degrees of freedom and increased condition number of the system matrices, as well as the counter-balancing effect of increased efficiency obtained due to the larger amount of work per processor required for the multi-cylinder case.

Case	Re	d	K	N	t_{final}	#time-steps	$\frac{seconds}{time-step}$	#-procs	CPU (hrs)
a	0	2	39	10	∞	1	228.	DEC	.063
b	100	2	39	10	95.5	26920	2.45	8	18.3
c	160	2	33	8	200.0	40000	2.50	8	28.0
d	600	2	39	10	200.0	79280	2.85	8	62.7
e	160	2	315	8	82.5	16540	18.8	16	86.5
f	150	3	99	8	200.0	40000	25.2	32	280.

Table 1: Computer performance for the eddy promoter problem.

Acknowledgements

We wish to thank Michael Schatz, Randall Tagg, and Harry Swinney of the University of Texas at Austin for their assistance and for the insight provided through their experimental work. This work was supported by the ONR and DARPA under contracts N00014-85-K-0208, N00014-88-K-0188, and N00014-89-J-1610, by NSF under Grants DMC-8704357, ASC-8806925, and Cooperative Agreement No. CCR-8809615, and by Intel Scientific Computers.

References

[1] S.A. Orszag and L.C. Kells (1980): Transition to turbulence in plane Poiseuille flow and plane Couette flow. *J. Fluid Mech.* 96, 159-205

[2] N.K. Ghaddar, G.E. Karniadakis, and A.T. Patera (1986): A conservative isoparametric spectral element method for forced convection; application to fully developed flow in periodic geometries. *Num. Heat Trans.* 9, 277-300

[3] R. Temam (1977): *Navier-Stokes Equations* (North-Holland, Amsterdam)

[4] A.T. Patera (1984): A spectral element method for fluid dynamics; Laminar flow in a channel expansion. *J. Comput. Phys.*, 54, 468-488.

[5] Y. Maday and A.T. Patera (1989): "Spectral element methods for the Navier-Stokes equations," in *State of the Art Surveys on Computational Mechanics* Eds. A.K. Noor and J.T. Oden, (ASME, New York), 71-143

[6] G.H. Golub and C.F. Van Loan (1983): *Matrix Computations*, (John Hopkins University Press, Baltimore, Maryland)

[7] P.F. Fischer and A.T. Patera (1991): Parallel Spectral Element Solution of the Stokes Problem. *J. Comput. Phys.*, **92**

[8] G.E. Karniadakis, B.B. Mikic, and A.T. Patera (1988): Minimum-dissipation transport enhancement by flow destabilization: Reynolds' analogy revisited. *J. Fluid Mech.*, **192**, 365-391

[9] M.F. Schatz, R.P. Tagg, H.L.Swinney, P.F. Fischer, and A.T. Patera (1990): Supercritcal transition in plane channel flow with spatially periodic perturbations. submitted to Phys. Rev. Lett.

[10] H. Kozlu (1989): *Experimental Investigation of Optimal Heat Removal From a Surface*. Ph.D. Thesis, Massachusetts Institute of Technology