INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2002; **40**:1133–1144 (DOI: 10.1002/fld.400)

LS-DYNA and the 8:1 differentially heated cavity

Mark A. Christon*,†

Computational Physics R&D Department, Sandia National Laboratories, M/S 0819, P.O. Box 5800, Albuquerque, 87185-0819 New Mexico

SUMMARY

This paper presents results computed using LS-DYNA's new incompressible flow solver for a differentially heated cavity with an 8:1 aspect ratio at a slightly super-critical Rayleigh number. Three Galerkin-based solution methods are applied to the 8:1 thermal cavity on a sequence of four grids. The solution methods include an explicit time-integration algorithm and two second-order projection methods—one semi-implicit and the other fully implicit. A series of *ad hoc* modifications to the basic Galerkin finite element method are shown to result in degraded solution quality with the most serious effects introduced by row-sum lumping the mass matrix. The inferior accuracy of a lumped mass matrix relative to a consistent mass matrix is demonstrated with the explicit algorithm which fails to obtain a transient solution on the coarsest grid and exhibits a general trend to under-predict oscillation amplitudes. The best results are obtained with semi-implicit and fully implicit second-order projection methods where the fully implicit method is used in conjunction with a 'smart' time integrator. Copyright © 2002 John Wiley & Sons, Ltd.

KEY WORDS: finite elements; LS-DYNA; thermal convection; CFD; Boussinesq equations; incompressible flow

1. INTRODUCTION

LS-DYNA is a multi-physics finite element code that provides a comprehensive set of simulation capabilities for problems ranging from automotive crash-worthiness and occupant safety, metal forming and fluid–structure interaction to heat transfer, compressible and incompressible flow. LS-DYNA's new incompressible CFD capability is applied to the 8:1 thermal cavity problem with $Ra = 3.4 \times 10^5$, $Pr = 0.71$ as defined in the paper by Christon, Gresho and Sutton in this special issue [1].

The thermal cavity problem is treated with an explicit time-integration method in addition to semi-implicit and fully implicit second-order projection methods (see Table I). The explicit time-integration method provides a baseline for comparison with the second-order projection methods. A number of variations on the formulation and time-integration methods have been experimented with in the context of the projection methods, and here a subset of the results

* Correspondence to: M. A. Christon, Computational Physics R&D Department, Sandia National Laboratories, M/S 0819, P.O. Box 5800, Albuquerque, New Mexico 87185-0819.
† E-mail: machris@sandia.gov

 Received 31 December 2001

Table I. Summary of LS-DYNA solution methods applied to the 8:1 thermal cavity problem.

	Time integration methods		
	Explicit	Semi-implicit	Fully implicit
Mass matrix	Lumped	Lumped/consistent	Lumped/consistent
Advection	1-pt. centroid	1-pt. centroid or 2×2 quadrature	2×2 quadrature
BTD	$\Delta t/2\ u_i u_j$	$3\Delta t/8\ u_i u_j$	N/A
Diffusion	1-pt.+ hourglass stabilization	2×2 quadrature	2×2 quadrature
Buoyancy	1-pt. lumped	Lumped or 2×2 consistent	Lumped or 2×2 consistent

from this experimentation is reported. The fully implicit projection method is new and combines a fully implicit advection treatment in the context of a second-order, incremental projection method with a variable-step time integrator based on a fixed CFL condition or temporal error control.

In the following section, a brief summary of the three time-integration methods used for the 8:1 thermal cavity is presented. In Section 3 the compulsory results are presented with a summary of the computational performance of the three solution methods. The sensitivity of the finite element operators to *ad hoc* 'tricks-of-the-trade' and the associated deleterious effects on the solution quality for the 8:1 thermal cavity are outlined. Finally, Section 4 presents a brief summary and conclusions.

2. FORMULATION ISSUES

LS-DYNA provides multiple segregated solution strategies for the incompressible Navier–Stokes that derive from a Galerkin finite element formulation. In this effort, attention has been restricted to the Q1Q0 element formulation. The Q1Q0 element provides a bilinear representation of the velocity and temperature with a piecewise constant pressure, i.e. node-centred velocities and temperature with an element-centered pressure.

Explicit time-integration method: The explicit time-integration method follows the formulation presented in Christon [2, 3] and uses single-point Gaussian quadrature in conjunction with a row-sum lumped mass matrix and hourglass stabilization. In this formulation, the advective terms are treated with a second-order explicit scheme that relies on balancing-tensor diffusivity (BTD), a form of streamline upwinding, while the viscous and thermal diffusion terms are treated with a first-order time integrator yielding conditional stability. In a one-dimensional sense, the stable time step is

$$\Delta t \leqslant \frac{\Delta x^2}{\gamma[1 + \sqrt{1 + (u\Delta x/\gamma)^2}]} \tag{1}$$

where $\gamma = \sqrt{Pr/Ra}$ for the Navier–Stokes equations, and $\gamma = 1/\sqrt{Ra\,Pr}$ for the energy equation. The explicit method constitutes a blend of finite element and finite difference philosophies with a concomitant cost in accuracy—second-order in space, second-order in time and phase accuracy for the advective terms, and first-order in time for the viscous/diffusion terms.

Semi-implicit time-integration method: The semi-implicit projection method derives from the optimal second-order projection method of Gresho and Chan [4, 5]. The semi-implicit algorithm uses Crank–Nicholson for the viscous/diffusion terms while the advective terms are treated explicitly. The associated BTD terms, that are necessary for second-order temporal accuracy, are treated via Crank–Nicholson as well. This algorithm remains stable for CFL numbers of $O(5–10)$. The compulsory data reported in Section 3 were computed using the semi-implicit projection method with all element-level operators integrated with 2×2 Gaussian quadrature. However, experimentation with centroid advection has indicated that this modification exhibits no significant side-effects on the 8:1 cavity problem.

Fully implicit time-integration method: The fully implicit projection method uses a simple linearization of the advective terms with a discrete advective operator that is essentially skew-symmetric. A variety of choices for the linearized velocities in the advective terms have been tested with satisfactory results obtained using the simplest choice for the advective velocity field, i.e. the lagged velocities. The implicit treatment of the skew-symmetric advection operator is unique because it provides unconditional stability without resorting to non-linear solution schemes, e.g. Newton's method (see Gresho and Sani [6, pp. 797–800]).

In the fully implicit projection method, a Crank–Nicholson time integrator is applied to all terms in the linearized momentum and energy equations except for the pressure gradient. There are no additional stabilizing terms, e.g. BTD, that modify the Galerkin finite element formulation. Automatic time-step selection for this algorithm is based on either a user-specified CFL condition or control of the local truncation error in time.

Both the semi-implicit and fully implicit methods are considered to be second-order in space and time with fourth-order phase accuracy when a consistent mass matrix is used. A consistent mass matrix was used for all of the compulsory data reported in Section 3 for the projection methods. The details of the semi-implicit and fully implicit formulations and implementation in LS-DYNA may be found in Christon [3].

Pressure-Poisson equation and linear algebra: In all three time-integration methods, a pressure-Poisson equation (PPE) must be solved at each time step. A single factorization is performed during initialization with one re-solve per time-step. Bandwidth/profile minimization is used for the PPE making the direct solve particularly efficient due to the 8:1 aspect ratio in the mesh, i.e. a minimum bandwidth corresponds approximately to the mesh density in the x-direction. The direct pressure solve yields an RMS divergence that is machine-zero for all calculations presented here.

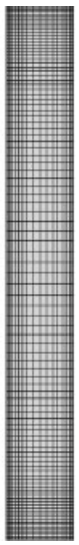
The semi-implicit projection method uses a Jacobi-preconditioned conjugate gradient solver for the momentum and energy equations. In order to treat the non-symmetric operators that derive from implicit advection treatment, the fully implicit projection method uses a Jacobi-preconditioned conjugate gradient-squared solver. The solution convergence is defined by both relative and residual error norms, i.e., $\|\Delta x\|/\|x\| \leqslant \varepsilon$ and $\|b - Ax\|/\|b\| \leqslant \varepsilon$. Testing has shown that the solution metrics for this benchmark problem are not sensitive to the convergence criteria for $\varepsilon \leqslant 10^{-5}$.

3. RESULTS

This section summarizes the compulsory results computed using LS-DYNA along with the computational resources required for the calculations. Although some testing was done

(a)

Grid	Resolution	Δx_{min}	Δy_{min}
A	21×101	1.725e-2	2.987e-2
B	41×201	8.175e-3	1.479e-2
B+	61×201	5.357e-3	1.479e-2
C	81×401	3.977e-3	7.369e-3


(b)

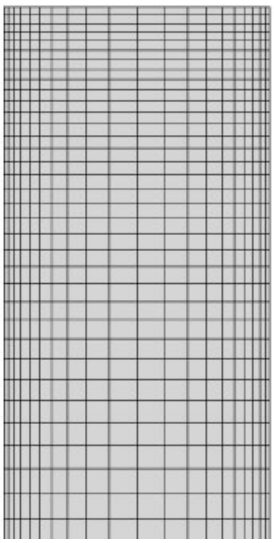

(c)

Figure 1. Grids used for the 8:1 thermal: (a) grid resolution and the minimum grid spacing in the x- and y-co-ordinate directions, (b) 21×101 grid, (c) close-up of the top of the 21×101 grid.

with initial conditions that are not skew-symmetric, all compulsory data reported here was computed using the skew-symmetric initial conditions defined for the benchmark problem by Christon *et al.* [1].

Grid resolution: Four graded meshes have been applied to the benchmark problem with the grid resolution shown in Figure 1. In each grid, four fixed mesh lines were used to insure that the *hard points* used to record the compulsory time-history data were identical for all grids. The minimum mesh size in the x-direction for the coarse grid was chosen to place approximately 2 grid points inside the boundary layers on the vertical walls based on the scaling estimates of Gill [7]). Grids A, B and C were generated in a sequence by approximate grid-doubling—the *hard points* used to record the compulsory time-history data and the mesh grading preclude exact grid-doubling. For grid B+, the horizontal resolution was intentionally increased only in the regions between vertical walls and the hard-lines, i.e. $0 \leqslant x \leqslant 0.1810$ and $0.8190 \leqslant x \leqslant 1$.

Compulsory results: Temperature time history plots at 'point-1' (defined in Christon *et al.* [1]) are shown in Figure 2 for all four grids for the explicit method and in Figure 3 for the semi-implicit projection method. The time histories in Figures 2 and 3 reveal several significant differences between the two methods. The explicit algorithm finds a steady-state solution on the coarsest grid (21×101) as shown in Figure 2(a). On grids B, B+ and C, the explicit method also takes longer to achieve a periodic solution with a fixed oscillation amplitude relative to the projection methods, and the oscillation amplitudes in the temperature, as well as, in the velocity, vorticity and pressure tend to be smaller. These numerical artifacts are thought to be attributable to the lumped mass matrix and BTD.

In contrast, the semi-implicit projection method appears to be somewhat more 'energetic' yielding larger temperature oscillations and achieving a periodic solution relatively sooner than the explicit method. With the exception of grid B+, the time history plots in Figures 2

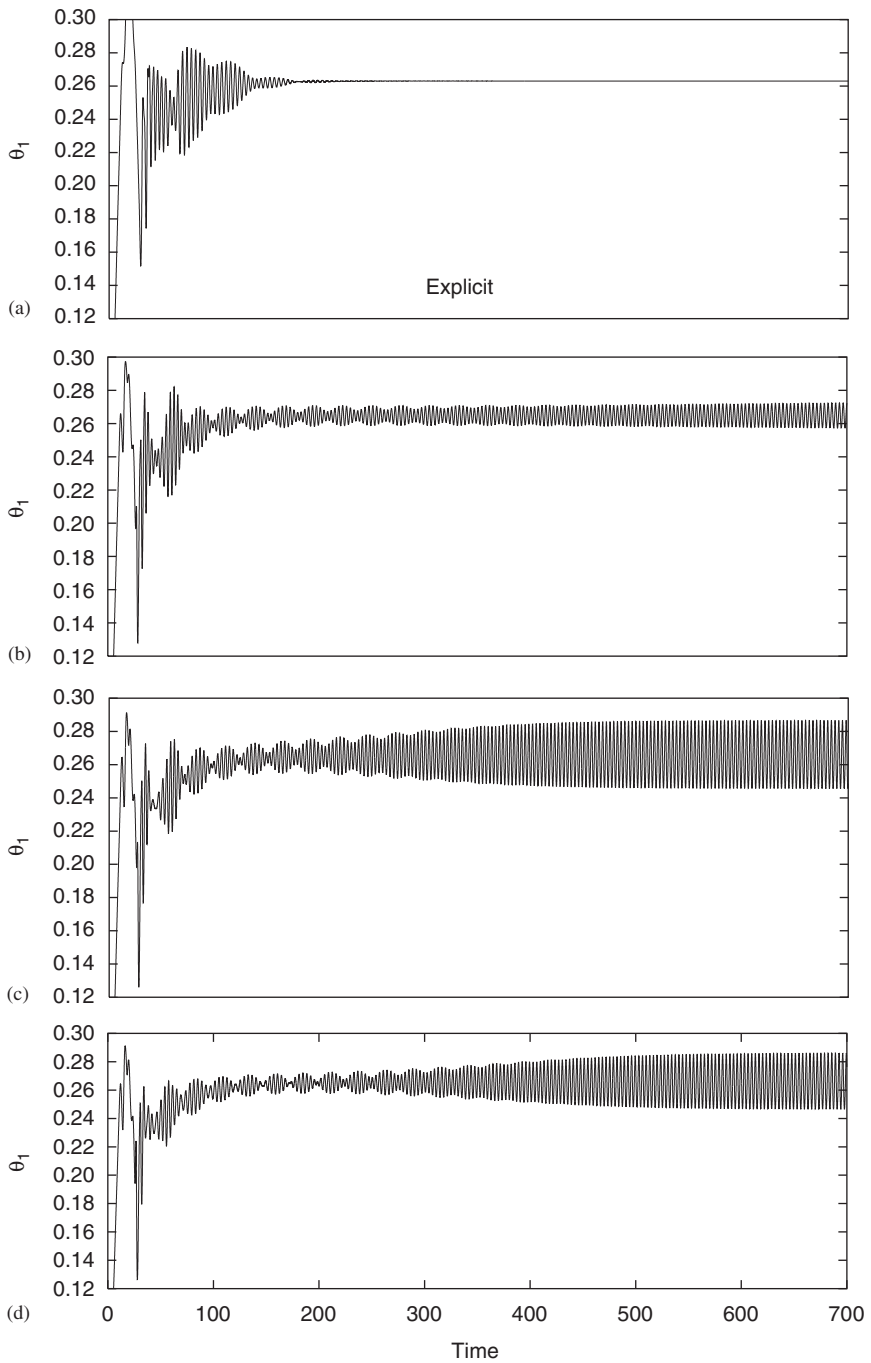


Figure 2. Temperature time history at point-1 using the explicit algorithm on (a) grid-A (21×101), (b) grid-B (41×201), (c) grid-B+ (61×201), and (d) grid-C (81×401).

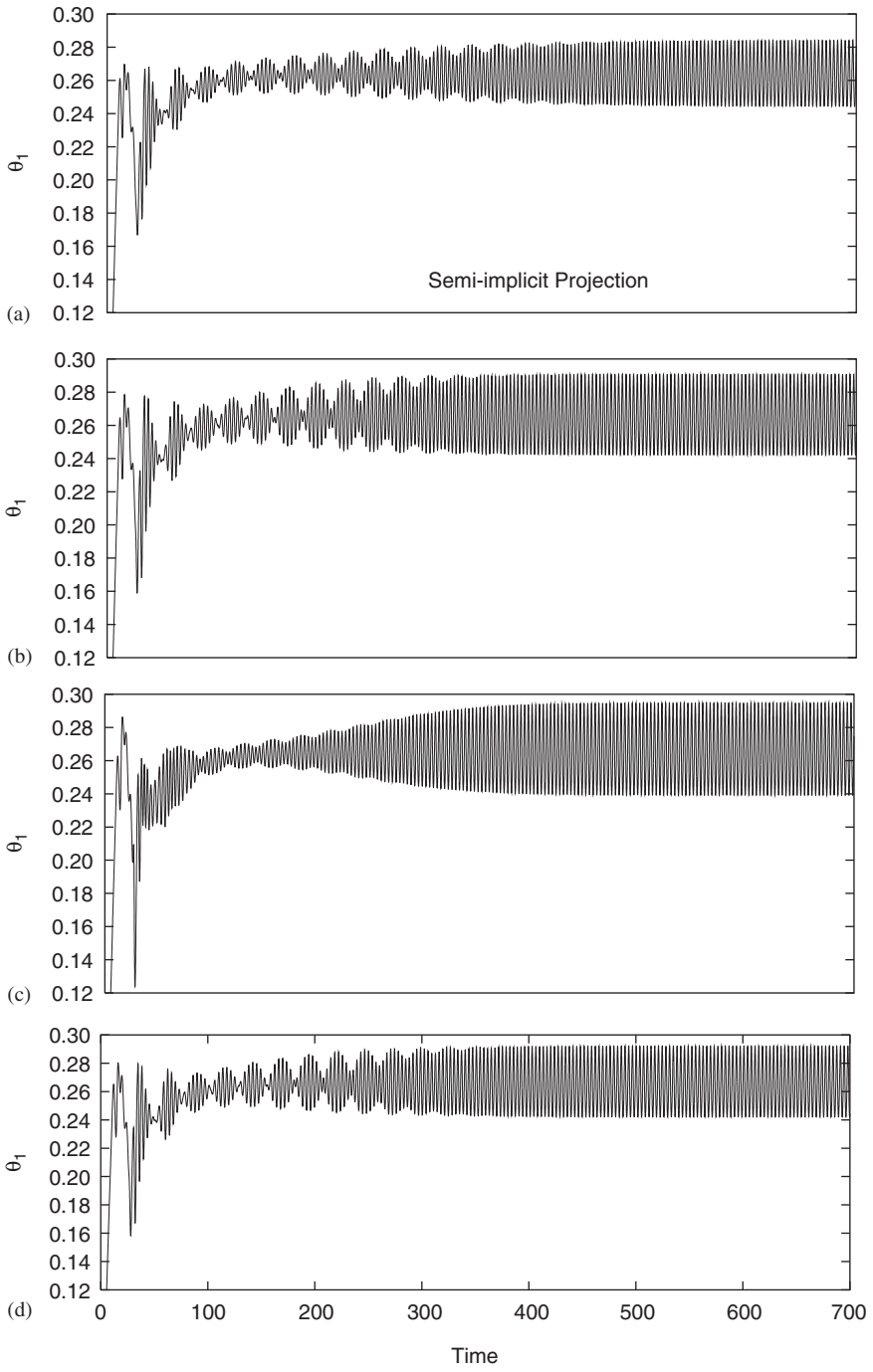


Figure 3. Temperature time history at point-1 using the semi-implicit projection algorithm with $\Delta t = 0.05$ on (a) grid-A (21×101), (b) grid-B (41×201), (c) grid-B+ (61×201), and (d) grid C (81×401).

and 3 show that the amplitude of the periodic temperature oscillation appears to be converging from below for both the explicit and semi-implicit projection methods. The time-history plots for grid B+ in Figures 2(c) and 3(c) reveal the sensitivity to changes in the boundary layer resolution highlighting the fact that grid convergence has not been achieved in these computations.

The compulsory time history, global and wall data are tabulated for the explicit method in Table II and for the semi-implicit projection method in Tables III and IV. For all computations, the average, amplitude and period were measured for a duration of approximately 1000 time units ($1000 \leqslant t \leqslant 2000$). For the explicit method, the stability-limited time-steps for the four grids were $\Delta t_{\mathrm{A}} = 4.0\text{e-}2$, $\Delta t_{\mathrm{B}} = 9.25\text{e-}3$, $\Delta t_{\mathrm{B+}} = 4.0\text{e-}3$, and $\Delta t_{\mathrm{C}} = 2.0\text{e-}3$. This corresponds to 385, 872 and 1725 time-steps per period on grids B, B+, and C, respectively. For the semi-implicit projection method, fixed time steps of 0.1 and 0.05 were used corresponding to approximately 35 and 69 time-steps per period, respectively, for each of the four grids.

For all computations, the skewness error metric ε_{12} was computed, but was found to be machine zero for the duration of all computations indicating a skew-symmetric temperature field. A global energy balance ($Q_{\mathrm{in}} - Q_{\mathrm{out}}$) was monitored for all computations and found to be machine zero as well. A series of tests were conducted using the skew-symmetric boundary conditions with the initial temperature field randomly perturbed about the average value by $\pm 20\%$. In all cases, the skewness error metric achieved a maximum of 0.01 at around 2000 time units and then decayed with time until the skew-symmetric periodic solution was recovered at approximately 5000 time units.

The fully implicit algorithm was applied to the 8:1 cavity with three time-integration options: fixed time-step, variable time-step based on a fixed CFL condition, and variable time-step based on local time truncation error. The fixed time-step and CFL methods yielded results similar to the semi-implicit results with similar computational costs. The variable time-step algorithm was applied using an error tolerance of $\varepsilon = 2.5\text{e-}4$, maximum time step of $\Delta t_{\max} = 0.02$, and maximum scale factor in the time step of $\mathrm{DTSF} = 1.25$ (see Reference [3] for details). The temperature and time-step time histories, and compulsory data are shown in Figure 4 for grid-A. During the computation on the coarse-grid, the error control constrained the time-step so that $\mathrm{CFL} \leqslant 0.5$ and approximately 85 time-steps per period were used for the periodic phase of the solution. The variation of the time-step in Figure 4(b) shows that the time-step grew by over a factor of 10 during the startup transient and then asymptoted to $\Delta t \approx 0.035$—slightly less than the fixed $\Delta t = 0.05$ used with the semi-implicit and fully implicit projection methods.

Computational resources: All computations were performed in serial (single-processor) with fully optimized code on a DEC Alpha 500 AU with a clock rate of 500 MHz, total memory of 256 Mbytes, and specfp95 rating of 19. Table V shows the memory footprint and CPU usage statistics for the explicit, semi-implicit and fully implicit methods. Most notable is the small additional memory cost of the semi-implicit and fully implicit projection methods relative to the explicit time-integration method.

In order to compare the CPU time between the methods, consider that the explicit method required 1725 time-steps per period of oscillation using grid-C. In contrast, the semi-implicit and fully implicit projection methods required only 69 time-steps per period. The consequence being that while the explicit algorithm requires about 8.8 times less CPU time per time-step, the strict stability limits results in the explicit algorithm being about 2.8 times more expensive per period of oscillation. This is due primarily to the stability limits for the explicit algorithm

Table II. Compulsory point, wall and global data computed using the explicit time-integration method. The time-steps based on stability corresponding to the four grids are: $\Delta t_A = 4.0\text{e-}2$, $\Delta t_B = 9.25\text{e-}3$, $\Delta t_{B+} = 4.0\text{e-}3$, and $\Delta t_C = 2.0\text{e-}3$.

Time history quantity	Grid: 21×101 Δx_{min}: 1.725e-2 Δy_{min}: 2.987e-2 Duration: $T_A \approx 1000.0$ Steps/period: –			Grid: 41×201 Δx_{min}: 8.175e-3 Δy_{min}: 1.479e-2 Duration: $T_B \approx 1000.0$ Steps/period: $N_{per_B} = 385$		
	Avg.	Amp.	Period	Avg.	Amp.	Period
u_1	6.098e-2	—	—	5.657e-2	2.404e-2	3.562
v_1	0.4787	—	—	0.4663	3.648e-2	3.562
θ_1	0.2630	—	—	0.2649	1.900e-2	3.562
ε_{12}	0.0000	—	—	0.0000	—	—
ψ_1	−7.482e-2	—	—	−7.480e-2	3.038e-3	3.562
ω_1	−2.218	—	—	−2.422	0.4326	3.562
ΔP_{14}	−1.407e-3	—	—	−1.996e-3	9.152e-3	3.562
ΔP_{51}	−0.5239	—	—	−0.5315	9.660e-3	3.562
ΔP_{35}	0.5253	—	—	0.5335	4.230e-3	3.562
$Nu\|_{x=0}$	4.579	—	—	4.578	3.460e-3	3.562
$Nu\|_{x=W}$	−4.579	—	—	−4.578	3.460e-3	3.562
$\hat{u}$	0.2449	—	—	0.2405	3.900e-5	3.565
$\hat{\omega}$	3.000	—	—	3.014	1.460e-3	3.562

	Grid: 61×201 Δx_{min}: 5.357e-3 Δy_{min}: 1.479e-2 Duration: $T_{B+} \approx 1000.0$ Steps/period: $N_{per_{B+}} = 872$			Grid: 81×401 Δx_{min}: 3.977e-3 Δy_{min}: 7.369e-3 Duration: $T_C \approx 1000.0$ Steps/period: $N_{per_C} = 1725$		
	Avg.	Amp.	Period	Avg.	Amp.	Period
u_1	6.195e-2	5.240e-2	3.492	6.112e-2	5.078e-2	3.447
v_1	0.4665	7.475e-2	3.493	0.4638	7.260e-2	3.446
θ_1	0.2661	4.092e-2	3.493	0.2663	3.975e-2	3.447
ε_{12}	0.0000	—	—	0.0000	—	—
ψ_1	−7.400e-2	6.616e-3	3.493	−7.405e-2	6.482e-3	3.446
ω_1	−2.292	1.0062	3.492	−2.320	0.9866	3.446
ΔP_{14}	−2.421e-3	1.934e-2	3.493	−2.328e-3	1.891e-2	3.446
ΔP_{51}	−0.5328	2.040e-2	3.493	−0.5332	2.062e-2	3.446
ΔP_{35}	0.5357	9.246e-3	3.493	0.5361	9.198e-3	3.446
$Nu\|_{x=0}$	4.578	7.040e-3	3.493	4.579	6.720e-3	3.446
$Nu\|_{x=W}$	−4.578	7.040e-3	3.494	−4.579	6.720e-3	3.446
$\hat{u}$	0.2401	4.100e-5	3.493	0.2398	3.500e-5	3.446
$\hat{\omega}$	3.015	3.040e-3	3.493	3.016	3.160e-3	3.446

which results in the computational cost increasing by a factor 16 for each factor of 2 increase in grid resolution.

Some failures: During the course of testing various algorithmic combinations, several failures were encountered that are briefly summarized here. The use of a lumped mass matrix

Table III. Compulsory point, wall and global data computed using the semi-implicit projection method with $\Delta t = 0.10$ (≈ 35 time-steps per period). All measurements were performed over a duration of ≈ 1000 time units.

Time history quantity	Grid: 21×101 $\Delta x_{\min}$: 1.725e-2 $\Delta y_{\min}$: 2.987e-2			Grid: 41×201 $\Delta x_{\min}$: 8.175e-3 $\Delta y_{\min}$: 1.479e-2		
	Avg.	Amp.	Period	Avg.	Amp.	Period
u_1	5.696e-2	4.804e-2	3.615	6.492e-2	6.368e-2	3.542
v_1	0.4544	6.900e-2	3.615	0.4576	8.790e-2	3.544
θ_1	0.2641	3.856e-2	3.614	0.2663	5.022e-2	3.544
ε_{12}	0.0000	—	—	0.0000	—	—
ψ_1	−7.371e-2	6.204e-3	3.614	−7.296e-2	8.160e-3	3.543
ω_1	−2.161	0.8440	3.614	−2.159	1.2288	3.544
ΔP_{14}	−4.245e-3	1.888e-2	3.614	−3.797e-3	2.450e-2	3.543
ΔP_{51}	−0.5347	1.658e-2	3.614	−0.5340	2.436e-2	3.543
ΔP_{35}	0.5391	1.127e-2	3.614	0.5384	1.472e-2	3.544
$Nu\|_{x=0}$	4.562	6.710e-3	3.614	4.560	8.690e-3	3.544
$Nu\|_{x=W}$	−4.562	6.710e-3	3.614	−4.560	8.690e-3	3.544
$\hat{u}$	0.2390	1.220e-4	3.614	0.2382	1.110e-4	3.544
$\hat{\omega}$	2.964	3.230e-3	3.614	2.993	3.930e-3	3.544

	Grid: 61×201 $\Delta x_{\min}$: 5.357e-3 $\Delta y_{\min}$: 1.479e-2			Grid: 81×401 $\Delta x_{\min}$: 3.977e-3 $\Delta y_{\min}$: 7.369e-3		
	Avg.	Amp.	Period	Avg.	Amp.	Period
u_1	6.715e-2	6.842e-2	3.531	6.636e-2	6.572e-2	3.526
v_1	0.4581	9.300e-2	3.531	0.4584	9.008e-2	3.528
θ_1	0.2670	5.384e-2	3.531	0.2670	5.164e-2	3.529
ε_{12}	0.0000	—	—	0.0000	—	—
ψ_1	−7.263e-2	8.732e-3	3.530	−7.281e-2	8.412e-3	3.529
ω_1	−2.133	1.3380	3.531	−2.171	1.2946	3.529
ΔP_{14}	−3.693e-3	2.602e-2	3.530	−3.628e-3	2.518e-2	3.529
ΔP_{51}	−0.5341	2.618e-2	3.530	−0.5339	2.570e-2	3.529
ΔP_{35}	0.5385	1.577e-2	3.530	0.5382	1.506e-2	3.529
$Nu\|_{x=0}$	4.560	9.290e-3	3.531	4.560	8.980e-3	3.529
$Nu\|_{x=W}$	−4.560	9.290e-3	3.531	−4.560	8.980e-3	3.529
$\hat{u}$	0.2381	1.120e-4	3.531	0.2380	1.100e-4	3.528
$\hat{\omega}$	2.996	4.110e-3	3.531	2.999	4.010e-3	3.529

in the semi-implicit projection method yields a steady solution only on the coarsest grid (21×101). For both projection methods, the buoyancy forces require a 2×2 Gaussian quadrature rule and no row-sum lumping. Row-sum lumping the buoyancy forces while retaining a consistent mass matrix in the time-integrator yielded oscillation periods that were 20% longer than for the consistent-mass version on the grid-A (21×101).

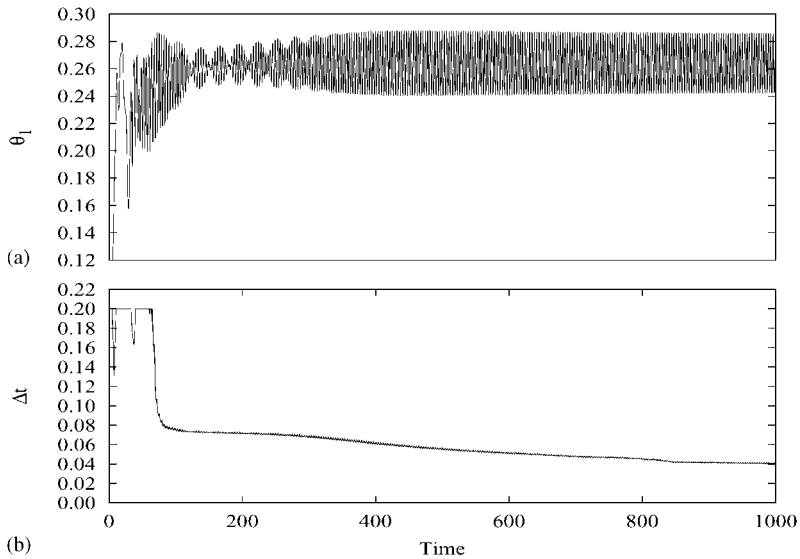
The use of BTD in the semi-implicit projection method yielded preliminary results in which the temperature oscillation amplitude and period were under-predicted relative to the explicit

Table IV. Compulsory point, wall and global data computed using the semi-implicit projection method with $\Delta t = 0.05$ (≈ 69 time-steps per period). All measurements were performed over a duration of ≈ 1000 time units.

Time history quantity	Grid: 21×101 $\Delta x_{\min}$: 1.725e-2 $\Delta y_{\min}$: 2.987e-2			Grid: 41×201 $\Delta x_{\min}$: 8.175e-3 $\Delta y_{\min}$: 1.479e-2		
	Avg.	Amp.	Period	Avg.	Amp.	Period
u_1	5.713e-2	4.996e-2	3.556	6.399e-2	6.228e-2	3.485
v_1	0.4566	7.150e-2	3.556	0.4600	8.612e-2	3.486
θ_1	0.2643	3.984e-2	3.555	0.2665	4.866e-2	3.486
ε_{12}	0.0000	—	—	0.0000	—	—
ψ_1	−7.385e-2	6.4580-3	3.556	−7.326e-2	7.978e-3	3.487
ω_1	−2.152	0.8816	3.555	−2.181	1.2032	3.485
ΔP_{14}	−3.727e-3	1.920e-2	3.556	−3.234e-3	2.346e-2	3.487
ΔP_{51}	−0.5346	1.824e-2	3.556	−0.5341	2.432e-2	3.486
ΔP_{35}	0.5387	1.011e-2	3.556	0.5380	1.248e-2	3.487
$Nu\|_{x=0}$	4.571	6.670e-3	3.556	4.569	8.170e-3	3.487
$Nu\|_{x=W}$	−4.571	6.670e-3	3.556	−4.569	8.170e-3	3.487
$\hat{u}$	0.2400	7.400e-5	3.556	0.2391	7.400e-5	3.487
$\hat{\omega}$	2.970	3.110e-3	3.556	3.000	3.690e-3	3.487
	Grid: 61×201 $\Delta x_{\min}$: 5.357e-3 $\Delta y_{\min}$: 1.479e-2			Grid: 81×401 $\Delta x_{\min}$: 3.977e-3 $\Delta y_{\min}$: 7.369e-3		
	Avg.	Amp.	Period	Avg.	Amp.	Period
u_1	6.839e-2	7.292e-2	3.464	6.537e-2	6.432e-2	3.471
v_1	0.4638	9.870e-2	3.464	0.4608	8.832e-2	3.471
θ_1	0.2670	5.564e-2	3.464	0.2671	5.004e-2	3.471
ε_{12}	0.0000	—	—	0.0000	—	—
ψ_1	−7.293e-2	9.238e-3	3.464	−7.313e-2	8.228e-3	3.471
ω_1	−2.116	1.4162	3.464	−2.195	1.2684	3.471
ΔP_{14}	−2.705e-3	2.592e-2	3.464	−3.072e-3	2.412e-2	3.471
ΔP_{51}	−0.5334	2.566e-2	3.465	−0.5340	2.546e-2	3.471
ΔP_{35}	0.5357	1.272e-2	3.464	0.5379	1.286e-2	3.471
$Nu\|_{x=0}$	4.578	9.290e-3	3.464	4.569	8.410e-3	3.471
$Nu\|_{x=W}$	−4.578	9.290e-3	3.464	−4.569	8.410e-3	3.471
$\hat{u}$	0.2403	4.100e-5	3.464	0.2388	7.200e-5	3.471
$\hat{\omega}$	3.011	3.780e-3	3.464	3.006	3.780e-3	3.471

algorithm. Testing demonstrated that the implicit treatment of the BTD terms requires that the BTD coefficient (see Gresho *et al.* [8]) be reduced from 1/2 to 3/8—an *ad hoc* modification begging for additional analysis.

In the fully implicit projection method, a Galerkin time weighting (i.e. $\theta = 2/3$) with a consistent mass matrix was tested on the coarse-grid, and as expected, the backward-Euler bias proved excessively diffusive producing a steady-state solution. In fact, deviations as small



(c)

	u_1	v_1	θ_1	ϵ_{12}	ψ_1	ω_1	$Nu\|_{x=0}$	$\hat{u}$
Avg.	0.05867	0.45943	0.26413	0.0000	-7.394e-2	-2.1213	4.58022	0.24030
Amp.	0.05512	0.07893	0.04242	0.00000	7.106e-3	0.97100	0.00710	7.000e-5

Figure 4. Results computed using the fully implicit projection method with variable time-step algorithm with an error tolerance $\varepsilon = 2.5\text{e-}4$, time-step limit of $\Delta t_{\max} = 0.2$, and maximum time-step scale factor DTSF = 1.25: (a) temperature time history at point-1, (b) time-step history, and (c) compulsory data for grid-A (21×101)—period for temperature oscillation $T = 3.53617$ measured over ≈ 1000 time units.

Table V. Computational resources showing CPU time per grid point per time-step [μs/node/Δt] and memory [Mbytes]. The semi-implicit and fully implicit methods used a fixed $\Delta t = 0.05$ for all timing comparisons.

Time-integration method	Grid resolution							
	A: 21×101		B: 41×201		B+: 61×201		C: 81×401	
	CPU	Memory	CPU	Memory	CPU	Memory	CPU	Memory
Explicit	9.785	3.5	11.18	14.5	9.868	23.4	12.11	67.1
Semi-implicit	33.04	4.2	51.05	17.2	71.10	27.4	104.8	77.8
Fully implicit	41.91	4.2	62.93	17.2	61.10	27.4	107.8	77.8

as 1/100 from $\theta = 1/2$ in the time-weighting produced steady-state solutions—a surprising statement of how sensitive the problem is to increased diffusion.

4. SUMMARY AND CONCLUSIONS

The results of this effort have demonstrated the sensitivity of the 8:1 thermally driven cavity to the numerics of the solution method—particularly near the critical Rayleigh number. The

enhanced phase accuracy of the consistent mass matrix, both for the advective terms and the buoyancy forces, appears to play an important role in capturing the proper fluid dynamical behavior. Although not explored in great deal, the variable time-step fully implicit projection method is a viable alternative to semi-implicit and possibly fully coupled solution methods with little additional computational cost relative to the semi-implicit method.

REFERENCES

1. Christon MA, Gresho PM, Sutton SB. Computational predictability of natural convection flows in enclosures. *International Journal for Numerical Methods in Fluids* 2002; **40**:953–980.
2. Christon MA. A domain-decomposition message-passing approach to transient viscous incompressible flow using explicit time integration. *Computer Methods in Applied Mechanics and Engineering* 1997; **148**:329–352.
3. Christon MA, Cook Jr GO. *LS-DYNA's Incompressible Flow Solver—User's Manual*. Livermore Software Technology Corporation: Livermore, CA, USA, October 2000.
4. Gresho PM. On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix. Part 1: Theory. *International Journal for Numerical Methods in Fluids* 1990; **11**:587–620.
5. Gresho PM, Chan ST. On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix. Part 2: Implementation. *International Journal for Numerical Methods in Fluids* 1990; **11**:621–659.
6. Gresho PM, Sani RL. *Incompressible Flow and the Finite Element Method*, *Advection–Diffusion and Isothermal Laminar Flow*. Wiley, Chichester, England, 1998.
7. Gill AE. The boundary-layer regime for convection in a rectangular cavity. *Journal of Fluid Mechanics* 1966; **26**(3):515–536.
8. Gresho PM, Chan ST, Lee RL, Upson CD. A modified finite element method for solving the time-dependent, incompressible Navier–Stokes equations. Part 1: Theory. *International Journal for Numerical Methods in Fluids* 1984; **4**:557–598.

