Available online at www.sciencedirect.com

ScienceDirect

Journal of Computational Physics 223 (2007) 121–126

JOURNAL OF COMPUTATIONAL PHYSICS

www.elsevier.com/locate/jcp

# Enhanced algorithm efficiency for phase change convection using a multigrid preconditioner with a SIMPLE smoother

Katherine J. Evans [a,*], D.A. Knoll [a,1], Michael Pernice [b]

[a] Theoretical Division, Fluid Dynamics Group (T-3), Los Alamos National Laboratory, MS B216, Los Alamos, NM 87545, USA
[b] Computer, Computational, and Statistical Sciences Division, Modeling, Algorithms and Informatics Group (CCS-3),
Los Alamos National Laboratory, MS B265, Los Alamos, NM 87545, USA

Received 28 April 2006; received in revised form 29 August 2006; accepted 2 September 2006
Available online 2 November 2006

## Abstract

This note extends previous work that used the pressure correction method SIMPLE as a preconditioner to solve two-dimensional unsteady phase change problems. Here we examine the impact of using SIMPLE as a smoother to multigrid for such problems.
Published by Elsevier Inc.

*Keywords:* Newton–Krylov; Geometric multigrid; SIMPLE smoother; Phase change convection

In this note, a basic geometric multigrid (MG) algorithm is implemented as a preconditioner to the Jacobian-Free Newton–Krylov (JFNK) solution method with a pressure-correction smoother, SIMPLE [6]. Recent work using JFNK with SIMPLE as a preconditioner to solve two-dimensional unsteady phase change convection demonstrated that, compared to the more standard SIMPLE solution method, JFNK–SIMPLE could converge to a chosen nonlinear tolerance criterion two orders of magnitude more quickly [3]. However, as we extend the benchmark phase change convection problem to more complex and realistic configurations on finer grids, unacceptable computational expense is unavoidable. Therefore, we seek additional gains in efficiency within the existing structure of the JFNK–SIMPLE algorithm. Past efforts have demonstrated that multigrid is effective as a preconditioner to the solution of nonlinear problems on fine grids [5,9]. Also, the SIMPLE algorithm has been shown to be an effective smoother to MG for the solution of the Navier–Stokes equations [7,8]. However, the inclusion of a phase front within natural convection changes the nature of the equations to be solved. Applying MG within a preconditioner to these types of problems is novel.

Here, two-dimensional phase change convection is modeled with the incompressible Navier–Stokes and energy equations in an enthalpy framework, expressed in matrix operator form as

[*] Corresponding author. Tel.: +1 505 667 4613; fax: +1 505 665 5926.
*E-mail address:* kevans@lanl.gov (K.J. Evans).

[1] Present address: Idaho National Laboratory, Advanced Nuclear Energy Systems Integration, P.O. Box 1625, Idaho Falls, ID 83415, USA.

0021-9991/$ - see front matter Published by Elsevier Inc.
doi:10.1016/j.jcp.2006.09.003

$$\mathbf{Q}_1(\mathbf{v}) + \rho^{-1}\nabla(p) - \mathbf{f}(H) - A(H)\mathbf{v} = F(\mathbf{v}) \tag{1}$$

$$\nabla \cdot \mathbf{v} = F(p) \tag{2}$$

$$\mathbf{Q}_2(H) = F(H), \tag{3}$$

where

$$\mathbf{Q}_1 = \frac{\partial()}{\partial t} + \nabla \cdot (\mathbf{v}()) - \frac{1}{Re}\Delta() \tag{4}$$

$$\mathbf{Q}_2 = \frac{\partial()}{\partial t} + \nabla \cdot (\mathbf{v}()) - \frac{1}{RePr}\Delta\tau(). \tag{5}$$

The horizontal and vertical velocities ($\mathbf{v} = u, v$), pressure ($p$), and total enthalpy ($H$) make up the state vector $\mathbf{x} = \{u, v, p, H\}^{\mathrm{T}}$ of dependent variables to be solved at each time step. $\mathbf{F}(\mathbf{x})$ is the set of nonlinear residuals for $\mathbf{x}$. The buoyancy forcing is the Boussinesq approximation, $\mathbf{f}(H) = \{0, \frac{Ra}{Re^2 Pr}\tau(H)\}$.

The nondimensionalized problem in this study is defined by constant values of the Rayleigh number ($Ra = 3000$), Prandtl number ($Pr = 1000$), Reynolds number ($Re = 1$), specific heat ($c_{\mathrm{P}} = 1$), latent heat ($L = 1$), and density ($\rho = 1$) to match closely with previous numerical analyses of phase change convection [10,1]. $\mathbf{\Delta}$, $\mathbf{\nabla}$, and $\mathbf{\nabla}\cdot$ are the two-dimensional Laplacian, gradient, and divergence operators, respectively. For the present pure material simulation study, the total enthalpy is related to the temperature, $T$, by $H = c_{\mathrm{p}}T + (1 - \epsilon_{\mathrm{s}})L$. The amount of latent heat released is determined by the solid fraction of the material, $\epsilon_{\mathrm{s}}$, where $\epsilon_{\mathrm{s}} = 1$ is solid, $\epsilon_{\mathrm{s}} = 0$ is liquid, and $0 \leqslant \epsilon_{\mathrm{s}} \leqslant 1$ is within a 'mushy' zone at the melting temperature, $T_{\mathrm{m}}$. Thus, enthalpy is piecewise smooth using temperature as a function of $H$,

$$T = \tau(H) = \begin{cases} H/c_{\mathrm{p}} & \text{if } H < c_{\mathrm{p}}T_{\mathrm{m}} \\ T_{\mathrm{m}} & \text{if } c_{\mathrm{p}}T_{\mathrm{m}} \leqslant H \leqslant c_{\mathrm{p}}T_{\mathrm{m}} + L \\ (H - L)/c_{\mathrm{p}} & \text{if } H > c_{\mathrm{p}}T_{\mathrm{m}} + L. \end{cases}$$

The material velocity is damped in the vicinity of the solidification front using a linear relationship to $\epsilon_{\mathrm{s}}$ scaled to resolve the time scale of the moving phase front, $A(H) = 1.56 \times 10^5 \epsilon_{\mathrm{s}}$.

To be consistent with earlier work, first-order fully implicit (Backward-Euler) time discretization and centered second-order spatial differencing is used. The domain is discretized spatially using a finite volume Cartesian staggered fixed grid in a square cavity of equally sized cells. As boundary conditions, insulating top and bottom walls and constant values of enthalpy on the left and right sidewalls are prescribed.

The JFNK algorithm is a nested iteration solution method ([4] provides a comprehensive explanation) that uses inexact Newton's methods to reduce the nonlinear residuals of (1)–(3), $\mathbf{F}(\mathbf{x})$, to a specified nonlinear tolerance ($\eta_{nl} = 1 \times 10^{-5}$). Expanding $\mathbf{F}(\mathbf{x})$ using a Taylor series to first order gives

$$\mathbf{J}(\mathbf{x}^k)\delta\mathbf{x}^k = -\mathbf{F}(\mathbf{x}^k), \quad \mathbf{x}^{k+1} = \mathbf{x}^k + \delta\mathbf{x}^k, \tag{6}$$

where $\mathbf{J}$ is the Jacobian of $\mathbf{F}$ and $k$ is the outer nonlinear iteration index.

GMRES, a solution method for nonsymmetric matrices, is used to generate $\delta\mathbf{x}^k$. With JFNK, a finite difference approximation of $\mathbf{J}$ times a vector is used to build the next element of the Krylov subspace

$$\mathbf{J}\mathbf{v} \simeq \frac{\mathbf{F}(\mathbf{x} + \epsilon\mathbf{v}) - \mathbf{F}(\mathbf{x})}{\epsilon} \tag{7}$$

using a small perturbation, $\epsilon$. This is performed until the update from a linear combination of the Krylov subspace elements satisfy a linear tolerance

$$\frac{\|\mathbf{J}\delta\mathbf{x}^k + \mathbf{F}(\mathbf{x}^k)\|_2}{\|\mathbf{F}(\mathbf{x}^k)\|_2} < \eta_k = 1 \times 10^{-2}. \tag{8}$$

Although using Eq. (7) avoids the cost of forming $\mathbf{J}$ to satisfy (8), significantly nonlinear problems such as phase change convection may require many GMRES iterations. Because each additional term of the Krylov vector is added to a linear combination of previous terms, its storage costs can grow quickly. Thus, it is crucial to mitigate storage requirements with an effective preconditioner.

For this discussion, the application of the preconditioner is written in symbolic matrix form, $\widetilde{\mathbf{M}}^{-1}$. To avoid operations on the residual norms, which can affect convergence, the right preconditioning option is used, and Eq. (6) becomes

$$\mathbf{J}\widetilde{\mathbf{M}}^{-1}\delta\mathbf{z} = -\mathbf{F}(\mathbf{x}), \tag{9}$$

where $\delta\mathbf{z} = \widetilde{\mathbf{M}}\delta\mathbf{x}$. Eq. (9) encapsulates a multi-step process. First, evaluate $\mathbf{J}(\widetilde{\mathbf{M}}^{-1}\delta\mathbf{z})$ with Eq. (7), until the linear tolerance is satisfied. Then evaluate $\delta\mathbf{x}$ using $\delta\mathbf{x} = \widetilde{\mathbf{M}}^{-1}\delta\mathbf{z}$.

Following [7], the application of the MG-SIMPLE preconditioner is a basic V-cycle multigrid method using a discretization coarse grid approximation [11]. Multigrid builds a solution to the original problem on a series of coarser grids using linear stationary methods, or 'smoothers'. These smoothers preferentially remove higher frequency error [8], including low frequency error mapped from fine to coarse grids. The smoother used here is the SIMPLE algorithm, which removes higher frequency error while also providing limited physical coupling. The grid dimensions are coarsened by a factor of two and the values are restricted using a weighted average. The same number of smoothing applications are performed after each coarsening. For this analysis, the matrix coefficients required by SIMPLE are rediscretized on each coarsened grid. The solution correction is prolongated back to finer grids using a bilinear interpolation formula and postsmoothing sweeps are applied on that grid. A specified number of sweeps through the SIMPLE smoother are applied as presmoothing, postsmoothing, and coarse grid sweeps to build $\widetilde{\mathbf{M}}^{-1}\delta\mathbf{z}$ as outlined below.

For phase change convection, the specifics of SIMPLE applied as a smoother to MG match the application of SIMPLE as a preconditioner in Evans et al. [3]. SIMPLE is a segregated solver, whereby each dependent variable defined by (1)–(3) is solved individually with frozen coefficients in the operators (4) and (5). As a further simplification within the preconditioner, no phase change physics is implemented in SIMPLE and first-order upwind spatial discretization is applied. Twenty SOR iterations and 10, 10, and 5 Gauss-Seidel iterations through variables $H$, $u$, $v$, and $p$, respectively, produce efficient results within the SIMPLE preconditioner and smoother.

To assess the performance of the JFNK–MG-SIMPLE algorithm, it is applied to both time dependent (a) natural convection and (b) phase change convection as defined by Eqs. (1)–(3). Problem (a) was selected because MG has been demonstrably successful as a preconditioner to solutions of time dependent natural convection problems [7] and this establishes a baseline. Problem (b) is a benchmark for two-phase flow and entry to more complex and realistic related problems. These analyses were performed within a square domain and cells initially set to $T = +0.5$. The left and right side walls are $T = -0.5$ and $T = +0.5$, respectively, at all times. For the natural convection problem there is no freezing, and for the phase change problem, $T_{\mathrm{m}} = 0$. The simulations are run to time = 200 and are displayed in Fig. 1. The thick black line denotes the $T = 0$ contour, which for case (a) is just a reference. Buoyancy induced convection is counterclockwise and has advected cooler material into the lower portion of the domain for both problems. For (b), phase change retards convection within the frozen region and near the boundary of the phase front. The model that produced the numerical solutions presented below was verified in [3] to be first order accurate in time using an $L_2$ norm of error compared to a base solution with small time step.

The time dependent natural convection problem is the same as the phase change convection problem given by Eqs. (1)–(3), but with no latent heat release or associated velocity attenuation. Table 1 displays the CPU time and linear to nonlinear iteration ratio for the first time step ($\Delta t = 1$) on a $128^2$ grid using the SIMPLE and MG-SIMPLE preconditioners. The number of grid coarsenings is varied for MG-SIMPLE runs to illustrate the reduction in CPU time and number of linear iterations performed as additional coarser levels are used to build a preconditioner update. On a $128^2$ grid, the reduction in CPU time with the multigrid preconditioner is almost an order of magnitude, with most of the gain due to the first coarsening.

Solutions with the MG-SIMPLE (with 4 coarsenings) and SIMPLE preconditioners are compared for a range of grid sizes, with simulations run to time = 200 (Fig. 1a). The average ratio of linear to nonlinear iterations as a function of grid size is displayed in Fig. 2a, and shows relatively weak growth with increasing grid size for MG-SIMPLE versus SIMPLE preconditioners. We observe that the associated CPU reductions at time = 1 (Table 1) are greater than when the simulation is run to time = 200. This may be due to the fact that the problem is approaching steady state and the effort to achieve a converged solution for both methods is reduced. Nonetheless, the time = 200 simulation achieves a CPU reduction of 60% on a $128^2$ grid.

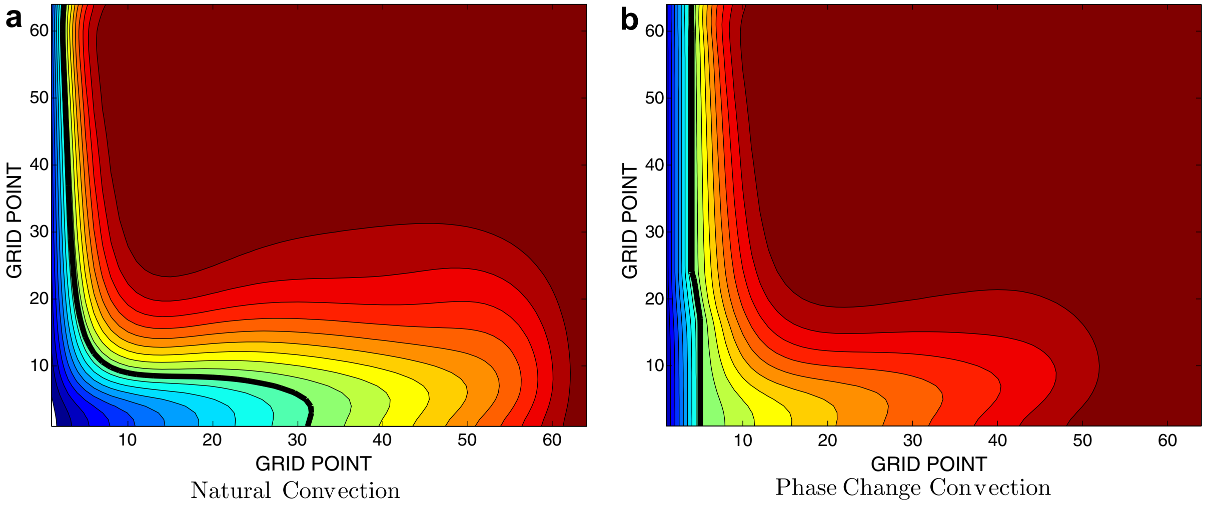

Fig. 1. Solutions to time dependent natural convection and phase change convection problems after time = 200 for a $64^2$ grid. The contours are the temperature ±0.05 from the thick black line, which denotes $T = 0$ for both plots and is the freezing point for (b). (a) Natural convection; (b) phase change convection.

Table 1
Performance statistics for SIMPLE and MG-SIMPLE preconditioners applied to a time dependent natural convection problem on a $128^2$ grid for a single time step of $\Delta t = 1$

| Preconditioner | # Coarsenings | Ratio | CPU |
|---|---|---|---|
| SIMPLE | 0 | 71.25 | 9.95 |
| MG-SIMPLE | 1 | 23.0 | 1.45 |
| MG-SIMPLE | 2 | 16.0 | 1.22 |
| MG-SIMPLE | 3 | 15.25 | 1.12 |
| MG-SIMPLE | 4 | 15.0 | ≡1 |

For the SIMPLE preconditioner, 2 sweeps are performed, and for MG-SIMPLE, 2 presmoothing, 1 postsmoothing, and 2 SIMPLE sweeps at coarsest level are performed. ‘Ratio’ refers to the number of linear per nonlinear iterations, and the CPU is normalized to the fastest run.

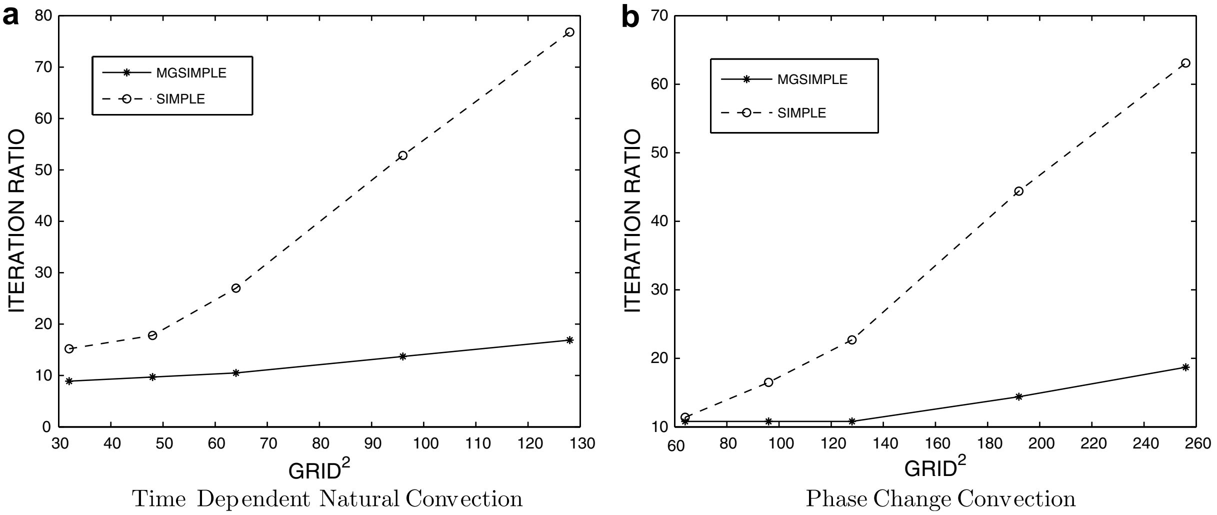

Fig. 2. Ratio of average linear to nonlinear iterations as a function of grid$^2$ size for (a) time dependent natural convection and (b) phase change convection. The time step size is halved with grid size doubling to keep the dynamic and phase front scaling consistent.

Table 2
Time dependent convection with solidification on a $256^2$ grid and time step of 0.25 at time = 200

| Preconditioner | # Coarsenings | Ratio | CPU |
|---|---|---|---|
| SIMPLE | 0 | 63.1 | 2.7 |
| MG-SIMPLE | 1 | 27.4 | 1.36 |
| MG-SIMPLE | 2 | 19.6 | 1.06 |
| MG-SIMPLE | 3 | 18.7 | ≡1 |

CPU time is normalized to the fastest run. CPU and the ratio of linear to nonlinear iterations are displayed for simulations using the SIMPLE preconditioner (4 sweeps) and the MG-SIMPLE preconditioner with 2 pre- and postsmoothing SIMPLE sweeps each and 4 sweeps at the coarsest grid.

Next, all the components of the phase change problem outlined in Eqs. (1)–(3) are included and run to time = 200 (Fig. 1b). Similar to Table 1 for natural convection, Table 2 displays normalized CPU and iteration count for phase change convection on a $256^2$ grid. With MG-SIMPLE versus SIMPLE as the preconditioner (3 grid coarsenings), the simulation is completed 63% faster. Coarsening beyond three levels did not produce additional reductions in the CPU time. Possibly, the moving phase front is not well resolved on the coarsest mesh for this MG preconditioner algorithm. As with the natural convection problem, the phase change convection simulation is run for a range of grid sizes to assess the performance of the MG-SIMPLE preconditioner with increasingly finer grids. Fig. 2b, analogous to Fig. 2a for natural convection, shows weaker growth of the linear to nonlinear iteration ratio with increasing grid size. As a result, relative CPU savings by using MG-SIMPLE versus SIMPLE preconditioners grows from 30% on a $128^2$ grid to 63% on a $256^2$ grid.

We are not aware of any application of multigrid methods to time dependent phase change problems. Since the partial differential equation systems representing these problems can have discontinuities and exhibit hyperbolic behavior they may render multigrid methods ineffective. However, by incorporating a basic multigrid method within a SIMPLE preconditioned JFNK algorithm for these equations, the number of function evaluations to build an approximate update and the associated CPU time are reduced. This effect becomes more pronounced on finer grids. Although parameters such as pre-, post-, and coarse grid sweeps were chosen to minimize the CPU time using MG-SIMPLE, further adjustment could improve results. Also, the multigrid method implemented here is rather basic. Using more sophisticated coarse grid matrices, restriction and/or prolongation operators designed for the more complex equations could yield more dramatic results. For example, a Galerkin method [2] may better translate phase change information from the finer grid.

## Acknowledgement

This work was performed under US government contract DE-AC52-06NA25396 for Los Alamos National Laboratory, which is operated by Los Alamos National Security, LLC (LANS) for the US Department of Energy (LA-UR-06-2600).

## Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.jcp.2006.09.003.

## References

[1] J. Danzig, Modelling liquid–solid phase changes with melt convection, Int. J. Numer. Meth. Eng. 28 (1989) 1769–1785.
[2] J. Dendy, Black box multigrid, J. Comput. Phys. 48 (1982) 366.
[3] K. Evans, D.A. Knoll, M.A. Pernice, Development of a 2-D algorithm to simulate convection and phase transition efficiently, J. Comput. Phys. 219 (2006) 404–417.
[4] D. Knoll, D. Keyes, Jacobian-free Newton–Krylov methods: a survey of approaches and applications, J. Comput. Phys. 193 (2004) 357–397.
[5] D.A. Knoll, W.J. Rider, A multigrid preconditioned Newton–Krylov method, SIAM J. Sci. Comput. 21 (1999) 691–710.

[6] S.V. Patankar, Numerical Heat Transfer and Fluid Flow, Hemisphere Publishing Corp., New York, 1980.

[7] M. Pernice, M. Tocci, A multigrid-preconditioned Newton–Krylov method for the incompressible Navier–Stokes equations, SIAM J. Sci. Comput. 23 (2001) 398–418.

[8] S. Sivaloganathan, G.J. Shaw, A multigrid method for recirculating flows, Int. J. Numer. Meth. Fluids 8 (1988) 417–440.

[9] Syamsudhuha, D. Silvester, Efficient solution of the steady-state Navier–Stokes equations using a multigrid preconditioned Newton–Krylov method, Int. J. Numer. Meth. Fluids 43 (2003) 1407–1427.

[10] V. Voller, M. Cross, N.C. Markatos, An enthalpy method for convection/diffusion phase change, Int. J. Numer. Meth. Eng. 24 (1987) 271–284.

[11] P. Wesseling, An Introduction to Multigrid Methods, John Wiley & Sons, Chichester, 1992.