Enabling Fluid-Structural Strong Thermal Coupling Within a Multi-Physics Environment

Russell W. Hooper, Thomas M. Smith and Curtis C. Ober

Sandia National Laboratories, MS 0316, P.O. Box 5800 Albuquerque, NM 87185-0316*

We demonstrate use of a Jacobian-Free Newton-Krylov solver to enable strong thermal coupling at the interface between a solid body and an external compressible fluid. Our method requires only information typically used in loose coupling based on successive substitution and is implemented within a multi-physics framework. We present results for two external flows over thermally conducting solid bodies obtained using both loose and strong coupling strategies. Performance of the two strategies is compared to elucidate both advantages and caveats associated with strong coupling.

I. Introduction

Over the past two decades several single and specialized physics codes have been developed at Sandia National Laboratories. During this time, computational capability has increased enormously as measured by gains in CPU speed and memory capacity. This has led to a desire to perform more realistic physical simulations comprising richer physics solved over finer discretizations. Enabling richer physics posed one of two possibilities: either expand existing mature physics codes to incorporate new physics models or bring existing physics codes together into an environment such that they can each provide a simulation component of a larger multi-physics simulation. Many arguments can be made for the latter choice while respecting the fact that the former choice has merit for special situations and for optimizing solution techniques for a fixed set of multi-physics.

Recognizing the need to enable multi-physics simulations in a general and extensible way, Sandia has recently worked to develop a common environment, i.e. framework, to leverage basic services such as I/O, message-passing, element libraries, etc. Application codes brought into this environment can then send and receive data from other applications, thereby enabling multi-physics simulations without requiring significant re-writing of code. Independent of an application framework, Sandia has also worked to develop an extensible solver framework, Trilinos.[1] Overarching these two independent efforts is a desire to unite actively developed application codes with actively developed solution software. The work presented in this paper represents an example of the realization of this goal.

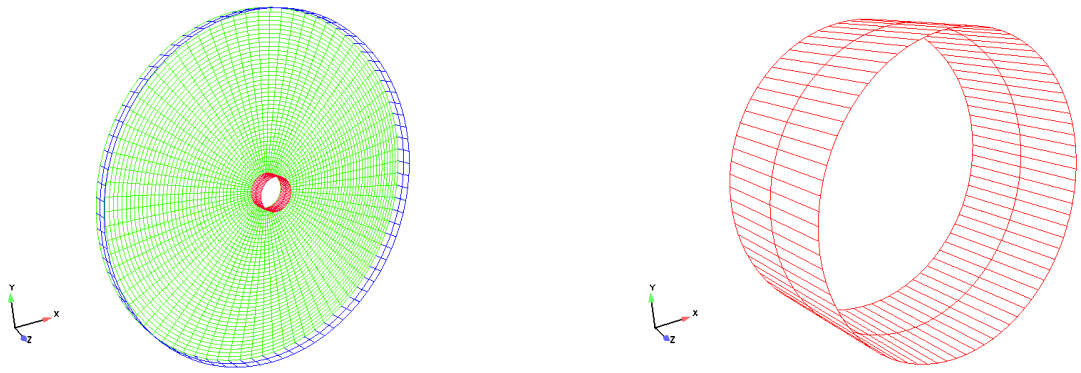
This paper addresses a particular multi-physics scenario comprising compressible external fluid flow over a thermally conducting body. Fluid flow is simulated using the compressible flow code, Premo, while thermal transport is solved using a separate code, Calore, both of which reside within the multi-physics framework, Sierra.[2] A common and straightforward coupling strategy consisting of successive substitution of results between the codes is considered first. We refer to this as loose coupling. Next, we describe how a stronger Newton-based coupling technique can be achieved using the same data employed to perform loose coupling.

Our paper is structured as follows. First, we describe the physical system to be modeled. This is followed by a brief overview of the simulation codes Premo and Calore which provide solutions to compressible fluid flow and thermal transport, respectively. The next section provides an overview of both loose coupling via successive substitution and stronger Newton-based coupling performed using extensions to a nonlinear solver package NOX[3] which is part of Trilinos. Results obtained using the two coupling strategies are then presented and performance compared. Finally, we provide our conclusions and ideas for future work.

*Sandia is a multiprogram laboratory operated by Sandia Corporation, A Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000. This material is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

II. Physical Problem

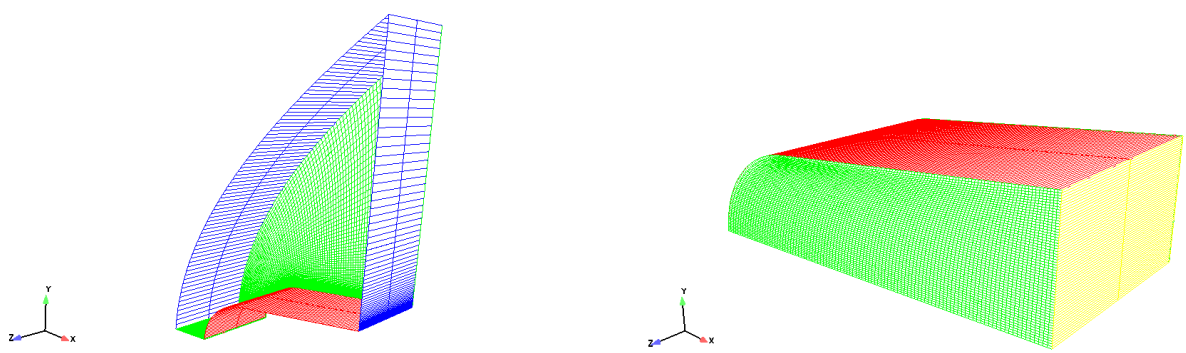
As shown in figures 1 and 2 we consider external compressible flow over a solid body whose temperature is governed by linear heat conduction.



(a) Fluid domain and mesh having outer radius of 10.0.

(b) Solid domain surface mesh having radial dimension of 1.0.

Figure 1. Outer fluid and inner solid domains for external compressible flow over a thermally conducting circular cylinder. A two-element planar slab of thickness 1.0 **is used.**



(a) Blunt-wedge fluid geometry and mesh configured to capture a shock at $Ma = 3.0$.

(b) Blunt-wedge conducting solid geometry.

Figure 2. Blunt-wedge geometry.

For both geometries, fluid flow external to the solid body is modeled as viscous and compressible via the Navier-Stokes equations. These can be expressed in the following form:

$$\int_{\mathcal{V}} \frac{\partial \mathbf{W}}{\partial t}\, d\mathcal{V} + \oint_{\Gamma} \left(\mathbf{F}^c + \mathbf{F}^v\right) \cdot \mathbf{n}\, d\Gamma = 0 \tag{1}$$

where $\mathbf{W} = \{\rho,\, \rho u,\, \rho v,\, \rho w,\, \rho E\}^T$ is the solution vector of conserved variables and

$$\mathbf{F}^c \cdot \mathbf{n} = \left\{ \begin{array}{c} (\rho u_i) n_i \\ (\rho u_1 u_i + p\delta_{i1}) n_i \\ (\rho u_2 u_i + p\delta_{i2}) n_i \\ (\rho u_3 u_i + p\delta_{i3}) n_i \\ ((\rho E + p)\, u_i) n_i \end{array} \right\}$$

is the inviscid flux vector, and

$$\mathbf{F}^v \cdot \mathbf{n} = \left\{ \begin{array}{c} 0 \\ \tau_{i1} n_i \\ \tau_{i2} n_i \\ \tau_{i3} n_i \\ (-q_i + u_j \tau_{ij}) n_i \end{array} \right\}$$

is the viscous flux vector. In the above, ρ is the density; $u_i = \mathbf{u}$ with $u = u_1$, $v = u_2$, and $w = u_3$ are the Cartesian velocity components; p is the pressure; E is the total energy defined by $E = e + \mathbf{u} \cdot \mathbf{u}/2$; e is the specific internal energy defined by $e = C_v T$; C_v is the specific heat at constant volume; T is the temperature defined for an ideal gas by $p = \rho R T$; R is the gas constant, $n_i = \mathbf{n}$ is the surface normal vector; and δ_{ij} is the Kronecker delta. Corresponding to the conservative state vector $\mathbf{W}$ we define a primitive state vector $\mathbf{U} = \{\rho, u, v, w, p\}^T$.

The viscous stress tensor is given by,

$$\tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3} \mu \frac{\partial u_k}{\partial x_k} \delta_{ij} \tag{2}$$

where μ is the molecular viscosity coefficient. Heat fluxes $\mathbf{q}$ are assumed to obey Fourier's Law,

$$\mathbf{q} = -\kappa \nabla T \ . \tag{3}$$

and a constant Prandtl number, Pr is used.

The solid body is shown as the red surfaces in figures 1 and 2 and is treated as impermeable to fluid flow and having no-slip. The former condition implies

$$\mathbf{F}^b = \mathbf{F} \cdot \mathbf{n}^b = \left\{ \begin{array}{c} 0 \\ p\mathbf{n}^b \\ 0 \\ 0 \end{array} \right\} \ ,$$

and the latter condition implies $\mathbf{u} = 0$. Superscript b denotes enforcement at the fluid-body interface. Far-field boundary conditions are depicted as blue surfaces in the figures and are specified via an assumed infinity profile, $\mathbf{u} = \mathbf{U}_\infty$. The green surfaces correspond to symmetry planes involving no flow across and no traction along them.

HEAT transfer within the solid body is modeled as a balance of thermal transport governed by the following equation,

$$\rho C \frac{\partial T}{\partial t} - \nabla \cdot (\mathbf{K} \nabla T) - \dot{q} = 0 \tag{4}$$

where T is the unknown temperature field within the body, $\mathbf{K}$ is a thermal conductivity matrix, and $\dot{q}$ is a volumetric source term. For the present work, $\mathbf{K}$ is treated as a scalar constant, e.g. $\mathbf{K} = \kappa$, and the source term is assumed to be zero. We also limit this work to steady-state conditions. Together, these assumptions reduce eq.(4) to the Laplace equation governing linear heat conduction within the solid body.

Thermal boundary conditions at the fluid-body interface (red surfaces) represent the coupling between the fluid and solid domains. Conjugate heat transfer is modeled by requiring temperature and heat flux to be continuous across the interface:

$$T_f^b = T_s^b \tag{5}$$

$$\mathbf{q}_f^b = \mathbf{q}_s^b \ , \tag{6}$$

where subscript f denotes the fluid phase, and subscript s denotes the solid body. Equation (5) represents a Dirichlet boundary condition, while eq.(6) is of Neumann type.

III. Physics Codes

THIS section briefly describes the two codes Premo and Calore which solve the equations governing compressible viscous fluid flow and heat transfer, respectively. Both codes utilize the Sierra framework[2]

for general support of common capabilities such as parsing user input, reading mesh data, preprocessing, post-processing results, interfacing to linear and nonlinear solvers, managing inter-mesh transfers, message-passing for distributed parallel execution, dynamic load balancing, and adaptivity. As such Sierra provides a massively parallel computing environment enabling multi-physics simulations.

Premo is based on a node-centered, edge-based, finite-volume algorithm similar to the formulation in Haselbacher and Blazek[4] and Luo *et al.*[5] More detail can be found in ref.[6,7] Calore is a computational heat transfer program based on galerkin finite elements that solves a wide variety of both linear and nonlinear thermal physics models.

The two codes are coupled within the Sierra framework through the boundary conditions of eqs.(5) and (6). The current state of each code requires eq.(5) to be enforced within Premo as a Dirichlet condition with values transferred from Calore. Conversely, eq.(6) is enforced by first having Premo compute a heat flux normal to the solid body at each surface node on the fluid/solid interface, then compute a corresponding set of convective heat transfer coefficients, and finally transfer these coefficients to Calore for use in its surface integration along the interface within the context of enforcing Neumann-type boundary conditions using Galerkin finite elements. Hence, continuity of heat flux across the interface is enforced via the expression:

$$-\kappa \mathbf{n} \cdot \nabla T = h\left(T_s^b - T_{ref}\right) \tag{7}$$

where Premo computes the left expression, and then determines values for h using an arbitrary but realistic value for T_{ref}.

IV. Coupling Strategies

THE two methods we wish to consider to effect multi-physics code coupling are presented in this section. Both are intended to find a solution to the following system that is nonlinear in general,

$$\mathbf{R}(\mathbf{W}, T) = (\mathbf{R_W}, \mathbf{R}_T) = \mathbf{0} \tag{8}$$

with $\mathbf{R_W}$ defined by eq.(1) and $\mathbf{R}_T$ by eq.(4). The values $\mathbf{W}$ represent the vector degrees of freedom for Premo (5 per node in the discretized fluid domain), and T is the dependent scalar field within the solid body and represent the unknowns for Calore.

A. Loose Coupling

Loose coupling represents the traditional and straightforward approach of coupling a fluids code to a thermal code across a fluid-solid interface. Several studies have been performed that elucidate issues that must be addressed to perform this coupling properly.[8,9] From a solver perspective, loose coupling is performed via successive substitution of results obtained from each code. For our work, we perform loose coupling using successive substitution of solutions for each physics application (Premo or Calore) each obtained by applying a Newton method. For the case of beginning the solution process by solving flow (Premo) first, loose coupling is represented by the following iteration cycle (i.e. W-T cycle):

$$\text{Do while} \quad \|\mathbf{R}(\mathbf{W}, T)\| > \epsilon \quad :$$

$$\text{Solve Premo for } \mathbf{W}: \quad \mathbf{R_W}(\mathbf{W}, \tilde{T}) = \mathbf{0} \tag{9}$$

$$\Downarrow \quad \tilde{\mathbf{W}} \leftarrow \mathbf{W} \tag{10}$$

$$\text{Solve Calore for } T: \quad \mathbf{R}_T(\tilde{\mathbf{W}}, T) = \mathbf{0} \tag{11}$$

$$\Downarrow \quad \tilde{T} \leftarrow T \tag{12}$$

$$\text{Evaluate}: \quad \|\mathbf{R}(\mathbf{W}, T)\| \tag{13}$$

where over-tildes indicate values obtained from another application and treated as fixed data during the solution step shown, and eqs.(10) and (12) represent a transfer of variables to a form needed by the subsequent physics application. The transfers are handled by the Sierra framework. The norm calculation of eq.(13) requires re-computation of all physics application residuals using updated variable values. The entire loose coupling cycle is repeated until some measure of overall convergence is attained, eg eq.(13). Solutions to eqs.(9) and (11) are obtained using Newton's method for each problem, eg for Calore's nonlinear iteration k,

$$\mathbf{J}_{TT}^k \Delta T^k = -\mathbf{R}_T^k \tag{14}$$

$$T^{k+1} = T^k + \Delta T \tag{15}$$

Iterations are continued until the norm $\|\mathbf{R}_T\| < \epsilon_T$. For our study, the linear problems of eq.(14) were solved iteratively using a Generalized Minimum Residual iterative method. Because one primary objective of our coupling strategies involves enabling coupled multi-physics while minimizing additional requirements on any existing application, we evaluate the Jacobians $\mathbf{J}_{TT}$ and $\mathbf{J}_{\mathbf{WW}}$ numerically using finite-differencing with coloring of the matrix graphs. This feature is available in the NOX nonlinear solver library.

B. Strong Coupling

We consider strong coupling to encompass methods in which updates to variables from each physics code are performed together with inter-problem dependencies included in computing an update. Compared to loose coupling, this approach represents a first-order Newton-based strategy. The Newton-based method requires only the same problem information as is used in performing loose coupling.

Our strong coupling approach is based on performing nonlinear iterations for the coupled system using a Newton-based approach that imposes no additional requirements on either physics application than what is needed for the loose coupling described above. This is achieved by using a Jacobian-free Newton-Krylov[10] implementation to effect strong coupling between the two physics codes. The iterative linear solve performed at each nonlinear iteration requires only the action of the Jacobian on a vector. Within the context of Newton's method, the action of the Jacobian on a vector $\mathbf{p}$ can be obtained without explicitly forming the matrix by using a directional derivative, ie

$$\mathbf{J}\mathbf{p} \approx \frac{\mathbf{R}(\mathbf{x}+\epsilon\mathbf{p}) - \mathbf{R}(\mathbf{x})}{\epsilon} \tag{16}$$

In practice, some form of preconditioning is required and leads to a preconditioned form of the directional derivative,

$$\mathbf{J}\mathbf{M}^{-1}\mathbf{p} \approx \frac{\mathbf{R}(\mathbf{x}+\epsilon\mathbf{M}^{-1}\mathbf{p}) - \mathbf{R}(\mathbf{x})}{\epsilon} \ . \tag{17}$$

with $\mathbf{M}$ being a preconditioning matrix. For our present work, we construct $\mathbf{M}$ as a block diagonal matrix consisting of the Jacobians used by each problem in performing loose coupling, i.e.

$$\mathbf{M} = \begin{bmatrix} \mathbf{J}_{\mathbf{WW}} & \\ & \mathbf{J}_{TT} \end{bmatrix} \tag{18}$$

In keeping with our priority of minimizing requirements for physics codes to participate in a coupled muti-physics simulation, we fill the blocks of the preconditioning matrix and perform the directional derivatives all numerically using only residual evaluations. Knowledge of the structure of the Jacobian matrices is also used with the coloring algorithm for achieving efficient numerical evaluation of the Jacobian blocks via finite-differences.

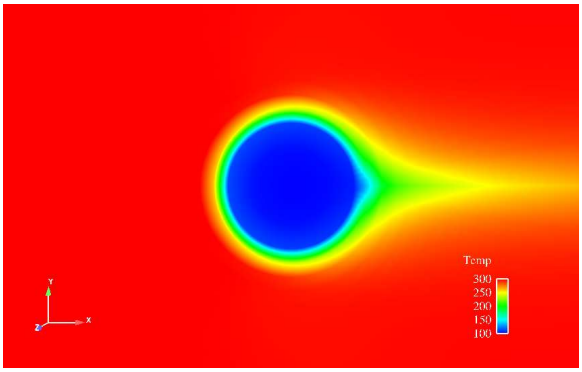
V. Results

In this section, we provide results for fluid-structural thermal coupling computed using both lose and strong coupling strategies for external flows over both the circular cylinder and blunt-wedge bodies.

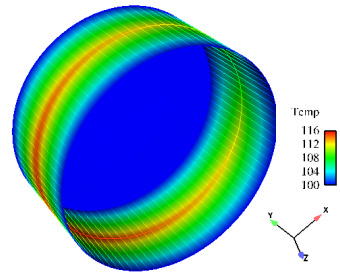
A. Circular Cylinder

The fluid and solid material properties for the circular cylinder geometry are chosen to reflect a somewhat facile problem in order to ensure converged solutions and thereby allow comparisons between the coupling solution strategies to be made. Accordingly, the flow is characterized by $Re = 374$, $Ma = 0.3$ and $Pr = 120$. The far-field flow is represented by $\mathbf{U}_\infty = \{\rho, u, v, w, p\} = \{$ 1.18 kg/m^3, 104 m/s, 0 m/s, 0 m/s, 101325 Pa $\}$. Assuming ideal gas gives $T_\infty = 300K$. The solid circular cylinder has thermal conductivity $\kappa = 401$. A Dirichlet temperature boundary condition $T = T_w$ is imposed on the faces of the circular cylinder at $z = \pm 0.5$. Three cases are considered: $T_w = 100, 300$ and $500K$. Results for the temperature profile on the cylinder surface as well as in the fluid in the vicinity of the cylinder are shown in figures 3 - 5, respectively.

Noteworthy in figures 3 - 5 are that the temperatures on the fluid/solid interface from the fluid and the solid temperature solutions agree to within the convergence tolerance as they should according to eq.(5)

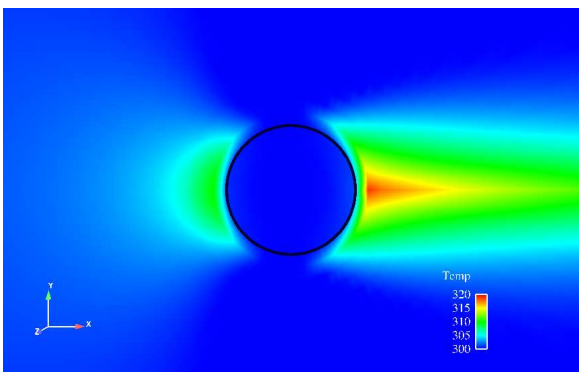


(a) Fluid temperature along plane $z = 0$

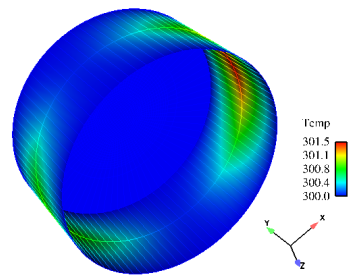


(b) Surface temperature profile.

Figure 3. Temperature profiles for case with $T_{z=\pm 0.5} = 100.0$.

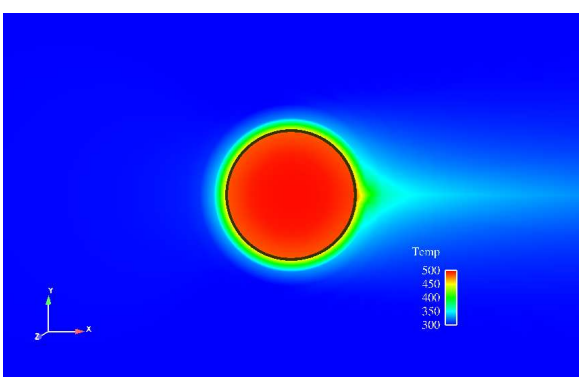


(a) Fluid temperature along plane $z = 0$

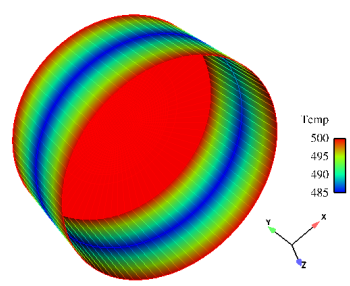


(b) Surface temperature profile.

Figure 4. Temperature profiles for case with $T_{z=\pm 0.5} = 300.0$.



(a) Fluid temperature along plane $z = 0$



(b) Surface temperature profile.

Figure 5. Temperature profiles for case with $T_{z=\pm 0.5} = 500.0$.

and that the fluid solutions **W** differ markedly from those obtained using Premo with either isothermal or adiabatic conditions applied to the cylinder surface.

Comparison of the solution convergence behavior using loose and strong coupling strategies are shown in figure 6 and are summarized with total required CPU times in table 1. Included in the figure and table are results for loose coupling performed by starting the successive substitution using either Premo (W-T cycle) or Calore (T-W cycle). For loose coupling the number of nonlinear iterations corresponds to cumulative Premo nonlinear iterations. This choice was made because computational cost is dominated by the Premo problem compared to the Calore problem which comprises only 4443 degrees of freedom out of a total of 53,643 and is linear. Clearly, the performance of loose coupling depends critically on the order of solves within each coupling iteration. Strong coupling consistently performs slightly better than loose coupling (T-W cycle) in terms of total time required for convergence.

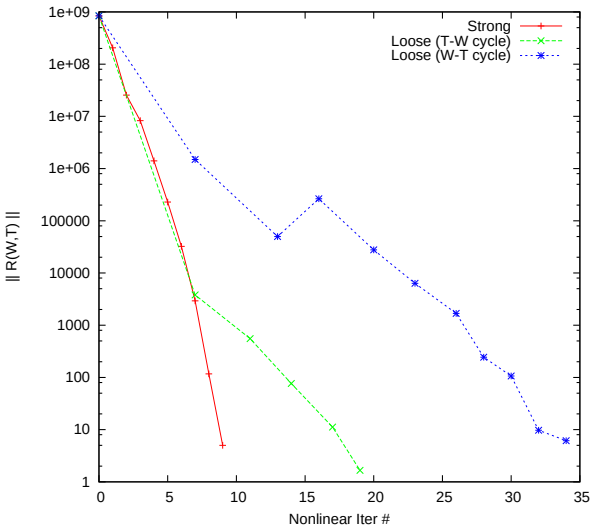
Table 1. Comparison of loose coupling and strong coupling for the circular cylinder geometry.

Coupling Method	Coupling Iters*	Total Time (CPU s)
$T_{z=\pm 0.5} = 100.0K$		
Loose (W-T cycle)	10	756
Loose (T-W cycle)	5	429
Strong	9	421
$T_{z=\pm 0.5} = 300.0K$		
Loose (W-T cycle)	8	543
Loose (T-W cycle)	3	276
Strong	7	267
$T_{z=\pm 0.5} = 500.0K$		
Loose (W-T cycle)	10	750
Loose (T-W cycle)	3	317
Strong	8	299

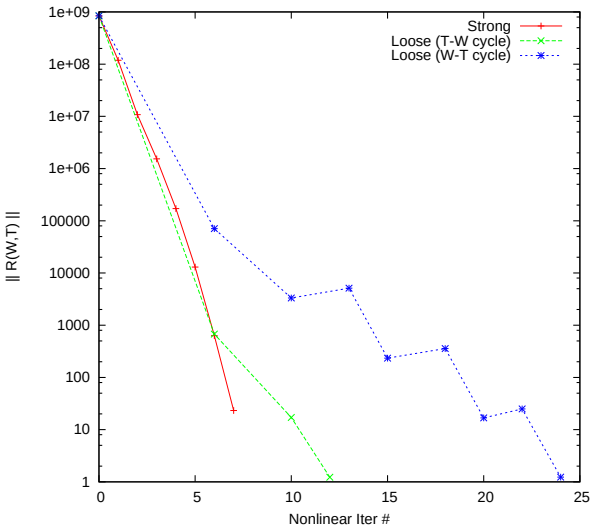
* Total number of coupling iterations. For loose coupling, this is the number of cycles of successive substitution and differs from the number of Premo nonlinear iterations reported in figure 6.

B. Blunt-wedge

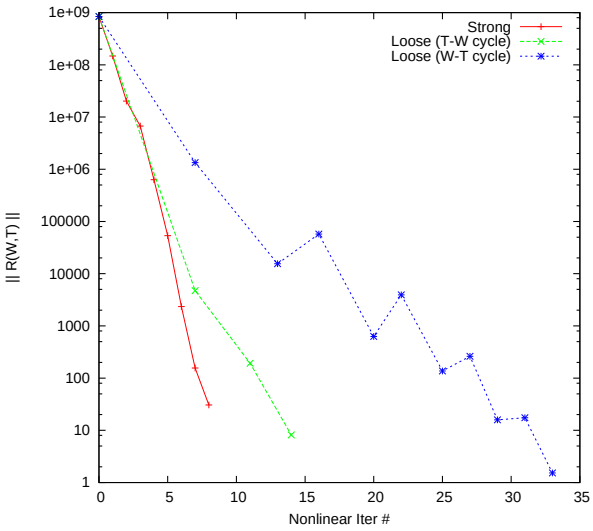
The blunt-wedge geometry was tailored and meshed to capture a $Ma = 3.0$ shock. This problem was made considerably more difficult than the circular cylinder problem by specifying $Re = 2.4x10^5$ and $Pr = 0.72$. Rather than beginning with uniform initial conditions, we ran Premo with an isothermal temperature of $300K$ on the blunt-wedge solid body to obtain a starting solution with the shock qualitatively formed. Coupling to the solid body was then effected by setting the back face of the body to $400K$ and all other surfaces except the fluid/solid interface to have no penetration and slip. The temperature field in the fluid and solid is shown in figure 7, and convergence behavior is compared in figure 8. For this problem, the quality of convergence is dramatically different with loose coupling (T-W cycle) behaving about as well as for the easier circular cylinder problem but with strong coupling exhibiting very poor behavior. Besides more challenging physics, this problem also manifests a non-overlapping surface mesh at the interface, having a solid surface interface meshed at twice the refinement level as the fluid mesh on the interface. This mismatch makes the transfer of variables non-trivial, and the action of the Jacobian performed using directional derivatives now captures the sensitivity of equations near the interface to the details of how the variable transfers are done. Evidence that this may be a source of pollution is observed in the sharp upturns in convergence behavior occurring at nonlinear iterations for which the iterative linear solver failed to achieve the requested linear solve tolerance. We are currently quantifying this potential source of error.



(a) Convergence comparison for case with $T_{z=\pm 0.5} = 100.0K$.

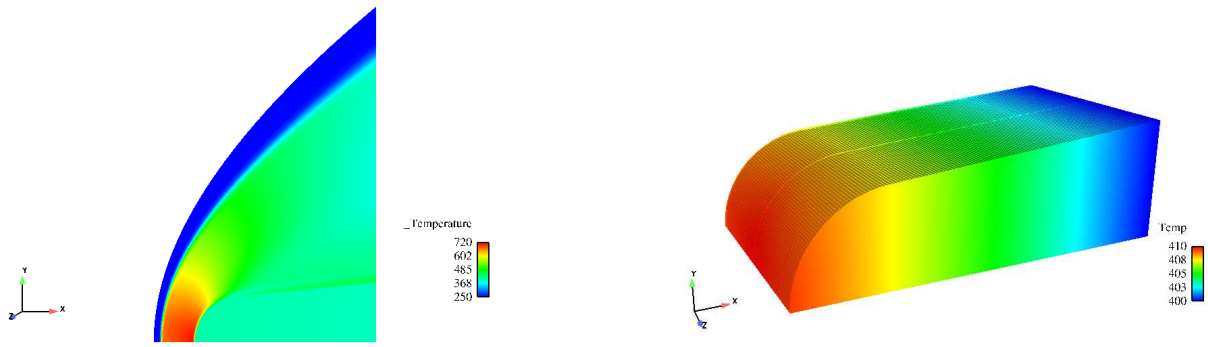


(b) Convergence comparison for case with $T_{z=\pm 0.5} = 300.0K$.



(c) Convergence comparison for case with $T_{z=\pm 0.5} = 500.0K$.

Figure 6. Comparison of convergence behavior for loose and strong coupling methods represented as coupled residual norm $||\mathbf{R}(\mathbf{W}, T)||$ vs number of nonlinear iterations. Note that the number of nonlinear iterations for loose coupling corresponds to those for the sequence of Premo problems.



(a) Fluid temperature profile.

(b) Surface temperature profile.

Figure 7. Temperature profiles for Blunt-wedge problem.

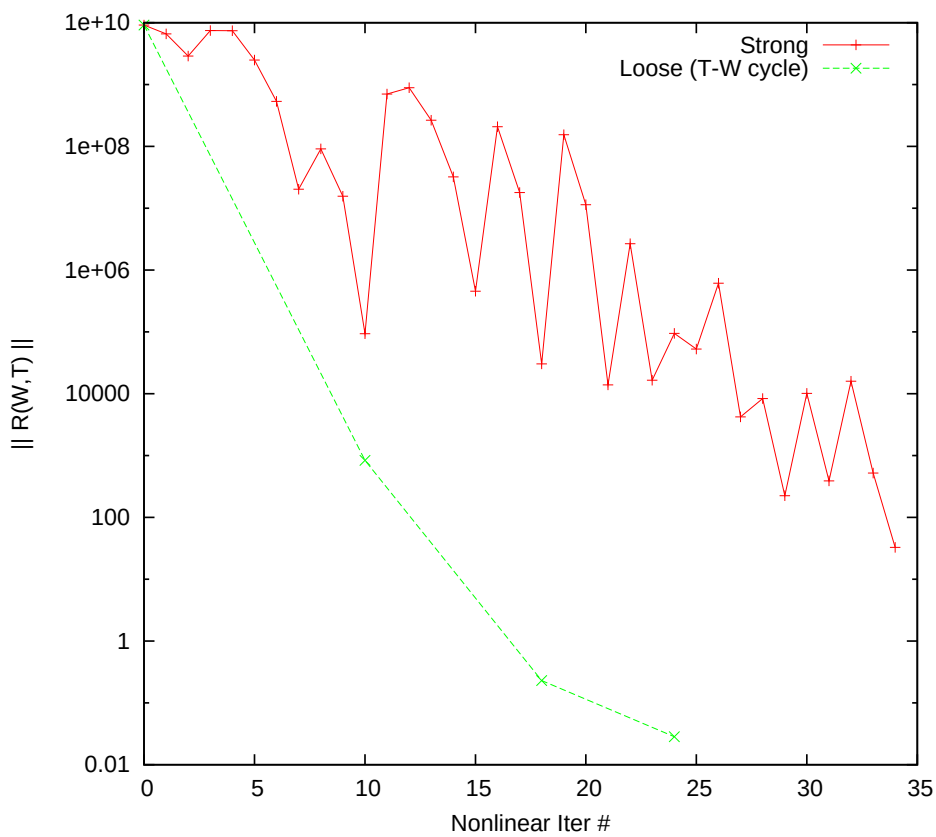


Figure 8. Comparison of convergence behavior for loose and strong coupling methods represented as coupled residual norm $||\mathbf{R}(\mathbf{W}, T)||$ vs number of nonlinear iterations for the Blunt-wedge problem. Note that the number of nonlinear iterations for loose coupling corresponds to those for the sequence of Premo problems.

VI. Conclusions

We have performed thermally coupled fluid-structural simulations of compressible external flow over two different thermally conducting solid bodies. We performed the coupled solves using both loose and strong strategies while requiring no additional information from either physics code beyond what each already employs when doing stand-alone solves. Strong coupling performs at least as well as loose coupling for a relatively mild problem, but encounters significant difficulty in converging a problem involving more difficult flow with non-conforming meshes at the coupling interface. Current work is aimed at addressing and rectifying the source of convergence breakdown. Lastly, we note that the the presence of a thermally conducting body dramatically alters the fluid flow profiles compared to either isothermal or adiabatic idealizations imposed at the solid body interface.

Acknowledgments

Authors R.W.H. and T.M.S. would like to thank Matthew Barone for providing the blunt-wedge meshes.

References

[1]Heroux, M. A. and Willenbring, J. M., "Trilinos Users Guide," Tech. Rep. SAND2003-2952, Sandia National Laboratories, 2003.

[2]Edwards, H. C., "SIERRA framework version 3 : core services theory and design," Tech. Rep. SAND2002-3616, Sandia National Laboratories, Albuquerque, NM, 2002.

[3]Kolda, T. G. and Pawlowski, R. P., NOX home page. http://software.sandia.gov, 2005.

[4]Haselbacher, A. and Blazek, J., "Accurate and Efficient discretization of Navier-Stokes Equations on Mixed Grids," *AIAA Journal*, Vol. 38, No. 11, November 2000, pp. 2094–2102.

[5]Luo, H., Baum, J. D., and Lohner, R., "Edge-Based Finite Element scheme for the Euler Equations," *AIAA Journal*, Vol. 32, No. 6, June 1994, pp. 1180–1190.

[6]Smith, T. M., Hooper, R. W., Ober, C. C., Lorber, A. A., and Shadid, J. N., "Comparison of Operators for Newton-Krylov Method for Solving Compressible Flows on Unstructured Meshes," *AIAA Paper 2004-0743*, January 2004.

[7]Smith, T. M., Hooper, R. W., Ober, C. C., and Lorber, A., "Intelligent Nonlinear Solvers for Computational Fluid Dynamics," *AIAA Paper 2006-1483*, January 2006.

[8]Radenac, E., Gressier, J., Millan, P., and Giovannini, A., "A Conservative Coupling Numerical Method for Transient Conjugate Heat Transfer," *Int. Conf. on Computational Methods for Coupled Problems in Science and Engineering*, CIMNE, Barcelona, 2005.

[9]Giles, M. B., "Stability and Accuracy of Numerical Boundary Conditions in Aeroelastic Analysis," *Int. J. Num. Meth. in Fluids*, Vol. 24, 1997, pp. 739–757.

[10]Knoll, D. A. and Keyes, D. E., "Jacobian-free Newton-Krylov methods: a survey of approaches and applications," *J. Comp. Phys.*, Vol. 132, No. 2, 2004, pp. 357–397.