47th AIAA Aerospace Sciences Meeting Including The New Horizons Forum and Aerospace Exposition
5 - 8 January 2009, Orlando, Florida

AIAA 2009-670

Implicit implementation of material response and moving meshes for hypersonic re-entry ablation

Alexandre Martin* and Iain D. Boyd†

Department of Aerospace Engineering, The University of Michigan, Ann Arbor, MI, 48109, USA

A one-dimensional material response implicit solver with surface ablation and pyrolysis is implicitly coupled to LeMANS, a CFD code for the simulation of weakly ionized hypersonic flows in thermo-chemical non-equilibrium. Following verification and validation of the blowing wall boundary conditions and ablation coupling, the implementation of a moving mesh algorithm in the flow solver is proposed. A set of simple test-cases is presented, proving that the method used preserves the fluxes. Using the well defined case of the IRV-2 vehicle, the implicit coupling method is tested over a re-entry trajectory.

Nomenclature

Symbols

$\mathbf{A}$	Jacobian matrix
a	Speed of sound
$\mathbf{C}$	Source term vector
$\mathbf{e}, E$	Energy
$\mathbf{F}$	Inviscid flux vector
$\mathbf{F_d}$	Diffusive flux vector
$\mathbf{F_n}$	Normal total flux vector
Fo	Forchheimer number
$\mathbf{G}$	Partial flux vector, $\mathbf{G} = \mathbf{F} - \mathbf{F_d} - \mathbf{u}\mathbf{U}$
h	Enthalpy
$\mathbf{I}$	Identity matrix
$\mathbf{J}$	Directional species diffusion flux tensor
K	Permeability
l	Characteristic length
l_s	Characteristic length of the volume increment
$\mathbf{L}$	Eigenvector similarity transformation matrix
$\dot{m}'''$	Volumetric mass source term
$\dot{m}$	Mass flux
p	Pressure
p_η	Conserved pressure in the normal direction
$\dot{q}''$	Internal heat flux
q	Surface heat transfer rate

$\mathbf{R}$	Right hand side term
$\mathbf{r}$	Distance vector
$\mathbf{S}, S$	Surface
t	Time
T	Temperature
T_v	Vibrational temperature
$\mathbf{U}$	Conservative vector
$\mathbf{u}, u, v$	Velocity
$\mathbf{w}, w$	Source term, Node velocity
$\bar{w}_n$	Mean normal face velocity
v'	Superficial velocity; $v' = \phi v$
V	Volume
x, y	Coordinates
Y_i	Species mass fraction
$\mathbf{\Lambda}$	Eigenvalue matrix
β	Forchheimer coefficient
μ	Dynamic viscosity
ρ	Density
τ	Viscous tensor
ϕ	Porosity
ψ	Damping factor

*Research Fellow, AIAA Member.

†Professor, Associate Fellow AIAA.

Subscripts and Superscripts

0	Non blowing	r	Rotational
c	Char	s	Solid
cv	Control volume	$\mathbf{T}$	Transposed
cs	Control surface	v	Virgin, Vibrational
e	Electron-Electronic	w	Wall
f	Fixed mesh	x, y, z	Coordinates
g	Gas	∞	Freestream

I. Introduction

The Thermal Protection System (TPS) of a re-entry vehicle is one of the key components of its design. The material used for the TPS can be classified into two main categories: ablative materials, as in the one used on Apollo missions, and non-ablative materials, such as the ceramic tiles used on the space shuttle. The theory behind the use of ablators is quite simple; the energy absorbed by the removal of material from the surface is not used to heat the TPS, thus keeping the vehicle at a reasonably "cold" temperature. In order to properly model the heat rate at the surface of the vehicle, the ablating boundary condition must take into account many phenomena: surface recession, wall temperature, blowing rates, gas composition, surface chemistry, etc. One way to account for effects of the TPS on the surface flow is to link a material response code to the flow solver. To dynamically account for the effects of the surface movements on the flow field, however, the flow boundary must also be allowed to move as the surface recesses.

As the next step in a continuing project to improve heat and ablation rate computations on re-entry vehicles, the present study outlines the verifications and validations of the implementation of a moving mesh algorithm in the hypersonic flow solver LeMANS, as well as the coupling with the material response code MOPAR. Comparisons with analytical and empirical solutions are presented, as well as an established test case.

II. LeMANS: an unstructured tridimensional Navier-Stokes solver for hypersonic nonequilibrium aerothermodynamics

II.A. Overview

LeMANS is a finite volume Navier-Stokes solver currently being developed at The University of Michigan.[1–4] The code assumes that the rotational and translational energy modes of all species can be described by their respective temperatures T_r and T, and that the vibrational energy mode of all species and the electronic energy can be described by a single temperature T_{ve}. The latter is computed using the species vibrational energy, modeled as a harmonic oscillator. The viscous stresses are modeled assuming a Newtonian fluid, using Stokes hypothesis, and the species mass diffusion fluxes are modeled using a modified version of Fick's law. Mixture transport properties are calculated using one of two models; the first uses Wilke's semi-empirical mixing rule with species viscosities calculated using Blottner's model and species thermal conductivities determined using Eucken's relation, and the other uses Gupta's mixing rule with species viscosities and thermal conductivities calculated using non-coulombic/coulombic collision cross section data. As for the heat fluxes, they are modeled according to Fourier's law for all temperatures. Finally, the source terms of the species conservation equations are modeled using a standard finite-rate chemistry model for reacting air in conjunction with Park's two-temperature model to account for thermal nonequilibrium effects on the reaction rates.

Numerically, the code has the capability to handle any mix of hexahedra, tetrahedra, prisms and pyramids meshes in 3D or triangles and quadrilaterals in 2D. Numerical fluxes between the cells are discretized using a modified Steger-Warming Flux Vector Splitting which has low dissipation and is appropriate to calculate boundary layers. A point or line implicit method is used to perform the time integration. The code has been extensively validated against experimental data, and has also been compared to other similar codes such as NASA Ames' DPLR[5] and NASA Langley's LAURA.[6]

II.B. Governing equations

With the approximations mentioned above, the conservation equations for a three-dimensional system can be written as

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot (\mathbf{F} - \mathbf{F_d}) = \mathbf{C} \tag{1}$$

where:

$$\mathbf{U} = \begin{pmatrix} \rho_1 \\ \vdots \\ \rho_{ns} \\ \rho u_x \\ \rho u_y \\ \rho u_z \\ E \\ E_{ve} \\ E_r \end{pmatrix} \qquad \text{and} \qquad \mathbf{C} = \begin{pmatrix} \dot{w}_1 \\ \vdots \\ \dot{w}_{ns} \\ 0 \\ 0 \\ 0 \\ 0 \\ \dot{w}_v \\ \dot{w}_r \end{pmatrix}$$

are the vector of conserved variables and the vector of source terms, respectively. In these expressions, $\rho_1 \ldots \rho_{ns}$ are the species densities, u_i are the bulk velocity components, E, E_{ve} and E_r are the total, the vibrational-electron-electronic and the rotational energy per unit volume of mixture, respectively.

The inviscid and diffusive flux matrices are given by

$$\mathbf{F} = \begin{pmatrix} \rho_1 u_x & \rho_1 u_y & \rho_1 u_z \\ \vdots & \vdots & \vdots \\ \rho_{ns} u_x & \rho_{ns} u_x & \rho_{ns} u_x \\ \rho u_x^2 + p & \rho u_y u_x & \rho u_z u_x \\ \rho u_x u_y & \rho u_y^2 + p & \rho u_z u_y \\ \rho u_x u_z & \rho u_y u_z & \rho u_z^2 + p \\ (E+p) u_x & (E+p) u_y & (E+p) u_z \\ E_{ve} u_x & E_{ve} u_y & E_{ve} u_z \\ E_r u_x & E_r u_y & E_r u_z \end{pmatrix}$$

and

$$\mathbf{F_d} = \left(\begin{array}{ccc} -J_{x,1} & -J_{y,1} & -J_{z,1} \\ \vdots & \vdots & \vdots \\ -J_{x,ns} & -J_{y,ns} & -J_{z,ns} \\ \tau_{xx} & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \tau_{yy} & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \tau_{zz} \\ \multicolumn{3}{c}{\mathbf{u}\boldsymbol{\tau} - (\mathbf{q}_t + \mathbf{q}_r + \mathbf{q}_{ve}) - (\mathbf{h}^{\mathbf{T}}\mathbf{J})} \\ \multicolumn{3}{c}{-\mathbf{q}_{ve} - (\mathbf{e}_{ve}^{\mathbf{T}}\mathbf{J})} \\ \multicolumn{3}{c}{-\mathbf{q}_r - (\mathbf{e}_r^{\mathbf{T}}\mathbf{J})} \end{array} \right)$$

where p is the pressure, τ_{ij} is the viscous tensor components and $\mathbf{q}_t$, $\mathbf{q}_r$ and $\mathbf{q}_{ve}$ are respectively the directional translational, rotational and vibrational-electron-electronic heat fluxes vector. Moreover, $\mathbf{h}$ is the species enthalpy vector and $\mathbf{J}$ is the directional species diffusion flux tensor. More details on these equations and on the modeling of the individual terms can be found in Ref. 2.

Equation 1 can be reduced to a simple vector form which splits the flux tensor using the conservative vector:

$$\frac{\partial}{\partial t}\begin{pmatrix}\boldsymbol{\rho}\\ \rho\mathbf{u^T}\\ E\\ E_{ve}\\ E_r\end{pmatrix} + \nabla\cdot\left[\begin{pmatrix}\boldsymbol{\rho}\\ \rho\mathbf{u^T}\\ E\\ E_{ve}\\ E_r\end{pmatrix}\mathbf{u} + \begin{pmatrix}0\\ \mathbf{I}p\\ \mathbf{u}p\\ 0\\ 0\end{pmatrix} - \begin{pmatrix}-\mathbf{J}\\ \boldsymbol{\tau}\\ \mathbf{u}\boldsymbol{\tau} - (\mathbf{q}_t+\mathbf{q}_r+\mathbf{q}_{ve}) - (\mathbf{h^T J})\\ -\mathbf{q}_{ve} - (\mathbf{e}_{ve}^{\mathbf{T}}\mathbf{J})\\ -\mathbf{q}_r - (\mathbf{e}_r^{\mathbf{T}}\mathbf{J})\end{pmatrix}\right] = \begin{pmatrix}\dot{\mathbf{w}}\\ 0\\ 0\\ \dot{w}_v\\ \dot{w}_r\end{pmatrix}$$

which can be re-reduced to:

$$\frac{\partial\mathbf{U}}{\partial t} + \nabla\cdot(\mathbf{U}\mathbf{u} + \mathbf{G}) = \mathbf{C} \tag{2}$$

III. MOPAR: a material response code

The material response code used in this validation is currently being developed at The University of Michigan, and is called MOPAR (Modeling of Pyrolysis and Ablation Response). The code is based on a Control Volume Finite-Element Method (CVFEM) developed at North Carolina State University and Sandia National Laboratory,[7–11] and is used to model surface ablation as well as pyrolysis. The problem is described by the following four equations:

Mixture Energy Equation

$$\frac{d}{dt}\int_{cv}\rho E dV - \int_{cs}\rho h v_{cs} dS + \int_{cs}\phi\rho_g h_g v_g dS \; + \int_{cs}\dot{q}'' dA = 0 \tag{3}$$

Solid Phase Equation

$$\frac{d}{dt}\int_{cv}\rho_s dV - \int_{cs}\rho_s v_{cs} dS - \int_{cv}\dot{m}_s''' dV = 0 \tag{4}$$

Gas Phase Continuity Equation

$$\frac{d}{dt}\int_{cv}\phi\rho_g dV - \int_{cs}\phi\rho_g v_{cs} dS + \int_{cs}\phi\rho_g v_g dS - \int_{cv}\dot{m}_g''' dV = 0 \tag{5}$$

Momentum Equation: Forchheimer's Law

$$\frac{\partial P}{\partial x} = -\frac{\mu}{K}v_g'(1 + \mathrm{Fo}) \tag{6}$$

The first term of Eqs. 3 to 5 accounts respectively for the energy, solid mass, and gas mass content, and the second term for the grid convection. The third term of Eqs. 3 and 5 is the gas flux, and the last term of Eqs. 4 to 5, the source term. As for the last term of Eq. 3, it accounts for the heat conduction within the solid. In Eq. 6, Forchheimer's Law, Fo, the Forchheimer Number, is given by

$$\mathrm{Fo} = \frac{\beta K \rho v_g'}{\mu}$$

This number indicates when microscopic effects (pore-size) are perceivable at a macroscopic (geometry size) level. In this formulation, it is easy to see that when $\mathrm{Fo} \ll 1$, the equation simply reduces to Darcy's law. Therefore, it is more logical to use the Forchheimer Number to predict non-Darcian flow, and thus more rigorous to use Forchheimer's law in the code.

The first two of these four equations are solved implicitly on an arbitrary contracting grid using Landau coordinates. Newton's method for non-linear systems is used to solve both equations sequentially. The third equation is straight forward, and does not need to be solved numerically. As for Forchheimer's law, it is explicitly solved for v_g and directly integrated in the gas-phase continuity equation.

In addition to the improvement in the momentum equation,[12] the present code also takes into account variable coordinate systems (cylindrical and spherical), and allows ablation on both sides of the domain, using a new tri-diagonal solver.[13] The code is validated against experimental data and code-to-code comparisons, as discussed in Ref. 12.

IV. Moving mesh

IV.A. Overview

In order to complete the implicit coupling of the thermal response code MOPAR to the hypersonic CFD code LeMANS, moving mesh capabilities are added to the latter code. The method chosen has been proposed by Ref. 14; the Geometric Conservation Law, is solved implicitly in the discretized governing equations. This method has the advantage of being valid for both explicit and implicit schemes, works on any kind of mesh cells, is easy to implement in a finite volume scheme developed for fixed meshes, and keeps the order of the scheme.

To implement the method, two sections of LeMANS must be modified: the computation of the flux, and the time integration.

IV.B. IGCL formulation

Integrated over an arbitrary volume, Eq. 2 can be written for an arbitrarily moving system (Arbitrary Lagragian-Eulerian formulation):

$$\frac{\partial}{\partial t}\int_{cv}\mathbf{U}dV + \oint_{cs}(\mathbf{Uu} - \mathbf{Uw} + \mathbf{G})\cdot\mathbf{dS} = \int_{cv}\mathbf{C}dV \tag{7}$$

where $\mathbf{S}$ is the vector area of surface cs, which is one of the enclosing surfaces of the time-dependent volume cv, $\mathbf{u}$ is the velocity of the fluid and $\mathbf{w}$ is the arbitrarily specified velocity of cs.

According to Ref. 14, the discretization of the governing equation for a finite volume scheme is:

$$V^{n+1}\mathbf{U}^{n+1} - V^n\mathbf{U}^n = -\sum_k \mathbf{F}_{\mathbf{n},k}S_k\Delta t + V\mathbf{C}\Delta t \tag{8}$$

where $\mathbf{F_n} = \mathbf{Fn^T}$ of which $\mathbf{n} = (n_x\ n_y\ n_z)$ is the normal vector of the face S. When applied to Eq. 2, we obtain:

$$\mathbf{F_n} = \mathbf{U}(u_n - \bar{w}_n) + \mathbf{Gn^T}$$

where:

$$\bar{w}_n = \frac{\Delta V}{S\Delta t}$$

and where ΔV is the face volumetric increment calculated according to the type of elements. These quantities are calculated in such a way that the volumetric increment is balanced by the flux generated by the face movement, so that in the end, the conserved quantity remains unchanged by the moving mesh. For a planar 2D geometry, the volume increment is:

$$\Delta V = \Delta t\ \mathbf{w_0} \times \Delta\mathbf{r}_{\mathbf{12}}^{n+1/2}$$

where $\mathbf{w_0}$ is the average velocity of the two nodes of the side, and $\mathbf{r}_{\mathbf{12}}^{n+1/2}$ is the time averaged side vector. For axi-symmetric geometry the volume becomes:

$$\Delta V = \frac{\Delta t}{6}(2y_1^{n+1/2} + y_2^{n+1/2})\mathbf{w_1} \times \Delta\mathbf{r}_{\mathbf{12}}^n + \frac{\Delta t}{6}(2y_2^{n+1/2} + y_1^{n+1/2})\mathbf{w_2} \times \Delta\mathbf{r}_{\mathbf{12}}^n + \frac{\Delta t^2}{4}(y_1^{n+2/3} + y_2^{n+2/3})\mathbf{w_1} \times \mathbf{w_2}$$

For 3D tetrahedron volumes, the expression is:

$$\Delta V = \Delta t\ \mathbf{w_0} \cdot \mathbf{S}^{n+1/2} + \frac{\Delta t^3}{24}\mathbf{w_1} \cdot \mathbf{w_2} \times \mathbf{w_3}$$

where $\mathbf{w_0}$ is now the average velocity of the three nodes of the side, and $\mathbf{S} = \frac{1}{2}\Delta\mathbf{r_{12}} \times \Delta\mathbf{r_{13}}$. This expression takes into account the fact that in 3D, the order in which the nodes are moved leads to different facial volumes. This expression is therefore an average of all the possible movement combinations. It is to be noted that this last equation can be used for any 3D volumetric cells by dividing them into multiple tetrahedrons.

IV.C. Flux splitting

The Jacobian matrix needed to compute the Steger-Warming Flux Splitting Scheme used by the inviscid part of the governing equations is :

$$\mathbf{A} = \mathbf{A}^f - \mathbf{I}\bar{w}_n$$

where superscript f refers to the value calculated for a fixed mesh. The eigenvalue matrix of $\mathbf{A}$ is therefore:

$$\mathbf{\Lambda} = \mathbf{\Lambda}^f - \mathbf{I}\bar{w}_n$$

As for the similarity transformation matrix $\mathbf{L}$, constructed from the eigenvectors, it is identical to the one calculated for a fixed mesh.

$$\mathbf{L} = \mathbf{L}^f$$

This development shows that in order to add moving grid capability to the Flux Splitting Scheme of LeMANS, only the eigenvalues need to be modified.

IV.D. Implicit time integration

For the types of hypersonic problems solved using LeMANS, an implicit time integration is necessary to take advantage of the larger allowable time steps. Using a Taylor expansion on Eq. 8, the implicit time integration becomes

$$\mathbf{R}^{t+1} = \mathbf{R}^t + \left[\frac{\partial \mathbf{R}}{\partial \mathbf{U}}\right]^t \left(\mathbf{U}^{t+1} - \mathbf{U}^t\right) + \left[\frac{\partial \mathbf{R}}{\partial V}\right]^t \left(V^{t+1} - V^t\right)$$

where $-\mathbf{R}$ is the right-hand-side of Eq. 8. After some manipulation, the time integration becomes:

$$\mathbf{U}^{t+1} = \mathbf{U}^t + \left[\frac{V^t + \Delta V}{\Delta t}\mathbf{I} + \left[\frac{\partial \mathbf{R}}{\partial \mathbf{U}}\right]^t\right]^{-1} \left(-\mathbf{R}^t - \left[\frac{\partial \mathbf{R}}{\partial V}\right]^t \Delta V - \mathbf{U}^t \frac{\Delta V}{\Delta t}\right)$$

It is to be noted that in order to balance the flux, it is necessary to evaluate the source term using the volumetric time increment: this translates to:

$$\mathbf{R}^t = \sum_k \mathbf{F}^t_{\mathbf{n},k} S^t_k - (V^t + \Delta V)\mathbf{C}$$

IV.E. Mesh movement description

Even though the mathematics of the moving mesh is fairly simple, the question of how the mesh moves needs to be defined. In the context of ablation, because only the wall moves, the rest of the mesh can simply follow proportionally, using a perpendicular line between the wall and the inflow as guidance.

The algorithm first identifies the closest wall nodes for every node of the mesh. Then, a reference length is attribute to each wall node; this length corresponds to the distance between the wall node and the farthest mesh point attributed to that node. Each mesh node is then non-dimensionalized according to these reference lengths. With this, each node has a coordinate system with a value ranging from 0 to 1, distributed on a perpendicular line starting at the wall. This type of coordinate system is very similar to Landau coordinates.

Even though this method is far from being general, it is sufficient for the geometry usually studied in LeMANS. Complex geometries, especially if the surface is not physically uniform, would probably cause problems with this method. One way to generalize it would be to apply a smoothing scheme (averaging) to the relative coordinate in order to account for acute discrepancies in neighboring nodes.

IV.F. CFL condition

Since a new flux is introduced to the equations, the CFL condition needs to be adjusted accordingly. In the context of an implicitly coupled ablation-flow code, the recession distance is imposed: the node velocity is therefore a function of the time step. This translates to:

$$\Delta t = \frac{l}{\sqrt{(u_n - \bar{w}_n)^2 + u_t^2 + u_p^2} + a}$$

This equation can be solved for Δt, and yields:

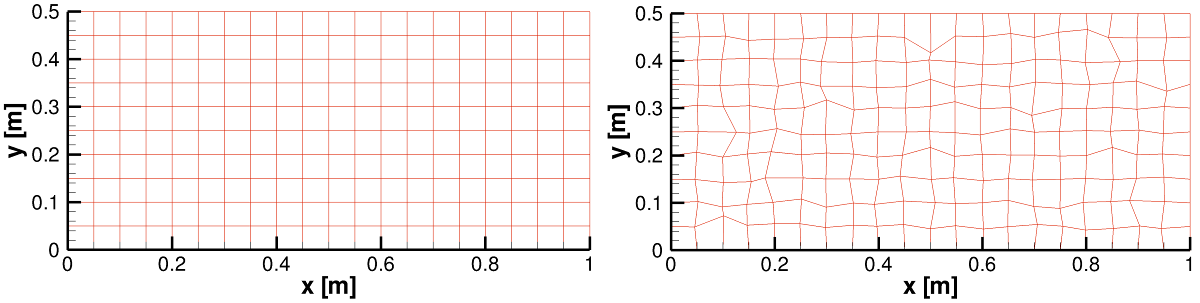
$$\Delta t = \frac{l^2 - l_s^2}{(la - l_s u_n) + \sqrt{(la - l_s u_n)^2 - (l^2 - l_s^2)(a^2 - |u|^2)}}$$

where l is the characteristic length of the cell, $l_s = \frac{\Delta V}{S}$ the characteristic length of the volume increment of the face, and a is the speed of sound.

V. Verification and validation

V.A. Mass, momentum and energy conservation

The first verification is performed on a simple geometry: a 0.5 × 1.0 m closed box, discretized with 10 × 20 cells. The walls are adiabatic, and the gas (air), at rest, is initially at 15 000 K, at a density of 0.1 kg/m^3. Because of chemistry, the temperature eventually relaxes to a lower, equilibrium value, although the total density remains the same (no change in physical volume). The mesh is moved randomly for 75 iterations; the final mesh is shown in Fig. 1. Figure 2 presents the isovalues for translational temperature, total density and velocity: in order to show the order of precision at which these quantities are computed, their theoretical value have been subtracted, so that they all should be zero (ρ and T are non-dimensionalized). As shown on the figures, every one of them are zero, at machine precision.



(a) Original mesh

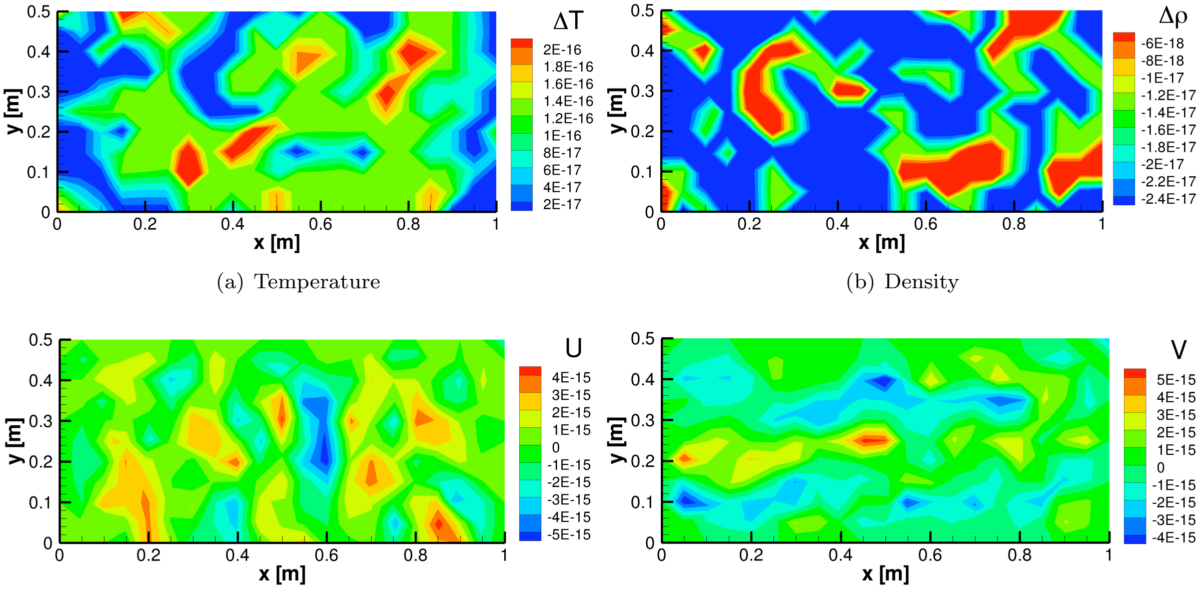
(b) Final mesh

Figure 1. Mesh movement for the first verification

V.B. Shock tube

The next verification demonstrates that the time integration on the moving mesh is preserved, and that the flux across the moving faces is computed correctly for a moving shock. The geometry consists of a 1 × 0.5 m closed box containing air at a temperature of 350 K; the left side has a density of 1 kg/m^3, and the right side at 0.01 kg/m^3. The shock is simulated over a period of 2.0×10^{-4} s. The initial mesh consists of 200 × 100 squares, and moves randomly (random walk) every iteration for a maximum allowable distance of 5×10^{-5} m. The simulation is run implicitly, without diffusion, with a CFL number of 1, so the simulation is time accurate. Using the normal CFL condition, the solver uses 53 iterations to reach the final simulation time.

Figure 3 shows a close-up view of the state of the mesh after having been moved randomly 53 times; as can be seen, some nodes have moved considerably over the time period. Figures 4 compare the moving mesh solution to the same solution on a fixed grid. The results show excellent agreement, even for the temperature, a quantity usually very sensitive to compression effects. The largest discrepancy is at the shock front; this is easily explained by the fact that the cells resolving the shock have their front consistently moved back and forth, which, in the end, can lead to a small displacement in the shock position. Figure 5 shows the uniformity of the solution in the y direction; it can be seen that the solution remains smooth and uniform, although multiple mesh movements have occurred.



(a) Temperature

(b) Density

(c) x-velocity

(d) y-velocity

Figure 2. Difference in computed and theoretical values after multiple random movements of the mesh.

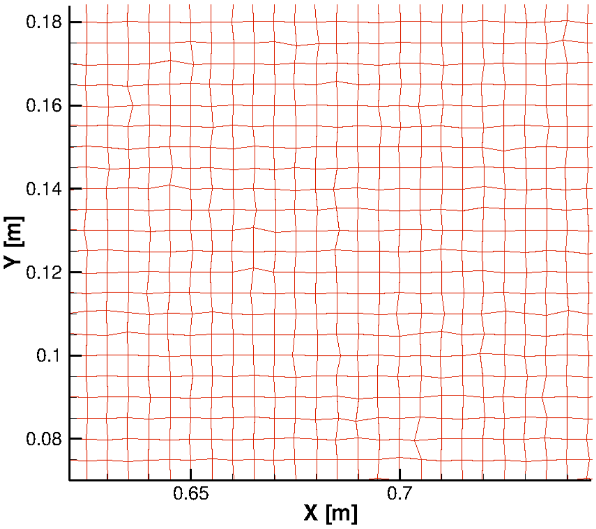
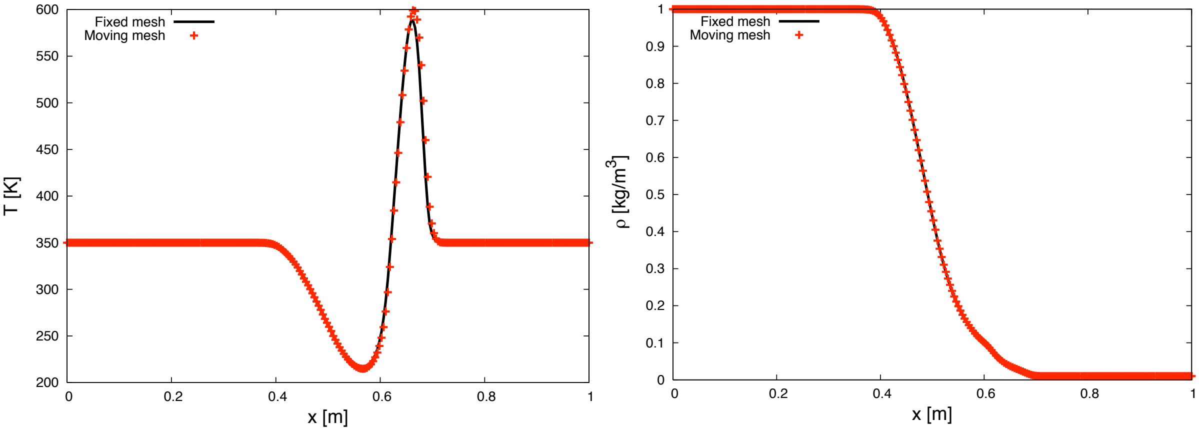


Figure 3. Close-up view of the final mesh after 53 random movements



(a) Temperature

(b) Density

Figure 4. Comparison of the shock tube problem between a fixed mesh and a randomly moved mesh.

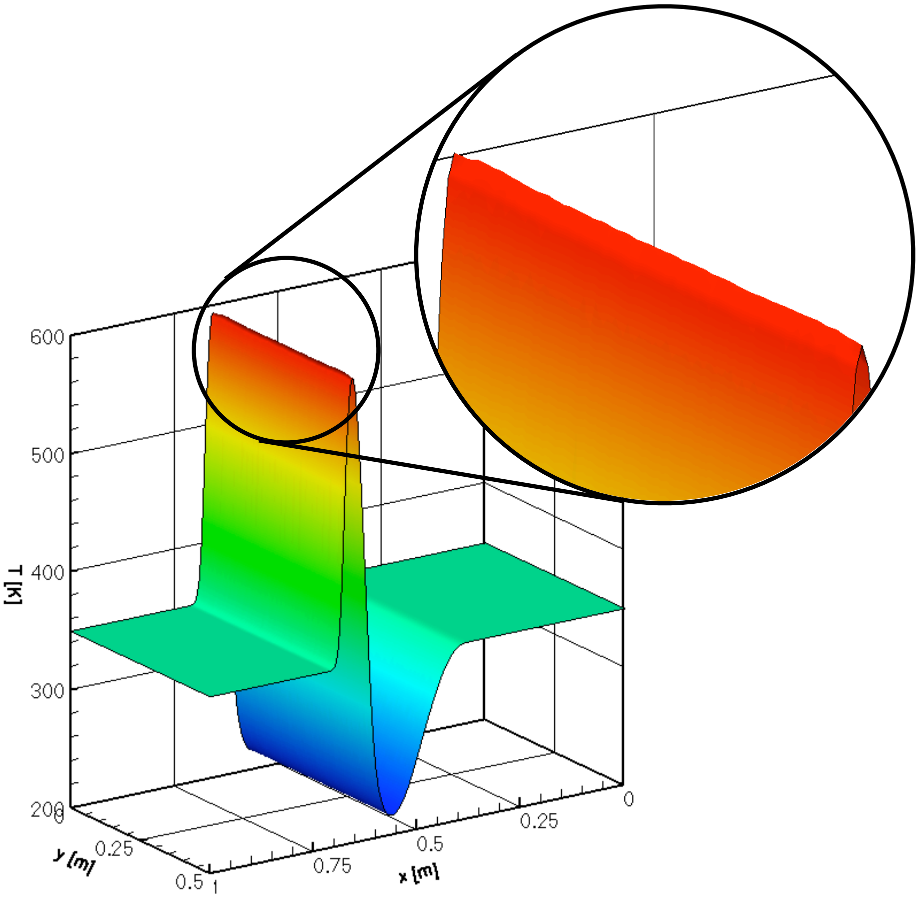


Figure 5. 3D view of the shock tube solution using a moving mesh

VI. Implicit coupling

VI.A. Coupling strategies

Because re-entry simulations are being performed by computing steady-state solutions at multiple points of a discretized trajectory, the thermal response code is directly integrated as a boundary condition subroutine of LeMANS, thus taking advantage of the implicit nature of the code as well as the aggressive CFL ramping. The method used is similar to the one described in Ref. 15, and is illustrated in Fig. 6. Since MOPAR is 1-D, normal solution lines within the wall are traced at each boundary cell, and are computed sequentially. Because there is no need to compute the material response at every flow field iteration, MOPAR is called at a pre-determined number of iterations. This coupling method was previously presented,[12] but without having the fluid mesh being adapted. Even though the method showed good results and proved to be efficient and robust, the material response was not accurate since the shock wave was calculated from the initial state of the geometry, without talking into account the recession of the wall.

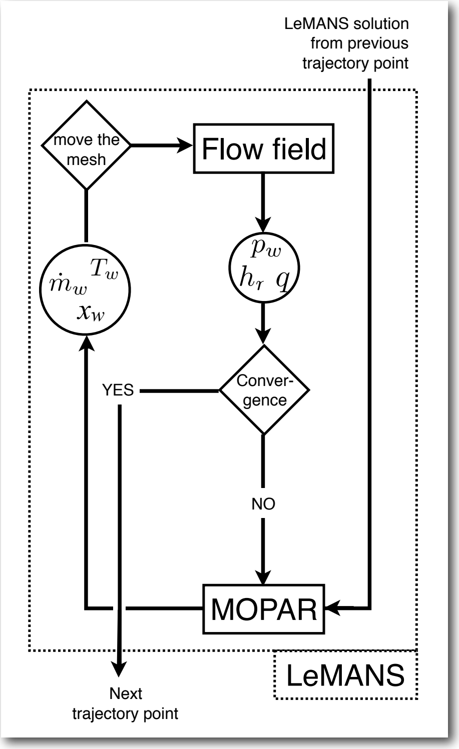


Figure 6. Coupling procedure for the integration of MOPAR in LeMANS

Three modifications are applied at the interface between the two codes to preserve stability and accelerate convergence. First, the convective heat flux used in MOPAR is adjusted using a hot-wall correction:[16]

$$q_{hw} = q_{cw}\Omega_{hw} = q_{cw}\left[\frac{\rho(T_{hw}, p_w)\mu(T_{hw})}{\rho(T_{cw}, p_w)\mu(T_{cw})}\right]^{0.1}$$

This method uses a boundary layer approximation to guess what the actual heat flux is supposed to be, and disappears once the wall has reached its converged values (i.e. when $T_{cw} = T_{hw}$, then $q_{cw} = q_{hw}$). The use of this correction speeds up the convergence of the wall temperature and ablation rates, and prevents the wall conditions from affecting the convergence of the flow field.

The second modification consists of damping the updated values at the wall. Instead of using the actual computed value given by the material response for recession distance, wall temperature and blowing rates, the value is combined with the one computed at the previous iteration:

$$T_{\text{assigned}} = (1-\psi)T_{\text{old}} + \psi T_{\text{computed}}$$

The ψ parameter, usually set to 0.75, prevents the solution from being "caught" in an oscillation between two values, and also prevents the values from being over-evaluated (or under) while the solution is still changing.

Finally, a modification is made to the moving boundaries condition by not imposing the mesh velocity at the wall. This way, while converging, the wall does not generate unphysical shock waves each time the mesh is moved back and forth.

Figure 7 shows a comparison between the convergence history of a simulation of the IRV-2 vehicle, at re-entry point 2 of Table 1, using a re-radiating wall temperature boundary condition, the coupled method with no moving mesh, and the coupling method with the moving mesh. The simulation uses ramping CFL number, with initial conditions set to the converged solution at the previous trajectory point, using their respective wall temperature boundary. For this coupled, case MOPAR is called every 100 iterations. The first observation to make is that it is impossible to see where MOPAR is called on the non-moving mesh curves for this particular simulation; this is explained by the fact that the residue L_2 norm at the wall is always smaller than in the rest of the flow field. It is, however, quite different for moving mesh curve: spikes can be seen on the curves each time the material response code is called. Those spikes are caused by the fact that the residue L_2 norm is calculated using the right-hand-side of equation 8; however, using this, the volume increment due to the moving mesh is not taken into account, and causes the spikes in the convergence curves. The addition of a term in the residue L_2 norm calculation would therefore remove the spikes, but since they give a good indicator of the degree of mesh movement, it was decided to leave them there. With these, it is possible to see when the freestream flow hits the wall (around 500 iterations), as well as the effects this sudden impact has on the wall conditions. We can also see the flow conditions have almost stabilized at 1000 iterations, having reached the convergence level of the flow field.

It can also be seen that the three curves follow the same trend, and show a similar convergence level. In fact, as shown on Fig. 8, these curves are virtually identical when plotted against the time of simulation instead of the number of iterations. Considering that the highest source of error (residue) is from the flow field, and not wall coupling, this is not surprising. The difference in time steps is caused by a combination of the additional fluxes and different cell sizes due to the mesh movement. Therefore, even though the wall clock time is obviously greater (around 10% in this particular test-case), the presented coupling method does not affect the convergence and robustness of the solution.

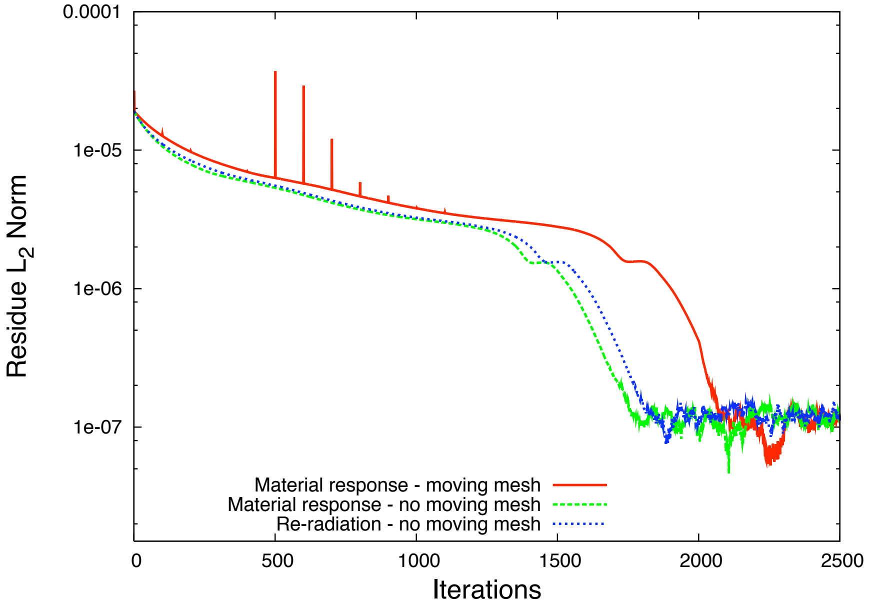


Figure 7. Convergence to steady state for the IRV-2 vehicle, at trajectory points 1 and 16: comparison between the regular solution and the coupled method

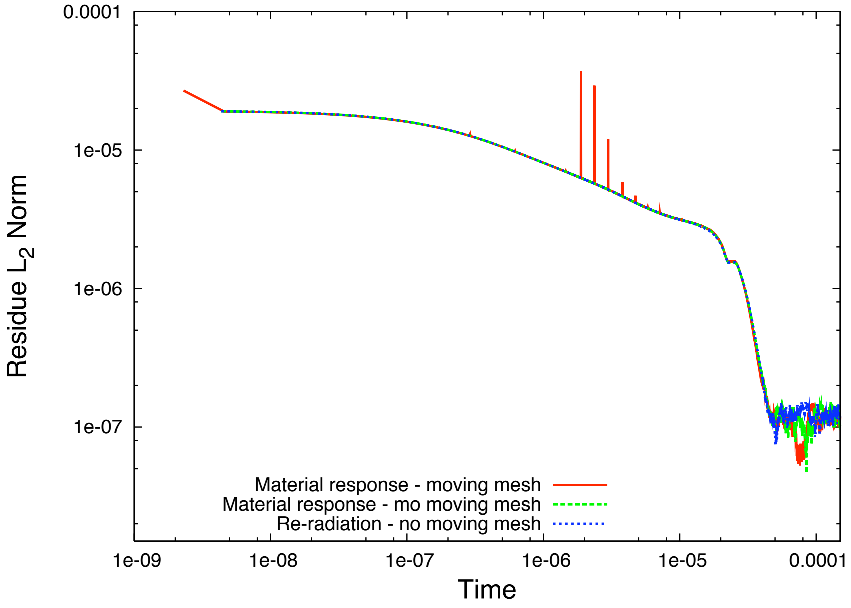


Figure 8. Convergence to steady state for the IRV-2 vehicle, at trajectory points 1 and 16: comparison between the regular solution and the coupled method

Finally, the method used to couple the recession distance must be discussed. Because MOPAR is employed in each cell neighboring the wall, the recession distance is therefore calculated at the face of the cell. However, the moving mesh scheme presented here uses node velocities to move the wall (and the rest of the cells). Therefore, the face recession distance must be transformed into node velocities. In order to do so, the displacement of each node is taken as an average of each of its neighboring faces, and then divided by the time step to obtain a velocity. It is assumed that the mesh is sufficiently dense on the wall that this averaging method is accurate, therefore preserving the shape of the wall.

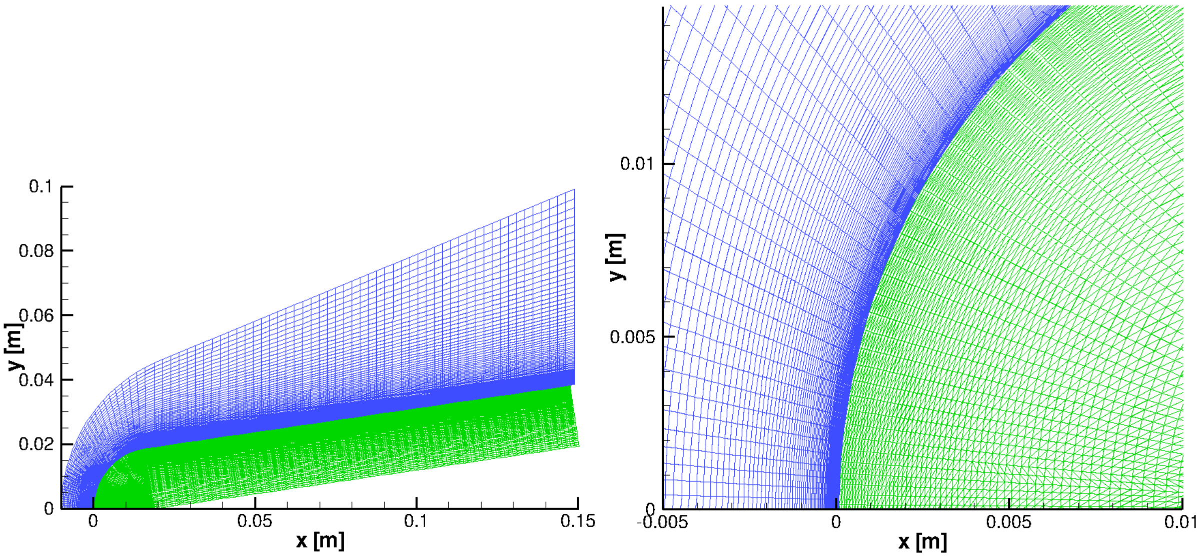
VI.B. IRV-2 test-case

In order to validate the implicit coupling between MOPAR and LeMANS, the well documented[15, 17] re-entry simulation of an IRV-2 vehicle is performed. This test-case provides good feedback since it can be compared to other CFD-material response code coupling scheme, as well as to the ASCC code which uses flight data to generate results. The freestream conditions used in the discretized trajectory are presented in Table 1, and the material properties are set to generic non-charring carbon, using the properties given by Ref. 7. The ablation rates are interpolated from thermochemical tables generated by ACE-SNL[18] for carbon in air. Re-radiation is also included at the boundary. The mesh used for the simulation is presented in Fig. 9; it is important to point out that the mesh for the material response is actually composed of straight lines, as described earlier. The triangular mesh presented in the figure is generated for post-processing analysis. First, the results (without the moving mesh) for the temperature at the stagnation point for the whole re-entry trajectory is shown in Fig. 10. The results are in the same range as those published in the literature; the discrepancy is due to the lack of proper chemistry in the flow, at the wall and for the material response. It is to be noted that the overvaluation in the first part of the trajectory, as well as the smoothness of the curve, is due to the fact that the whole material response surface history is plotted, as opposed to only the value at the discretized trajectory points. For this particular example, the coupled algorithm proved to be fast and robust for all the trajectory points (the results were obtained in roughly 18 hours, using 32 processors).

Figure 11 presents the results using the moving mesh algorithm presented here. As can be see for the first two trajectory points, the surface remains smooth even though the coupling is aggressive in this case,

Table 1. Freestream conditions for the re-entry trajectory of the IRV-2 vehicle (from Ref. 15)

Trajectory point	Time [s]	Velocity [m/s]	Temperature [K]	Density [kg/m^3]
1	0.00	6780.6	227.81	1.2505×10^{-4}
2	4.25	6788.3	258.02	5.0454×10^{-4}
3	6.75	6785.2	270.65	1.1344×10^{-3}
4	8.75	6773.0	261.40	2.2593×10^{-3}
5	10.25	6752.4	250.35	3.9957×10^{-3}
6	11.50	6722.0	241.50	6.4268×10^{-3}
7	12.50	6684.3	234.30	9.5832×10^{-3}
8	13.25	6644.9	228.76	1.3145×10^{-2}
9	13.95	6596.7	226.91	1.7313×10^{-2}
10	14.75	6527.1	224.73	2.4310×10^{-2}
11	15.50	6428.3	222.35	3.5348×10^{-2}
12	16.25	6286.6	219.47	5.5888×10^{-2}
13	17.00	6091.7	216.65	9.1741×10^{-2}
14	17.75	5836.4	216.65	1.5635×10^{-1}
15	18.25	5631.8	216.65	2.2786×10^{-1}
16	18.50	5519.6	216.65	2.7946×10^{-1}
17	18.75	5401.2	216.65	3.3743×10^{-1}
18	19.00	5277.1	221.31	3.9840×10^{-1}
19	19.50	5014.3	236.86	5.3196×10^{-1}
20	20.00	4736.5	252.11	6.9366×10^{-1}



(a) Overall view of the meshes

(b) Close up view of the meshes

Figure 9. Flow field mesh and material response mesh used for the simulation of the re-entry of the IRV-2 vehicle, at trajectory point 2 of Table 1

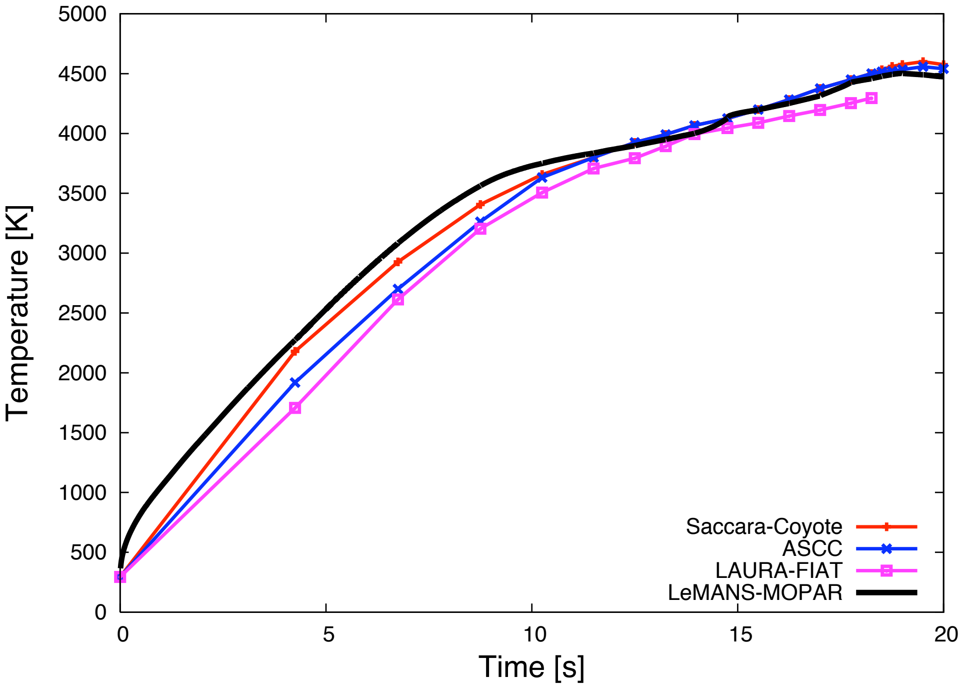


Figure 10. Temperature at the stagnation point of the IRV-2 re-entry vehicle at the trajectory points of Table 1: comparison with numerical results of Ref. 15 and Ref. 17

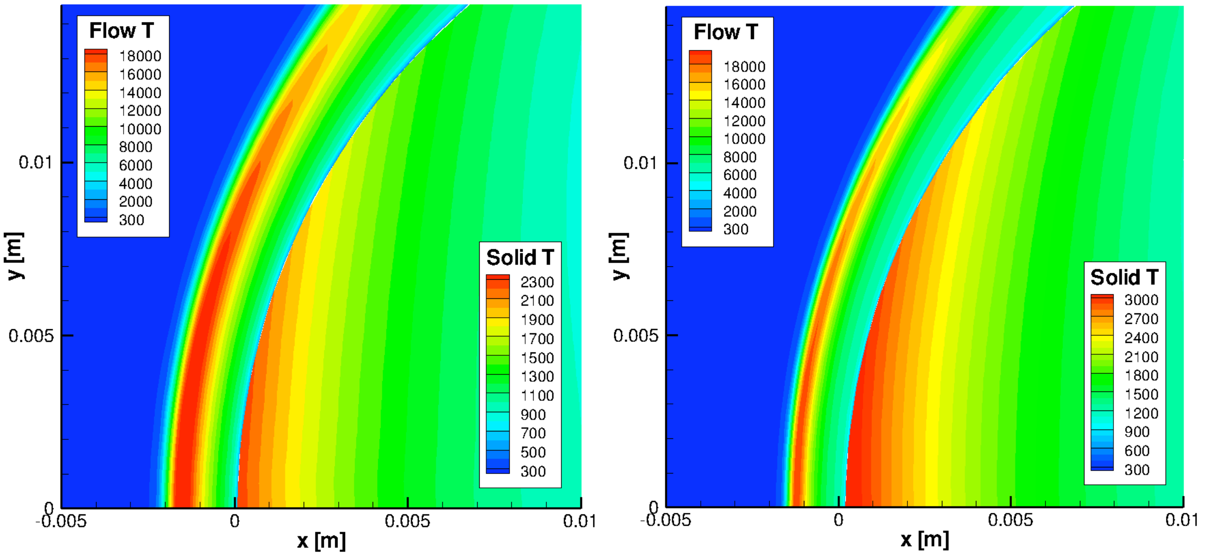
as the vehicle ablates towards its optimal shape. It is also important to note that even though the mesh is relatively rough at the wall, especially in the tangential direction, the face-to-nodes averaging remains very accurate. Finally, the recession distance is compared to results obtain with the ASCC code,[15] which includes flight data. Figure 12 presents this comparison, for the first two trajectory points; as can be seen, the results are within the expected range, especially given that thermodynamic and ablation rates values for a generic carbon-carbon ablator are used.

VII. Conclusion

As part of a continuing project to improve heat and ablation rate modeling on hypersonic re-entry vehicles, verification and validation of a moving mesh algorithm is presented. The first test case insures that no mass, energy or momentum is created when the mesh is aggressively moved. The second test validates that the mesh movement does not change the ability of the code to capture moving shocks. For both cases, the results are well within the expected range. These tests also show that the complex non-equilibrium and chemistry source term as well as the very stiff fluxes are unaffected by the moving mesh. In order to validate the coupling between between the flow solver LeMANS and the material response code MOPAR, the simulation of the initial phase of the re-entry trajectory of an IRV-2 vehicle is presented. Although wall clock-time is obviously affected when computing material response, the results show that the simulation time and convergence are unaffected by the coupling method and moving mesh. The results presented are within the expected range, and show promising results for the simulation of a complete trajectory.

Acknowledgments

The authors would like to thank the *Government of Québec* which, through the *Fonds de recherche sur la nature et les technologies*, provides a fellowship to the first author. Additional funding is provided by the *Constellation University Institutes Program*, under NASA grant NCC3-989.



(a) $t = 4.25$ s

(b) $t = 6.75$ s

Figure 11. Temperature distribution of the flow field and in the solid wall of the IRV-2 re-entry vehicle at trajectory points 2 and 3 of Table 1

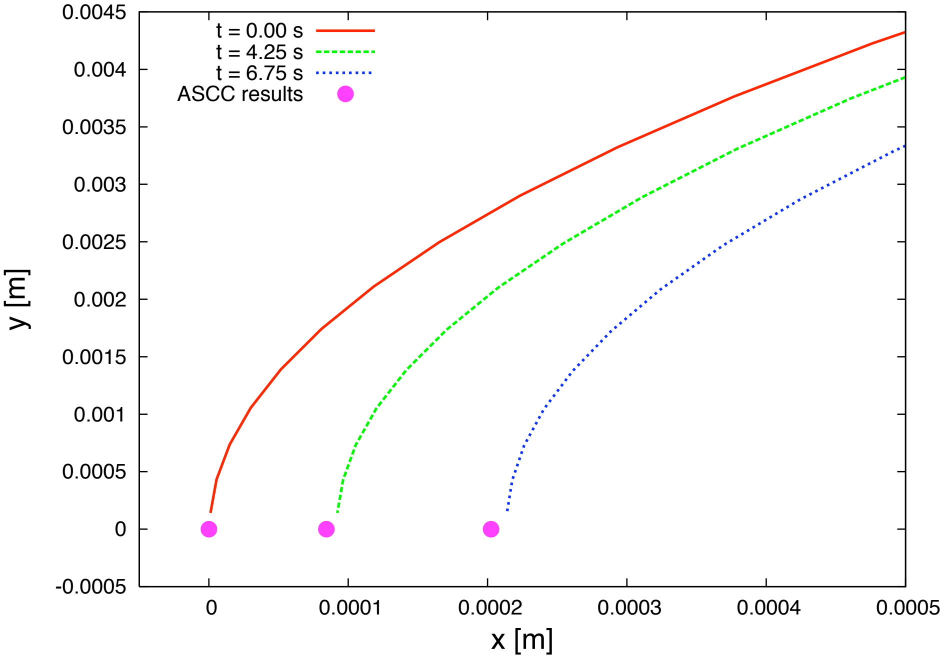


Figure 12. Surface recession comparison for the IRV-2 re-entry vehicle at trajectory points 1, 2 and 3 of Table 1

References

[1] Scalabrin, L. C. and Boyd, I. D., "Numerical Simulations of the FIRE-II Convective and Radiative Heating Rates," *39th AIAA Thermophysics Conference*, No. AIAA-2007-4044, Miami, FL, 25 - 28 June 2007, p. 17.

[2] Scalabrin, L. C., *Numerical Simulation of Weakly Ionized Hypersonic Flow Over Reentry Capsules.*, Ph.D. thesis, The University of Michigan, Ann Arbor, MI, 2007.

[3] Scalabrin, L. and Boyd, I. D., "Development of an Unstructured Navier-Stokes Solver for Hypersonic Nonequilibrium Aerothermodynamics," *38th AIAA Thermophysics Conference*, No. AIAA-2005-5203, Toronto, Ontario, June 6-9 2005.

[4] Scalabrin, L. and Boyd, I. D., "Numerical Simulation of Weakly Ionized Hypersonic Flow for Reentry Configurations," *9th AIAA/ASME Joint Thermophysics and Heat Transfer Conference*, No. AIAA-2006-3773, San Francisco, CA, June 5-8 2006, p. 18.

[5] Wright, M. J., Candler, G. V., and Bose, D., "Data-Parallel Line Relaxation method for the Navier-Stokes equations," *AIAA Journal*, Vol. 36, No. 9, September 1998, pp. 1603–1609.

[6] Gnoffo, P. A., "Upwind-Biased, Point-implicit Relaxation Strategies for Viscous Hypersonic Flows," AIAA-1989-1972-CP, July 1989.

[7] Amar, A. J., *Modeling of One-Dimensional Ablation with porous Flow Using Finite Control Volume Procedure*, Master's thesis, North Carolina State University, Raleigh, NC, 2006.

[8] Amar, A. J., Blackwell, B. F., and Edward, J. R., "One-Dimensional Ablation with Pyrolysis Gas Flow Using a Full Newton's Method and Finite Control Volume Procedure," *39th AIAA Thermophysics Conference*, No. AIAA-2007-4535, Miami, FL, 25-28 June 2007, p. 41.

[9] Amar, A. J., Blackwell, B. F., and Edward, J. R., "One-Dimensional Ablation Using a Full Newton's Method and Finite Control Volume Procedure," *9th AIAA/ASME Joint Thermophysics and Heat Transfer Conference*, No. AIAA-2006-2910, San Francisco, California, 5-8 June 2006, p. 26.

[10] Amar, A. J., Blackwell, B. F., and Edwards, J. R., "One-Dimensional Ablation Using a Full Newton's Method and Finite Control Volume Procedure," *Journal of Thermophysics and Heat Transfer*, Vol. 22, No. 1, January 2008, pp. 72–82.

[11] Blackwell, B. F. and Hogan, R. E., "One-Dimensional Ablation Using Landau Transformation and Finite Control Volume Procedure," *Journal of Thermophysics and Heat Transfer*, Vol. 8, No. 2, April-June 1994, pp. 282–287.

[12] Martin, A. and Boyd, I. D., "Simulation of pyrolysis gas within a thermal protection system," *AIAA 40th Thermophysics Conference*, No. AIAA-2008-3805, Seattle, WA, 2008.

[13] Martin, A. and Boyd, I. D., "Variation of the Thomas algorithm for opposed-border tridiagonal systems of linear equations," *Communications in Numerical Methods in Engineering*, Vol. Accepted for publication, 2008.

[14] Zhang, H., Reggio, M., Trépanier, J.-Y., and Camarero, R., "Discrete form of the GCL for moving meshes and its implementation in CFD schemes," *Computers & Fluids*, Vol. 22, No. 1, 1993, pp. 9–23.

[15] Kuntz, D., Hassan, B., and Potter, D., "Predictions of Ablating Hypersonic Vehicles Using an Iterative Coupled Fluid/Thermal Approach," *Journal of Thermophysics and Heat Transfer*, Vol. 15, No. 2, 2001, pp. 129–139.

[16] Cohen, C. B. and Reshotko, E., "Similar Solutions for the Compressible Laminar Boundary Layer with Heat Transfer and Pressure Gradient," Technical Note 3325, NACA, 1955.

[17] Thompson, R. A. and Gnoffo, P. A., "Implementation of a Blowing Boundary Condition in the LAURA Code," *46th AIAA Aerospace Sciences Meeting and Exhibit*, AIAA-2008-1243, Reno, NV, Jan. 7-10 2008.

[18] Powars, C. A. and Kendall, R. M., *Aerotherm Chemical Equilibrium (ACE) Computer Program - User's Manual*, Aerotherm Corporation, Mountain View, California, May 1969.