1 Introduction

This document outlines an approach for a conservative remap of mass, linear momentum and internal energy. The description follows closely that of Jaio and Heath [1] and Farrell and Maddison [2] where a conservative interpolation is achieved using a Galerkin projection approach. Details of the algorithms described in those manuscripts differ mostly in the way element intersections are found but the mathematical foundations are the same. Another implementation of mesh-mesh intersections is the library `r3d` [3] and supports general polyhedral intersections. `r3d` is being used at Los Alamos for several remap code projects.

Begin by noting the variables that need to be remapped in ALEXA. These include internal energy, e, velocity, v and density ρ. In the current formulation both internal energy and density are element-centered variables while velocity is node-centered. These require slightly different treatments as outlined below. Note that as more physics are included in ALEXA additional element centered data will need to be remapped/transferred. It is anticipated that the only nodal quantity needing remap is velocity (in the guise of momentum transfer). These centerings assume a staggered arrangement of kinematic and thermodynamic data which is in line with the current element choice. Other element types may change data centerings but a node and element centered remap sufficient for now.

Finally, note that the element for magnetics will require edge and face variable remap. This is an area of on-going research and beyond the scope of the current document.

2 Algorithm

Following [1, 2] define a region of space with a *donor* field, p, and a *target* field, g and note that the regions are assumed to share a boundary. A weighted-residual formulation can be used to minimize the L_2 norm of the error between two fields,

$$\int_\Omega wgdV = \int_\Omega wpdV \qquad \forall w. \tag{1}$$

In the approximate space it is assumed that donor region is approximated with a mesh, $\mathcal{T}_D$, discretized with $\mathcal{N}_D$ basis functions ϕ_I, such that the discrete representation of the donor field is,

$$p^h(\mathbf{x}) = \sum_I^{\mathcal{N}_D} p_I \phi_I(\mathbf{x}). \tag{2}$$

The target region mesh, $\mathcal{T}_T$, is defined similarly so that a field on it may be discretized as,

$$g^h(\mathbf{x}) = \sum_I^{\mathcal{N}_T} g_I \varphi_I(\mathbf{x}) \tag{3}$$

where $\mathcal{N}_T$ are the number of bases, φ_I, in $\mathcal{T}_T$. As noted above the donor and target regions are assumed to share the same boundary and an identical discretization of this boundary for donor and target meshes.

The discrete form of Eqn. (1) can be formed by approximating the weighting function with the target fields basis such that,

$$\sum_J \int_\Omega \varphi_I \varphi_J dV g_J = \sum_K \int_\Omega \varphi_I \phi_K dV p_K. \tag{4}$$

or,

$$\mathbf{Mg} = \mathbf{f} \tag{5}$$

where,

$$\mathbf{M} = \int_\Omega \boldsymbol{\varphi}\boldsymbol{\varphi}^t dV \tag{6}$$

$$\mathbf{f} = \int_\Omega \boldsymbol{\varphi}\boldsymbol{\phi}^t \mathbf{p} dV \tag{7}$$

which is easily solved for the vector of target mesh field values, $\mathbf{g}$, as the mass matrix, $\mathbf{M}$, is symmetric positive definite. Note that when integrated properly (more on this later), Eqn. (4), conserves (globally) the volume integrated p (cf. [2, 1, 4]). As an aside, both the consistent mass matrix, $\mathbf{M}$, above and the lumped (diagonalized) matrix, $\mathbf{M}_L$, yield conservative results (cf. [2]) with $\mathbf{M}_L$ being simpler to invert but more diffusive. A convex combination of these might be used to produce something akin to a FCT (flux corrected transport) algorithm to provide necessary field variable limiting [5], but this is beyond the scope of this report.

3 Approximations

Eqns. (6—7) require some form of numerical integration. Quadrature of the mass matrix, $\mathbf{M}$, presents no special challenges and can be evaluated as an integral over the target mesh region element-by-element as,

$$\mathbf{M}_e = \int_{\Omega_e} \mathbf{N}\mathbf{N}^t dV \tag{8}$$

$$\tag{9}$$

where N is the restriction of the global basis functions φ to the target mesh element, e. The global element matrix is then assembled in the usual way $\mathbf{M} = \mathcal{A}_e \mathbf{M}_e$.

Evaluation of f (and in particular its integration) requires considerably more care to preserve conservation and solution accuracy. Several approaches to evaluating the integral associated with the forcing vector have been assessed in the literature and include i) donor-mesh integration, ii) target-mesh integration and iii) common-mesh-refinement (also called supermesh approach). These approaches are outlined as follows.

Donor-mesh discretization evaluates f on the donor mesh, $\mathcal{T}_D$ while the mass matrix is integrated as indicated above on the target-mesh, $\mathcal{T}_T$ [6]. This scheme is sensitive to relative mesh resolution, is zero-th order accurate if the meshes are refined simultaneously (error does not converge with mesh refinement) and can cause high-frequency oscillations [1].

Target-mesh discretization evaluates f by integrating Eqn. (7) on the target mesh and is one of the older approaches [7]. The advantage of this method is that it is straight-forward to implement, is exact for linear functions and is less oscillatory than the source-based approach [1]. Unfortunately the method is not conservative and so not applicable here.

The common-mesh-refinement (see also, supermesh construction) approach was developed to properly integrate the system in Eqns. (6—7) in order to preserve the conservative properties of the continuous Galerkin Projection of Eqn. (1). As before, the mass matrix

can be properly integrated on the target mesh. In the common-mesh-refinement approach, however, the forcing vector, $\mathbf{f}$, is integrated over (a tessellation of) the polygons created by intersecting the target are donor meshes. This "common-mesh" is required to accurately integrate the product of the donor and target basis functions. More precisely, the donor basis functions are continuous on the donor mesh while they are only $\mathcal{C}_0$ continuous on the target mesh. The inverse is true for the target basis functions. However, on the intersected polygon both target and donor basis functions are continuous with their product being quadratic. Hence a quadrature rule of degree 2 integrates them exactly.

4 Field Data Transfer

Density, ρ, and internal energy, e, are constant over an element. Hence, transfer of these quantities may be performed using Eqns. (6—7) assuming φ and ϕ are one on the element and zero elsewhere. A key point is that both density and internal energy should be transferred per unit volume (rather than per unit mass). While ρ is already mass per unit volume we may use the equations directly. Transfer of internal energy, however, should be performed as the transfer of ρe. Internal energy per unit mass is recovered once ρe is computed on the target mesh by dividing by target ρ, element-by-element.

Velocity, v, is transferred as momentum once the density has been transferred. Specifically we compute the mass and forcing terms as,

$$\sum_J \int_\Omega \rho_T \varphi_I \varphi_J dV g_J = \sum_K \int_\Omega \rho_D \varphi_I \phi_K dV p_K. \tag{10}$$

where ρ_D and ρ_T are the donor and target densities, p_K is the (nodal) velocity on the donor mesh and g_J is the solution vector for the velocity on the target mesh. Obviously, the mass matrix and forcing vector now incorporate a density field as,

$$\mathbf{M} = \int_\Omega \rho_T \boldsymbol{\varphi}\boldsymbol{\varphi}^t dV \tag{11}$$

$$\mathbf{f} = \int_\Omega \rho_D \boldsymbol{\varphi}\boldsymbol{\phi}^t dV \boldsymbol{p} \tag{12}$$

As indicated earlier, $\mathbf{M}$ is integrated over the target mesh while $\mathbf{f}$ uses the common-mesh-refinement for quadrature.

5 Approaches to Limiting

The Galerkin projection approach is conservative when the forcing term is integrated properly as described above. Another issue, however, is that data transfer may result in oscillatory target mesh solution when the source and target bases are of higher-order than element-wise constant. This can be particularly harmful for fields that have inherent physical limits (e.g. $\rho \geq 0$) as there are no constraints introduced in the galerkin projection that can prevent this behavior. Oscillations are also dangerous as they can cause instabilities in non-linear problems (e.g. shock mechanics).

6 Higher-order Approximations

The accuracy of the transfer operation can be increased by increasing the order of approximation of the source field data. For example, if density is represented as a constant over the source mesh then the Galerkin projection will only be first-order accurate (also known as donor method). Higher-order accuracy can be achieve by constructing a patch of elements around the element of interest (home element) and generating a mean-preserving, least-squares-reconstruction of the field on this element. Note that these reconstructions are discontinuous at element faces. The Galerkin projection approach outlined above still applies but with a modified $\mathbf{f}$. Note that the mass matrix, $\mathbf{M}$, remains unchanged as the target field is unchanged from the standard approach. Limiting approaches described above are generally required as described in § 5.

References

[1] Xiangmin Jiao and Michael T. Heath. Common-refinement-based data transfer between non-matching meshes in multiphysics simulations. *International Journal for Numerical Methods in Engineering*, 61:2402—2427, 2004.

[2] P.E. Farrell and J.R. Maddison. Conservative interpolation between volume meshes by local galerkin projection. *Computer Methods in Applied Mechanics and Engineering*, 201:89–100, 2011.

[3] Devon Powell. r3d software: Fast, robust polyhedral intersections, analytic integration, and conservative voxelization. `https://github.com/devonmpowell/r3d`.

[4] F. Alauzet. A parallel matrix-free conservative solution interpolation on unstructured tetrahedral meshes. *Computer Methods in Applied Mechanics and Engineering*, 299:116—142, 2016.

[5] Pavel Bochev, Denis Ridzal, and Kara Peterson. Optimization-based remap and transport: A divide and conquer strategy for feature-preserving discretizations. *Journal of Computational Physics*, 257:1113—1139, 2014.

[6] J.R. Cerbal and R. Lohner. Conservative load projection and tracking for fluid-structure problems. *AIAA Journal*, 35:687—692, 1997.

[7] N. Maman and C. Farhat. Matching fluid and structure meshes for aeroelastic compuations: A parallel approach. *Computers and Structures*, 54:779—785, 1995.