49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition
4 - 7 January 2011, Orlando, Florida

AIAA 2011-651

A Conservative Meshless Scheme: General Order Formulation and Application to Euler Equations

Edmond Kwan-yu Chiu* Qiqi Wang† and Antony Jameson‡
Stanford University, Stanford, CA 94305, USA
Massachusetts Institute of Technology, Cambridge, MA 02139, USA

In this paper, we present a mesh-free, finite-volume-like scheme designed for numerically solving conservation laws. We first derive a conservative formulation for computing mesh-free first derivatives based on a set of unknown coefficients satisfying polynomial consistency and other conditions. This formulation leads to a generalization that allows flexible choices of flux schemes while remaining locally conservative. We present an algorithm, based on minimum-norm solutions, for calculating unique meshless coefficients given a point distribution and connectivity. Numerical examples of solving the 2D Euler equations demonstrate the flexibility and practicality of the conservative meshless scheme.

I. Introduction

In the past two decades, many have developed various meshless algorithms to circumvent mesh generation. The strategies behind these algorithms range from applying variational formulations in mesh-free settings to employing point collocation strategies for approximating quantities of interest. In the former category, Nayroles et al.[1] developed the Diffuse Element (DE) method. Belytschko et al.[2] extended DE to Element Free Galerkin (EFG) method. Both methods required a background grid to compute required integrals in the variation formulation. Duarte and Oden[3] then provided an *h-p* meshless method in a more general partion-of-unity framework. Oñate et al.[4] proposed the Finite Point Method (FPM), which was also later used by Löhner et al.[5] Although its derivation contains some flavor of finite element methods, FPM uses point collocation in their final discretization to avoid the computation of integrals involving test functions. Batina[6] had also previously arrived at a very simular formulation using polynomial basis from a different perspective. In other work involving point collocation techniques, Kansa[7,8], and later Shu et al.[9] and Tota and Wang,[10] applied radial basis functions in their meshless algorithms. Sridar and Balakrishan,[11] and Katz and Jameson[12] also developed meshless methods that resemble traditional finite difference methods by using Taylor series.

The work mentioned above, and much more others, demonstrate the great potential in applying meshless methods to tackling complex problems in scientific computing. However, to the best of our knowledge, all meshless methods documented in the literature face a fundamental challenge of the lack of formal conservation. Because of the local nature of these schemes, they do not preserve conservation at the discrete level except in very limited situations. Compared to mesh-based approaches, the lack of conservation hinders computation speed as one cannot compute reciprocal fluxes as efficiently as in edge-based approaches. More importantly, non-conservation leads to unpredictable errors when sharp discontinuities exist in the solution. The difficulty in quantifying the effects of non-conservation on accuracy and stability of algorithms have often raised doubts about meshless methods in the scientific community.

In this paper, we report a novel mesh-free scheme that not only possesses various conservation and mimetic properties at the discrete level, but also provide the option for one to incorporate existing flux

*Ph. D. Candidate, Department of Aeronautics and Astronautics, William F. Durand Building, 496 Lomita Mall, Stanford, CA 94305-4035; AIAA Member
†Assitant Professor, Department of Aeronautics and Astronautics, Room 33-408, Massachusetts Institute of Technology, Cambridge, MA 02139; AIAA Member
‡Thomas V. Jones Professor of Engineering, Department of Aeronautics and Astronautics, William F. Durand Building, 496 Lomita Mall, Stanford, CA 94305-4035; AIAA Fellow

schemes when solving conservation laws. Accordingly, the rest of the paper is organized as follows: We first define the discrete derivative operator for our meshless scheme in section II, along with the reciprocity and consistency conditions we impose. Based on those conditions, we prove in sections III and IV, respectively, the resulting global and local conservation properties of the scheme. These properties lead to analogs of finite volume schemes and a generalization, also presented in section IV, that allows flexible flux schemes. Section V outlines the scheme's discrete geometric properties, which contributes to the design of the procedure in section VI for generating the meshless coefficients. Section VII contains results from testing the coefficient generation procedure on airfoil geometries and using the generalized meshless scheme to numerically solve the 2D Euler equations in both subsonic and transonic regimes. Finally, section VIII concludes the paper with a few remarks and highlights of future work related to the current meshless scheme.

II. Formulation

Consider a domain Ω filled with a set of n_p points, with coordinates $\vec{x}_i$, $i = 1, \ldots, n_p$, both in the interior of the domain and on the domain boundary $\partial\Omega$. Given a scalar u, we define its discrete derivative $\delta^k u_i$ at point i by

$$m_i \delta^k u_i = a^k_{ii} u_i + \sum_{j \in s_i} a^k_{ij} u_j \ , \tag{1}$$

where $k = I, II, III$ indexes the spatial dimension, s_i is the set of neighboring points around i (does not include point i itself), and m_i and a^k_{ij} are coeffcients to be determined. For consistent dimensions of the equation, one can consider m_i to represent a virtual volume associated with each point, while the coefficients a^k_{ij} for the point pair (i, j) have corresponding dimensions of area (this characterization is justified in section IV).

In addition, we enforce the following conditions on a^k_{ij} and m_i:

C-1. *Reciprocity and boundary conditions*:

$$\begin{aligned} a^k_{ij} &= -a^k_{ji} \ , \quad i \neq j, \qquad j \in s_i \Leftrightarrow i \in s_j \ , \\ a^k_{ii} &= 0 \ , \qquad i \in \Omega \ , \\ a^k_{ii} &= \frac{1}{2} n^k_i \ , \quad i \in \partial\Omega \ , \end{aligned}$$

where n^k_i is the k-th component of the area-weighted outward facing boundary normal at the boundary point i.

C-2. *Consistency of order L*:

$$a^k_{ii} \phi(\vec{x}_i) + \sum_{j \in s_i} a^k_{ij} \phi(\vec{x}_j) = m_i \partial^k \phi(\vec{x}_i)$$

for all multivariate polynomials ϕ of total order L, where $\vec{x}_i$ is the vector coordinates of point i.

We shall now investigate the properties of discretizations satisfying Conditions C-1 and C-2.

III. Global Conservation

We shall first show that a discrete differentiation operator satisfying the conditions in section II also preserves global conservation discretely.

Theorem 1 (Discrete Conservation)**.** *If m_i and a^k_{ij} satisfy conditions C-1 and C-2, then*

$$\sum_{i=1}^{n_p} m_i \delta^k u_i = \sum_{i \in s_B} u_i n^k_i \ , \tag{2}$$

where s_B is the collection of all boundary points.

Proof. We can write the discrete first derivative as

$$m_i \delta^k u_i = \sum_{j=1}^{n_p} \tilde{a}_{ij}^k u_j \ , \tag{3}$$

where $\tilde{a}_{ii}^k = a_{ii}^k$, $\tilde{a}_{ij}^k = a_{ij}^k$ if $j \in s_i$, and $\tilde{a}_{ij}^k = 0$ otherwise. Let $\phi \equiv 1$ in C-2, and use $a_{ij}^k + a_{ji}^k = 0$ from C-1, we get

$$\tilde{a}_{ij}^k = -\tilde{a}_{ji}^k \qquad (\textit{only if } i \neq j), \tag{4}$$

$$\sum_{\substack{j=1 \\ j \neq i}}^{n_p} \tilde{a}_{ji}^k = a_{ii}^k, \quad \text{or} \quad \sum_{j=1}^{n_p} \tilde{a}_{ji}^k = 2a_{ii}^k \ . \tag{5}$$

Incorporating equation (3) into the left hand side of equation (2) and using equation (4), we have

$$\sum_{i=1}^{n_p} m_i \delta^k u_i = \sum_{i=1}^{n_p} \sum_{j=1}^{n_p} \tilde{a}_{ij}^k u_j = \sum_{j=1}^{n_p} (\sum_{i=1}^{n_p} \tilde{a}_{ij}^k) u_j = \sum_{j=1}^{n_p} 2a_{jj}^k u_j = \sum_{i=s_B} n_i^k u_i \ ,$$

where we changed the dummy index from j to i in the last step. □

The equality (2) is just the discrete analog of global conservation

$$\int_\Omega \partial^k u dx = \int_{\partial\Omega} u n^k ds \ .$$

In addition to conservation, the meshless operator also possesses certain discrete mimetic properties, namely summations by parts and energy conservation.

Theorem 2 (Summation by parts)**.** *If m_i and a_{ij}^k satisfy conditions C-1 and C-2, then*

$$\sum_{i=1}^{n_p} m_i v_i \delta^k u_i + \sum_{i=1}^{n_p} m_i u_i \delta^k v_i = \sum_{i \in s_B} v_i u_i n_i^k \ . \tag{6}$$

Corollary 3 (Discrete energy conservation)**.** *If m_i and a_{ij}^k satisfy conditions C-1 and C-2, then*

$$\sum_{i=1}^{n_p} m_i u_i \delta^k u_i = \sum_{i \in s_B} \frac{1}{2} u_i^2 n_i^k \ . \tag{7}$$

One can see that equations (6) and (7) represent respective discrete versions of integration by parts

$$\int_\Omega v \partial^k u dx = \int_{\partial\Omega} v u n^k ds - \int_\Omega u \partial^k v dx$$

and energy conservation

$$\int_\Omega u \partial^k u dx = \int_{\partial\Omega} \frac{1}{2} u^2 n^k ds \ .$$

Appendix A contain the omitted proofs for theorem 2 and corollary 3.

IV. Local Conservation and Generalization for Arbitrary Flux

In this section, we demonstrate that the current meshless scheme possesses a discrete analog of local conservation

$$\int_{\omega_i} \partial^k u dx = \int_{\partial\omega_i} u n^k ds$$

and use this property to obtain a generalized framework that accommodates a large class of flux schemes.

Theorem 4 (Local Discrete Conservation)**.** *If m_i and a_{ij}^k satisfy conditions C-1 and C-2, then, defining $n_{f_j}^k = 2a_{ij}^k$ to be virtual face area vectors for the corresponding interior point pair (i,j), the following conditions hold:*

1. *At each point i,*

$$m_i \delta^k u_i = \begin{cases} \sum_{j \in s_i} u_{f_j} n_{f_j}^k + u_i n_i^k & i \in s_B \\ \sum_{j \in s_i} u_{f_j} n_{f_j}^k & i \notin s_B \ , \end{cases} \tag{8}$$

   where u_{f_j} represents an interface value of u.

2. *In addition, each m_i represents a virtual volume associated with point i that is enclosed by some virtual faces f_j associated with the point pair (i,j) (plus boundary faces if $i \in s_B$). Each face f_j has a corresponding area of $\vec{n}_{f_j}$ and an interface function value of u_{f_j}.*

To faciliate the proof, we introduce the following corollary.

Corollary 5 (Local Discrete Geometric Conservation Law)**.** *If m_i and a_{ij}^k satisfies conditions C-1 and C-2, and the vector valued multivariate function $\vec{\phi}$ satisfies the divergence free condition*

$$\nabla \cdot \vec{\phi} = 0 \ ,$$

then the following condition holds:

$$\begin{aligned} \vec{\phi}_i \cdot \vec{n_i} + \sum_{j \in s_i} \vec{\phi}_{f_j} \cdot \vec{n}_{f_j} = 0 \qquad & i \in s_B \\ \sum_{j \in s_i} \vec{\phi}_{f_j} \cdot \vec{n}_{f_j} = 0 \qquad & i \notin s_B \ . \end{aligned} \tag{9}$$

We shall proof theorem (4) and corollary (5) together.

Proof. Applying condition C-2 to $\phi \equiv 1$ leads to $a_{ii}^k + \sum_{j \in s_i} a_{ij}^k = 0$. Multiplying this by u_i and adding the result to the definition of the first derivative operator (1), we have

$$\begin{aligned} m_i \delta^k u_i &= a_{ii}^k u_i + \sum_{j \in s_i} a_{ij}^k u_j + a_{ii}^k u_i + \sum_{j \in s_i} a_{ij}^k u_i \\ &= 2a_{ii}^k u_i + \sum_{j \in s_i} a_{ij}^k (u_i + u_j) \\ &= 2a_{ii}^k u_i + \sum_{j \in s_i} n_{ij}^k \frac{(u_i + u_j)}{2} \ . \end{aligned} \tag{10}$$

For interior points, $a_{ii}^k = 0$. For boundary points, $a_{ii}^k = \frac{1}{2} {n_i}^k$. Thus, we obtain equation (8) with $u_{f_j} = \frac{1}{2}(u_i + u_j)$.

To prove corollary 5, let $u = \phi$ be a polynomial of total order L. Consistency of order L gives

$$m_i \partial^k \phi_i = \begin{cases} \sum_{j \in s_i} \phi_{f_j} n_{f_j}^k + \phi_i n_i^k & i \in s_B \\ \sum_{j \in s_i} \phi_{f_j} n_{f_j}^k & i \notin s_B \ . \end{cases} \tag{11}$$

If $\vec{\phi}$ is divergence free, summing over equation (11) applied to each component of $\vec{\phi}$ results in equation (9).

To complete the proof of theorem 4, it remains to show that the coefficients m_i and a_{ij}^k are consistent and does not lead to numerical sources.

First, let $u = x^k$ (recall k indexes the spatial dimension). Equation (8) becomes

$$m_i = \begin{cases} \sum_{j \in s_i} n^k_{f_j} \dfrac{x^k_i + x^k_j}{2} & i \notin s_B \\ \sum_{j \in s_i} n^k_{f_j} \dfrac{x^k_i + x^k_j}{2} + n^k_i x^k_i & i \in s_B \ . \end{cases} \tag{12}$$

Then, corollary 5 applied to

$$\phi^{\hat{k}} = \begin{cases} 1 & \hat{k} = k \\ 0 & \hat{k} \neq k \end{cases}$$

yields $\sum_{j \in s_i} n^k_{f_j} = 0$ for an interior point i and $\sum_{j \in s_i} n^k_{f_j} + n^k_i = 0$ for a boundary point. These conditions guarantee that a volume of size m_i is fully enclosed by its interfaces, which include boundary elements if appropriate. Thus, no numerical sources arise from inconsistent definition of virtual faces and volumes. The scheme is locally conservative. □

Theorem 4 justifies the geometric interpretation of the coefficients a^k_{ij} and m_i as analogs of face areas and volumes. This geometric interpretation also allow us to generalize our scheme. A set of a^k_{ij}, n^k_i and m_i (such as one generated by the algorithm in Section VI) can still respresent interface faces, boundary faces, and volumes. However, with the above geometric information, we can replace the interface flux $u_f = \frac{u_i + u_j}{2}$ with more sophisticated fluxes of choice, while preserving conservation, by generalizing equation (10) as

$$m_i \delta^k_f u_i = 2 a^k_{ii} u_i + \sum_{j \in s_i} 2 a^k_{ij} F^k_{ij} \ , \tag{13}$$

where, for example, the interface flux F_{ij} can be a function of u_i, u_j and the derivative of u at points i and j. This general formulation also preserves the global conservation property of the original scheme (1).

In section VII, one can see the results from using this approach to successfully tackle transonic flow problems. In particular, we demonstrate the potential to apply the generalized scheme (13) with a wide variety of existing flux schemes by incorporating into it a numerical flux formulation originally developed for stabilizing finite volume discretizations.

V. Additional Global Properties

Before we present the algorithm for computing a^k_{ij} and m_i in section VI, we discuss some extra global properties of our scheme. Resulting from conditions C-1 and C-2, these properties parallel equation (11) and corollary 5. They provide important insight into constructing the coefficient generation algorithm.

Theorem 6 (Discrete divergence theorem)**.** *If a^k_{ij} and m_i satisfy conditions C-1 and C-2, then the following condition holds for all multivariate polynomials ϕ of total order $2L$:*

$$\sum_{i \in s_B} \phi(\vec{x}_i) n^k_i = \sum_{i=1}^{n_p} m_i \partial^k p(\vec{x}_i) \ . \tag{14}$$

Proof. It is sufficient to prove (14) for all multivariate monomials L of order less than or equal to $2L$. Let $\phi = \phi_1 \phi_2$, where both ϕ_1 and ϕ_2 are monomials or order less than or equal to L, and thus satisfy condition C-2:

$$a^k_{ii} \phi_1(\vec{x}_i) + \sum_{j \in s_i} a^k_{ij} \phi_1(\vec{x}_j) = m_i \partial^k \phi_1(\vec{x}_i) \tag{15}$$

$$a^k_{ii} \phi_2(\vec{x}_i) + \sum_{j \in s_i} a^k_{ij} \phi_2(\vec{x}_j) = m_i \partial^k \phi_2(\vec{x}_i) \ . \tag{16}$$

We first multiply equation (15) by $\phi_2(\vec{x}_i)$ and (16) by $\phi_1(\vec{x}_i)$. Then, adding the results together and summing over $i = 1, \ldots, n_p$, using the fact (derived from $a^k_{ij} + a^k_{ji} = 0$ in condition C-1) that

$$\sum_{(i,j) \in E} a^k_{ij} \big(\phi_1(\vec{x}_i) \phi_2(\vec{x}_j) + \phi_1(\vec{x}_j) \phi_2(\vec{x}_i) \big) = 0 \ ,$$

where E is the set of all neighborhood pairs in the domain, we get

$$\sum_{i=1}^{n_p} 2a_{ii}^k \phi_1(\vec{x}_i)\phi_2(\vec{x}_i) = \sum_{i=1}^{n_p} m_i \partial^k \big(\phi_1(\vec{x}_i)\phi_2(\vec{x}_i)\big) \ .$$

Using the formula for a_{ii}^k in condition C-1, we obtain equation (14) for $\phi = \phi_1\phi_2$. □

In this proof, we obtained equation (14) by taking linear combinations of the constraints in condition C-2 and canceling out all a_{ij}'s in the process. Therefore, this theorem shows that conditions C-1 and C-2, as linear constraints for a_{ij}, are **linearly dependent**. In other words, these conditions can not be satisfied simultaneously unless m_i satisfies equation (14).

The following corollary shows that equation (14) as linear constraints for m_i are also linearly dependent. A solution for (14) can only exist if the boundary faces n_i^k are chosen appropriately.

Corollary 7 (Geometric conservation law)**.** *If a_{ij}^k and m_i satisfy conditions C-1 and C-2, and the vector-valued multivariate polynomials $\vec{\phi}$ of order $2L$ satisfies the divergence free condition*

$$\nabla \cdot \vec{\phi} = 0 \ ,$$

then the following condition holds:

$$\sum_{i \in s_B} \vec{\phi}(\vec{x}_i) \cdot \vec{n}_i = 0 \ . \tag{17}$$

Proof. Summing equation (14) over k and using the divergence free condition, we obtain quation (17). □

VI. Construction of Meshless Coefficients

As we saw in the previous section, theorem 6 and corollary 7 means that n_i^k must be first chosen to satisfy equation (17). Here, we describe an algorithm for obtaining such n_i^k's and, afterwards, computing the coefficients $a_{ij}^k$and m_i that satisfies conditions C-1 and C-2. The steps are as follows:

1. Calculate estimates of $\vec{n}_i$ for all boundary points based on the geometry of the domain boundary.
2. Project the estimates of $\vec{n}_i$ into the linear subspace that satisfies (17).
3. Solve a quadratic program for $\vec{a}_{ij}$ and m_i to enforce C-1 and C-2 while minimizing

$$\sum_{(i,j)\in E} \|\vec{a}_{ij}\|_2^2 \ ,$$

   where E is the set of all neighborhood pairs $\{(i,j) \mid j \in s_i\}$.

The projection in step 2 involves a small number of constraints when L is small. We carry out this projection using QR decomposition of the matrices involved. Specifically, let $\boldsymbol{n}$ be a column vector that contains $\vec{n}_i$ for all boundary points $i \in s_B$. We can write the geometric conservation law (17) in matrix form as

$$G^{\mathrm{T}}\boldsymbol{n} = 0 \ . \tag{18}$$

The number of columns of the matrix G is equal to the number of linearly independent vector-valued, **divergence-free** multivariate polynomials of order up to $2L$. Each column of G contains the values of one of these polynomials at all boundary points. When L is small, G is a thin matrix.

To ensure that the total volume enclosed by the boundary faces does not change during the projection process, we also enforce the constraint

$$\frac{1}{n_d} \sum_{i \in s_B} \vec{x}_i \cdot \vec{n}_i = m_0 \tag{19}$$

for each closed boundary of the domain, where n_d is the number of spatial dimensions and

$$m_0 = \frac{1}{n_d} \sum_{i \in s_B} \vec{x}_i \cdot \vec{n0}_i \ .$$

Let $\boldsymbol{n}_0$ be the initial estimate of $\boldsymbol{n}$ based on the geometry of the domain boundary, we can calculate $\boldsymbol{n}$ by

$$\begin{aligned} R^T \boldsymbol{y} &= \boldsymbol{g} - G^T \boldsymbol{n}_0 \\ \Delta \boldsymbol{n} &= Q\boldsymbol{y} \\ \boldsymbol{n} &= \boldsymbol{n}_0 + \Delta \boldsymbol{n} \ , \end{aligned} \tag{20}$$

where $\boldsymbol{g} = (0, \ldots, 0, m_0)^{\mathrm{T}}$ and $QR = G$ is the (thin) QR decomposition of G. The projected $\boldsymbol{n}$ satisfies the linear equation (18), which is equivalent to the geometric conservation law (17).

For step 3, we denote $\boldsymbol{a}$ as a column vector that contains $\vec{a}_{ij}$ for all neighborhood pairs, and $\boldsymbol{m}$ as the vector containing m_i for all points. For each neighborhood pair (i, j), we store either $\vec{a}_{ij}$ or $\vec{a}_{ji}$ such that the reciprocity condition in C-1 is automatically satisfied. We write condition C-2 in the linear form

$$\mathbf{C}^{\mathrm{T}} \boldsymbol{a} + \mathbf{D}^k \boldsymbol{m} = \boldsymbol{d}^k \ ,$$

where $\mathbf{C}^{\mathrm{T}} \boldsymbol{a}$ contains the term $\sum_{j \in s_i} a_{ij}^k \phi_j$, $\mathbf{D}^k \boldsymbol{m}$ contains the $m_i (\partial^k \phi)_i$ terms, and $\boldsymbol{d}^k$ contains boundary terms of the type $a_{ii}^k \phi_i$. We can rewrite the system as

$$\begin{bmatrix} \mathbf{C}^{\mathrm{T}} & \mathbf{0} & \mathbf{0} & \mathbf{D}^I \\ \mathbf{0} & \mathbf{C}^{\mathrm{T}} & \mathbf{0} & \mathbf{D}^{II} \\ \mathbf{0} & \mathbf{0} & \mathbf{C}^{\mathrm{T}} & \mathbf{D}^{III} \end{bmatrix} \begin{bmatrix} \boldsymbol{a}^I \\ \boldsymbol{a}^{II} \\ \boldsymbol{a}^{III} \\ \boldsymbol{m} \end{bmatrix} = \begin{bmatrix} \boldsymbol{d}^I \\ \boldsymbol{d}^{II} \\ \boldsymbol{d}^{III} \end{bmatrix} \tag{21}$$

In additiion, just like in finite volume schemes, we require $m_i > 0$. To enforce this positivity constraint, we set $m_i > m_{min}$, where m_{min} is a user-selected parameter, typically on the order of $\sqrt{\epsilon_{mac}}$ (ϵ_{mac} is the machine zero) to avoid the virtual volume at any location to be arbitrarily close to zero.

As long as enough neighbors per solution point exist, the system (21) remain underdetermined, and one can solve the resulting problem using QP or convex optimization tools, such as `CVX`[13,14] and CVXOPT.[15] Here, we use Saunder's PDCO[16], a software, written in MATLAB®, that solves convex optimization problems using interior point methods. We apply right preconditioning by scaling the columns of $\mathbf{C}^{\mathrm{T}}$ by $\|\vec{\Delta x}_{ij}\|_2$ when enforcing the constraints (and scaling the objective function accordingly).

Note that, although the constraint matrix does not have full row rank because of the discrete divergence theorem (14), we did not experience any problem of infeasibility during the solution procedure after constructing the right-hand side using a set of $\vec{n}_i$'s that satisfies equation (18).

VII. Application to Solving the Euler Equations

In this section, we present results from applying the generalized scheme to obtain steady-state numerical solutions to the 2D Euler equations. In the process, we also show the results from generating the required meshless coefficients using the algorithm in section VI.

The 2D Euler equations are

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y} = 0 \tag{22}$$

with

$$\mathbf{w} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{pmatrix}, \qquad \mathbf{f} = \begin{pmatrix} \rho u \\ \rho u^2 + P \\ \rho uv \\ \rho uH \end{pmatrix}, \qquad \mathbf{g} = \begin{pmatrix} \rho v \\ \rho vu \\ \rho v^2 + P \\ \rho vH \end{pmatrix}$$

and

$$E = \frac{P}{(\gamma - 1)\rho} + \frac{1}{2}(u^2 + v^2), \quad H = E + \frac{P}{\rho},$$

where ρ, u, v, P, E, and H are, respectively, the density, x- and y-components of the flow velocity, pressure, total energy and total enthalpy.

VII.A. Discretization

As mentioned in section IV, we use the generalized form of our scheme as shown in equation (13). We compute the flux divergence as

$$m_i(\delta_f \cdot \mathbf{F})_i = \sum_{j \in s_i} \tilde{\mathbf{f}}_{ij} \ , \tag{23}$$

where

$$\tilde{\mathbf{f}}_{ij} = \frac{1}{2}(\mathbf{f}_{ni} + \mathbf{f}_{nj}) - \mathbf{d}_{ij} \ , \tag{24}$$

is the the numerical flux at each connect edge that consists of the central flux and the diffusive flux $\mathbf{d}_{ij}$ for stability, and $\mathbf{f}_n$ denotes the area-weighted normal flux at the virtual interface between points (i, j). One can show that equations (23) and (24) are equivalent to (13) with F_{ij}^k containing components of $\mathbf{d}_{ij}$. To construct the diffusive flux, we employ the H-CUSP scheme[17], which admits constant enthalpy in steady flow, combined with symmetric limited positive (SLIP) reconstruction[17] to retain higher order of accuracy in regions with smooth solutions. Together, these schemes give

$$\mathbf{d}_{ij} = \frac{1}{2}\alpha^* c(\mathbf{w}_{hR} - \mathbf{w}_{hL}) + \frac{1}{2}\beta(\mathbf{f}_{nR} - \mathbf{f}_{nL}) \tag{25}$$

where $\mathbf{w}_h$ denotes the conservative variables with energy replaced by enthalpy, the subscripts "L" and "R" denote limited reconstructed left and right states on both sides of the edge, c represents the speed of sound at the virtual interface, and α^* and β denote computed coefficients[17] based on the average state.

For boundary conditions, we enforce flow tangency at solid walls and enforce constant stagnation enthalpy with vortex correction[18,19] for non-zero circulation.

We march the solution to a steady state using a modified Runge-Kutta scheme[20] designed for an enlarged stability region. Given the semi-discrete set of equations

$$\frac{\partial \mathbf{w}_i}{\partial t} + \mathbf{R}_i = 0 \ , \tag{26}$$

where the total residual

$$\mathbf{R}_i = \frac{1}{m_i}(\mathbf{Q}_i - \mathbf{D}_i) \ , \tag{27}$$

is separated into the convective part $\mathbf{Q}$ and diffusive part $\mathbf{D}$. For the current spatial discretization,

$$\mathbf{Q}_i = \sum_{j \in s_i} \frac{1}{2}(\mathbf{f}_{nj} + \mathbf{f}_{ni}), \quad \mathbf{D}_i = \sum_k \sum_{j \in s_i} \mathbf{d}_{ij} \tag{28}$$

For a scheme with q stages, we have

$$\begin{aligned}
\mathbf{w}^{(n+1,0)} &= \mathbf{w}^n \\
&\vdots \\
\mathbf{w}^{(n+1,s)} &= \mathbf{w}^n - \alpha_s \Delta t \left(\mathbf{Q}^{(s-1)} + \mathbf{D}^{(s-1)} \right) \\
&\vdots \\
\mathbf{w}^{(n+1)} &= \mathbf{w}^{(n+1,q)} \ ,
\end{aligned}$$

where

$$\begin{aligned}
\mathbf{Q}^{(0)} &= \mathbf{Q}(\mathbf{w}^n), & \mathbf{D}^{(0)} &= \mathbf{D}(\mathbf{w}^n) \\
&\vdots & &\vdots \\
\mathbf{Q}^{(s)} &= \mathbf{Q}(\mathbf{w}^{(n+1,s)}), & \mathbf{D}^{(s)} &= \beta_s \mathbf{D}(\mathbf{w}^{(n+1,s)}) + (1 - \beta_s)\mathbf{D}^{(s-1)} \ ,
\end{aligned}$$

and α_s and β_s are the stage coefficients. We use a 5-stage scheme that Mavriplis[21] and many others have reported success with the following coefficients.

$$\vec{\alpha}_s = (\frac{1}{4}, \frac{1}{6}, \frac{3}{8}, \frac{1}{2}, 1), \qquad \vec{\beta}_s = (1, 0, 0.56, 0, 0.44)$$

To accelerate convergence, we also apply techniques such as local time stepping, residual smoothing[22] and enthalpy damping[23] on the total residual.

VII.B. Test Geometries, Connectivities and Meshless Coefficients

To most directly compare results among different spatial schemes, we use the same geometries and point clouds as those in Katz and Jameson[24]. To obtain valid surface face areas for all boundary points, we remove any points at geometric singularities (e.g. trailing edge) and connect each neighbor point of the boundary singularity to the closest boundary point without intersecting any part of the geometry.

Then, we construct the 2D version of system (21) for linear consistency ($L = 1$). Following the procedure in section VI, it takes less than twenty seconds for PDCO to obtain a set of meshless coefficients, even for the finest point distributions, on a workstation with Intel Xeon 5160 processors in a quad-core configuration. Moreover, the coefficients satisfy nodal polynomial constraints to order 10^{-6} or better in all cases.

Figure 1 shows scatter plots of the virtual volumes and virtual face area magnitudes corresponding to the finest point cloud around the RAE 2822 airfoil used in some of the test cases. Interestingly, by enforcing linear consistency, we obtain meshless coefficients that adjust to local point spacings in the point cloud, i.e. the virtual volumes and face areas are smaller at locations which the point cloud is denser, and vice versa. The meshless coefficients behave similarly in all of the test geometries. It is very encouraging that the algorithm in section VI produce solutions, unique for each point cloud, that make sense physically.

VII.C. Compressible Flow Results

The selected test problems are classical subsonic and transonic benchmark aerodynamics problems listed in AGARD documents[25,26] and investigated by many researchers using various numerical algorithms. It is very natural to validate the current meshless framework on these test problems. In particular, Katz and Jameson[24] used some of the test problems for comparing a few strong-form-based meshless schemes against a finite volume discretization on structured grids.[23] Besides comparing the current results to their finite volume counterparts, we shall also compare the current results to those obtained using different meshless algorithms, presented in Katz and Jameson[24,27].

Theoretically, inviscid isentropic flow around objects has zero drag. We first look at some shock-free problems to confirm that the computed drag does converge to zero with increasingly refined point distributions.

Table 1 lists drag convergence results for the test cases of flow over the RAE 2822 and KORN airfoils. Since the point distributions were not regular, the tables also list the total number of points. From figure 2, one can see the asymptotic second-order convergence, even thought the scheme is only formally first-order accurate.

(a) RAE 2822, $M = 0.5$, $\alpha = 3^\circ$

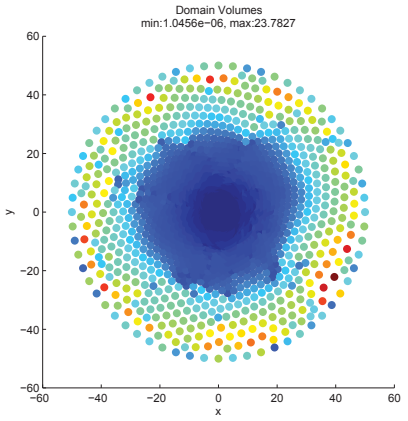
n_s	n_p	c_d
40	456	0.0056
80	1579	0.0013
160	5732	0.0002

(b) KORN, $M = 0.75$, $\alpha = 0^\circ$

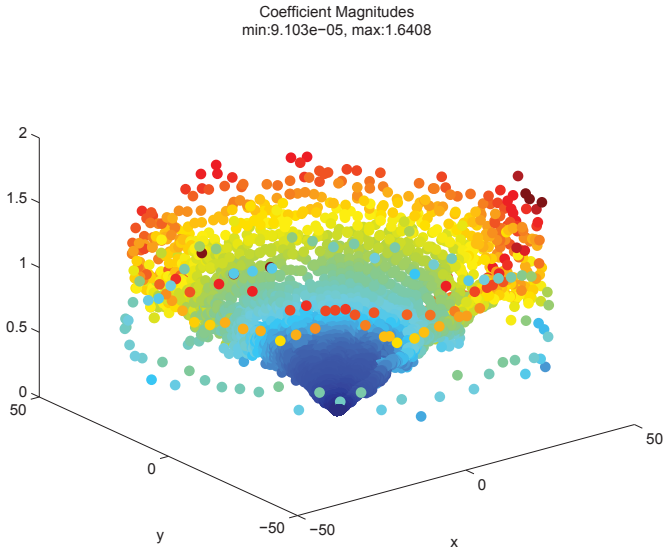
n_s	n_p	c_d
40	473	0.0063
80	1918	0.0019
160	7809	0.0003

Table 1: Inviscid drag convergence for subsonic flow over airfoils

Now, we present results from transonic test cases. Table 2 lists the lift and drag coefficients computed with the current scheme along with structured finite volume and other meshless results listed in Katz and Jameson[24,27]. In each test geometry, there are 160 points on the airfoil surface. Recall, from section IV, that the current generalized meshless framework allows us to implement the conservative meshless and finite volume discretizations in the same way using point locations and reciprocal coefficients at interfaces (as opposed to some finite volume implementations that compute face normals and other necessary quantities from a set of dual or face vertices and face edges). As a result, we also compare our meshless results



(a) Virtual volumes around RAE 2822



(b) Virtual face magnitudes around RAE 2822

Figure 1: Virtual volumes and face magnitudes around RAE 2822

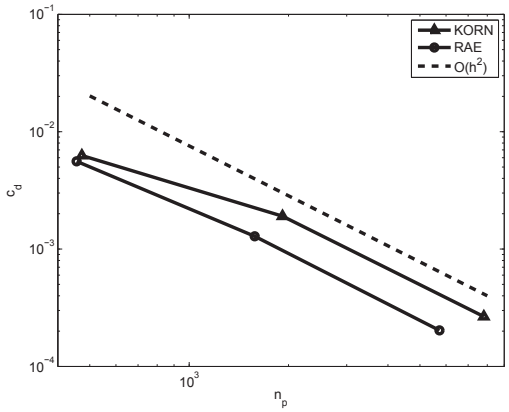


Figure 2: Inviscid drag convergence for subsonic flow over airfoils

with unstructured finite volume results obtained using the same software without any modifications. In the tables, "CM" represents the current results obtained using our conservative meshless framework. "FV-U" represents unstructured finite volume results obtained using the same software used for computing the conservative meshless results. "FV-S" represents structured finite volume results obtained using Jameson's FLO82. Other acronyms are consistent with those in Katz and Jameson[24, 27]. "MV" denotes the alignment-based "meshless volume" scheme by Katz and Jameson[27], "TLS" denotes a Taylor-series-based meshless scheme, "PLS" denotes a polynomial-based meshless scheme, similar to the Finite Point Method, and "RBF" denotes a meshless method based on radial basis functions.

As one can see, results from the conservative meshless scheme compares extremely well with those obtained using structured and unstructured finite volume methods and other meshless techniques. Figures 3 through 6 show the surface pressure coefficient plot (overplotted on FLO82 results) and glyph plot of the Mach number around the airfoil. The surface pressure coefficient plots show sharp capturing of shocks. Also, the shock-free results in the KORN airfoil case indicate high accuracy of the simulations. These results show excellent potential of the current scheme for wider applications.

VIII. Conclusion and Future Work

We addressed the long-standing issue of lack of formal conservation of meshless schemes by formulating a mesh free derivative operator that preserves both global and local conservation at the discrete level. Formal discrete conservation results from a set of reciprocity and polynomial consistency constraints. Furthermore, based on the local conservation properties of the scheme, we generalized the scheme to a finite-volume-like formulation that allows our scheme to be used with a broad range of existing flux schemes. We constructed an algorithm for computing the conservative meshless coefficients satisfying the desired constraints and other derived necessary conditions. For test geometries involving airfoils, the coefficient generation algorithm produced meshless coefficients with very little overhead. When used in computing steady inviscid subsonic and transonic flows over airfoils, the generated coefficients and the generalized formulation led to stable, convergent and accurate results that compare very well to structured and unstructured finite volume counterparts.

Given the encouraging results in this paper, we would like to test the current meshless scheme on problems of larger scale and complexity. Important topics of interest in this aspect include 3D implementation of the current scheme, parallelization and domain decomposition. We also plan on exploring possible improvements or variants to the meshless scheme itself, such as the possibility to further compactify the connectivity stencil, or improvements in the coefficient generation process.

(a) NACA 0012, $M = 0.8$, $\alpha = 1.25°$

	c_l	diff. (%)	c_d	diff. (%)
FV-S	0.3737	-	0.0237	-
FV-U	0.3639	2.6	0.0221	6.8
CM	0.3615	3.3	0.0226	4.6
MV	0.3713	0.6	0.0229	3.4
TLS	0.3616	3.2	0.0231	2.5
PLS	0.3684	1.4	0.0242	2.1
RBF	0.3986	6.7	0.0269	13.5

(b) NACA 0012, $M = 0.85$, $\alpha = 1.0°$

	c_l	diff. (%)	c_d	diff. (%)
FV-S	0.3891	-	0.0582	-
FV-U	0.3904	0.3	0.0559	4.0
CM	0.3895	0.1	0.0568	2.4
MV	0.3923	0.8	0.0572	1.7
TLS	0.3830	1.6	0.0565	2.9
PLS	0.3883	0.2	0.0593	1.9
RBF	0.3343	14.1	0.0570	2.1

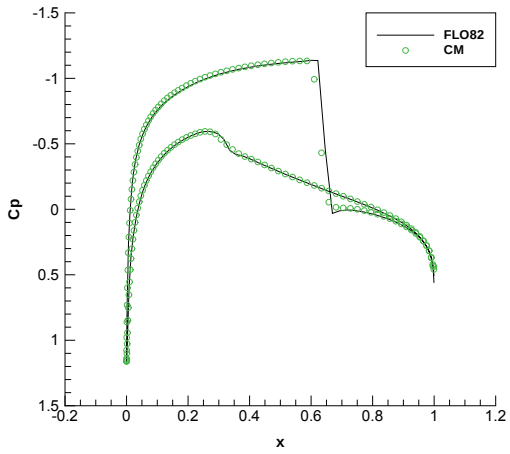
(c) RAE 2822, $M = 0.75$, $\alpha = 3.0°$

	c_l	diff. (%)	c_d	diff. (%)
FV-S	1.1481	-	0.0486	-
FV-U	1.1552	0.6	0.0475	2.3
CM	1.1589	0.9	0.0479	1.4
MV	1.1338	1.2	0.0474	2.5
TLS	1.1319	1.4	0.0481	1.0
PLS	1.1347	1.2	0.0491	1.0
RBF	1.1744	2.3	0.0545	12.1

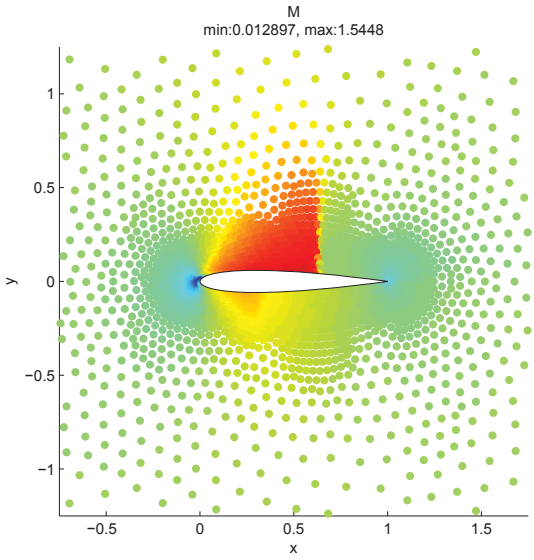
(d) KORN, $M = 0.75$, $\alpha = 0.0°$

	c_l	diff. (%)	c_d	diff. (%)
FV-S	0.6308	-	0.0000	-
FV-U	0.6379	1.1	0.0002	-
CM	0.6372	1.0	0.0003	-
MV	0.6353	0.9	0.0000	-
TLS	0.6178	2.1	0.0009	-
PLS	0.6164	2.3	0.0011	-
RBF	0.6200	1.7	0.0010	-

Table 2: Inviscid lift and drag coefficients for transonic flow over airfoils

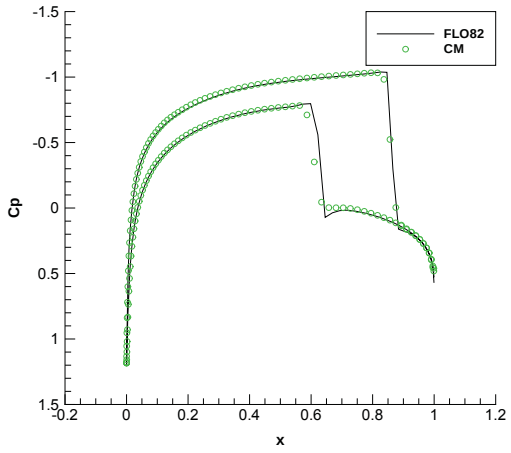


(a) Surface pressure coefficient

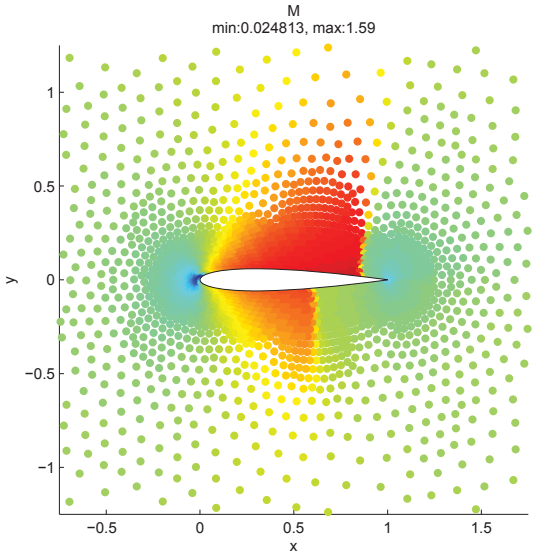


(b) Mach glyph plot

Figure 3: Flow over NACA 0012, $M = 0.80$, $\alpha = 1.25^\circ$

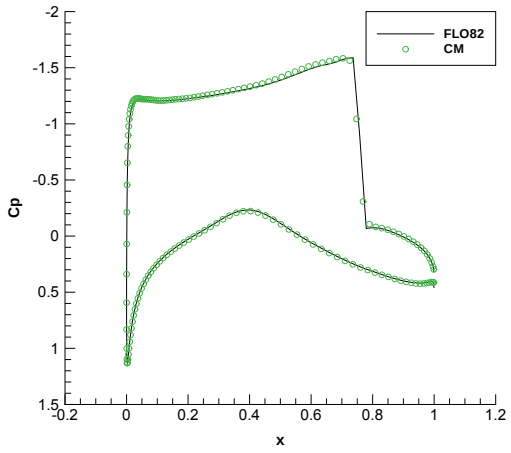


(a) Surface pressure coefficient

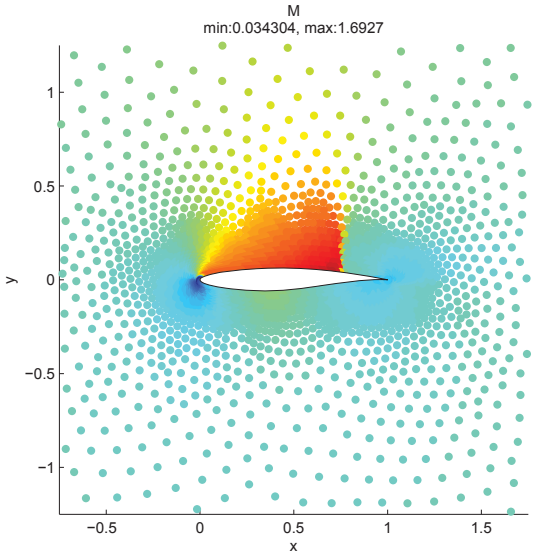


(b) Mach glyph plot

Figure 4: Flow over NACA 0012, $M = 0.85$, $\alpha = 1.0^\circ$

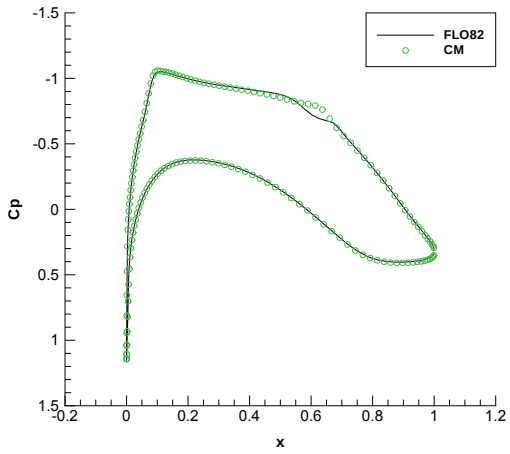


(a) Surface pressure coefficient

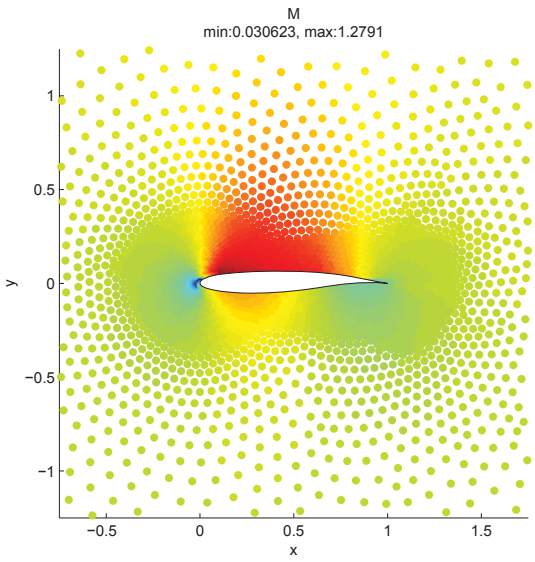


(b) Mach glyph plot

Figure 5: Flow over RAE 2822, $M = 0.75$, $\alpha = 3.0^\circ$



(a) Surface pressure coefficient



(b) Mach glyph plot

Figure 6: Flow over KORN airfoil, $M = 0.75$, $\alpha = 0.0^\circ$

IX. Acknowledgment

The first author acknowledges the support of this research by the PACCAR, Inc. Stanford Graduate Fellowship.

References

[1]Nayorles, B., Touzot, G., and Villom, P., "Generalizing the finite element method: Diffuse Approximation and Diffuse Elements," *Computational Mechanics*, Vol. 10, No. 5, 1992, pp. 307–318.

[2]Belytschko, T., Lu, Y. Y., and L., G., "Element-free Galerkin Methods," *International Journal for Numerical Methods in Engineering*, Vol. 37, No. 2, 1994, pp. 229–256.

[3]Duarte, C. A. and Oden, J., "H^p clouds - and h-p meshless method," *Numerical Methods for Partial Differential Equations*, Vol. 12, No. 6, 1996, pp. 673–705.

[4]Oñate, E., Idelsohn, Sergio Zienkiewicz, O., , and Taylor, R. L., "A Finite Point Method in Computational Mechanics. Applications to Convective Transport and Fluid Flow," *International Journal for Numerical Methods in Engineering*, Vol. 39, 1996, pp. 3839–3866.

[5]Löhner, R., Sacco, C., Oñate, E., and Idelssohn, S., "A Finite Point Method for Compressible Flow," *International Journal for Numerical Methods in Engineering*, Vol. 53, No. 8, 2002, pp. 1765–1779.

[6]Batina, J. T., "A Gridless Euler/Navier-Stokes Solution Algorithm for Complex Aircraft Applications," AIAA Paper 1993-0333, 31st AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA, 11–14 Jan. 1993.

[7]Kansa, E., "Multiquadrics–A scattered data approximation scheme with applications to computational fluid-dynamics–I surface approximations and partial derivative estimates," *Computers and Mathematics with Applications*, Vol. 19, No. 8–9, 1990, pp. 127–145.

[8]Kansa, E., "Multiquadrics–A scattered data approximation scheme with applications to computational fluid-dynamics–II solutions to parabolic, hyperbolic and elliptic partial differential equations," *Computers and Mathematics with Applications*, Vol. 19, No. 8–9, 1990, pp. 147–161.

[9]Shu, C., Ding, H., Chen, H., and Wang, T., "An upwind local RBF-DQ method for simulation of inviscid compressible flows," *Computer Methods in Applied Mechanics and Engineering*, Vol. 194, No. 18-20, 2005, pp. 2001 – 2017.

[10]Tota, P. V. and Wang, Z. J., "Meshfree Euler Solver Using Local Radial Basis Functions for Inviscid Compressible Flows," AIAA Paper 2007-4581, 18th AIAA Computational Fluid Dynamics Conference, Miami, FL, USA, 25–28 June 2007.

[11]Sridar, D. and Balakrishnan, N., "An upwind finite difference scheme for meshless solvers," *J. Comput. Phys.*, Vol. 189, No. 1, 2003, pp. 1–29.

[12]Katz, A. and Jameson, A., "Edge-based Meshless Methods for Compressible Flow Simulations," AIAA Paper 2008-699, 46th AIAA Aerodpace Sciences Meeting and Exhibit, Reno, NV, USA, 7–10 Jan. 2008.

[13]Grant, M. and Boyd, S., "Matlab software for disciplined convex programming (web page and software)," `http://stanford.edu/~boyd/cvx`, June 2009.

[14]Grant, M. and Boyd, S., *Graph implementations for nonsmooth convex programs, Recent Advances in Learning and Control (a tribute to M. Vidyasagar)*, Springer, 2008.

[15]CVXOpt, `http://abel.ee.ucla.edu/cvxopt/index.html`.

[16]Saunders, M. A. and Tomlin, J. A., "Interior-point solution of large-scale entropy maximization problems," Tech. rep., 18th International Symposium on Mathematical Programming, Copenhagen, Denmark, 18–22 Aug. 2003.

[17]Jameson, A., "Analysis and Design of Numerical Schemes for Gas Dynamics 2: Artificial Diffusion and Discrete Shock Structure," *International Journal of Computational Fluid Dynamics*, Vol. 5, 1995, pp. 1–38.

[18]Salas, M., Jameson, A., and Melnik, R., "A Comparative Study of the Nonuniqueness Problem of the Potential Equation," AIAA paper 83-1888, AIAA 6th Computational Fluid Dynamics Conference, Danvers, MA, 13–15 July 1983.

[19]Pulliam, T. H., "Efficient Solution Methods for the Navier-Stokes Equations," *Lecture Notes for the Von Karman Institute for Fluid Dynamics Lecture Series*, 1986.

[20]Jameson, A., "Analysis and Design of Numerical Schemes for Gas Dynamics 1: Artificial Diffusion, Upwind Biasing, Limiters and Their Effect on Accuracy and Multigrid Convergence," *International Journal of Computational Fluid Dynamics*, Vol. 4, 1995, pp. 171–218.

[21]Mavriplis, D. and Jameson, A., "Multigrid Solution of the Two-Dimensional Euler Equations on Unstructured Triangular Meshes," AIAA paper 1987-0353, AIAA 25th Aerospace Sciences Meeting, Reno, NV, 12–15 Jan. 1987.

[22]Jameson, A. and Mavriplis, D., "Finite Volume Solution of the Two-Dimensional Euler Equations on a Regular Triangular Mesh," AIAA Paper 1985-0435, 23rd Aerospace Sciences Meeting, Reno, NV, USA, 14–17 Jan. 1985.

[23]Jameson, A., Schmidt, W., and Turkel, E., "Numerical Solution of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time Stepping Schemes," Tech. rep., 23–25 June 1981.

[24]Katz, A. and Jameson, A., "A Comparison of Various Meshless Schemes Within a Unified Algorithm," AIAA Paper 2009-594, 47th AIAA Aerodpace Sciences Meeting and Exhibit, Orlando, FL, USA, 5–8 Jan. 2009.

[25]"Experimental Data Base for Computer Program Assessment," Tech. Rep. AR-138, AGARD, May 1979.

[26]"Test Cases for Inviscid Flowfield Methods," Tech. Rep. AR-211, AGARD, May 1985.

[27]Katz, A. and Jameson, A., "A Meshless Volume Scheme," AIAA Paper 2009-3534, 19th AIAA Computational Fluid Dynamics, San Antonio, TX, USA, 22–25 June 2009.

Appendix

A. Proofs of Global Conservation and Mimetic Properties

Here, we repeat the proof for global consertaion and include proofs for the summation by parts and energy conservation properties of the meshless differentiation operator.

Theorem 1 (Discrete Conservation)**.** *If m_i and a_{ij}^k satisfy conditions C-1 and C-2, then*

$$\sum_{i=1}^{n_p} m_i \delta^k u_i = \sum_{i \in s_B} u_i n_i^k \tag{2}$$

where s_B is the collection of all boundary points.

Proof. We can write the discrete first derivative as

$$m_i \delta^k u_i = \sum_{j=1}^{n_p} \tilde{a}_{ij}^k u_j \tag{3}$$

where $\tilde{a}_{ii}^k = a_{ii}^k$, $\tilde{a}_{ij}^k = a_{ij}^k$ if $j \in s_i$, and $\tilde{a}_{ij}^k = 0$ otherwise. Let $\phi \equiv 1$ in C-2, and use $a_{ij}^k + a_{ji}^k = 0$ from C-1, we get

$$\tilde{a}_{ij}^k = -\tilde{a}_{ji}^k \qquad \textit{only if } i \neq j \tag{4}$$

$$\sum_{\substack{j=1 \\ j \neq i}}^{n_p} \tilde{a}_{ji}^k = a_{ii}^k, \quad \text{or} \quad \sum_{j=1}^{n_p} \tilde{a}_{ji}^k = 2a_{ii}^k \tag{5}$$

Incorporate equation (3) into the left hand side of equation (2) and use equation (4), we have

$$\sum_{i=1}^{n_p} m_i \delta^k u_i = \sum_{i=1}^{n_p} \sum_{j=1}^{n_p} \tilde{a}_{ij}^k u_j = \sum_{j=1}^{n_p} (\sum_{i=1}^{n_p} \tilde{a}_{ij}^k) u_j = \sum_{j=1}^{n_p} 2a_{jj}^k u_j = \sum_{i=s_B} n_i^k u_i$$

where we changed the dummy index from j to i in the last step. □

Theorem 2 (Summation by parts)**.** *If m_i and a_{ij}^k satisfy conditions C-1 and C-2, then*

$$\sum_{i=1}^{n_p} m_i v_i \delta^k u_i + \sum_{i=1}^{n_p} m_i u_i \delta^k v_i = \sum_{i \in s_B} v_i u_i n_i^k \tag{6}$$

Proof. Substitute equation (3) into the left hand side of equation (6), we have

$$\begin{aligned}
&\sum_{i=1}^{n_p} m_i v_i \delta^k u_i + \sum_{i=1}^{n_p} m_i u_i \delta^k v_i \\
&\qquad = \sum_{i=1}^{n_p} v_i \sum_{j=1}^{n_p} \tilde{a}_{ij}^k u_j + \sum_{i=1}^{n_p} u_i \sum_{j=1}^{n_p} \tilde{a}_{ij}^k v_j \\
&\qquad = \sum_{i=1}^{n_p} v_i \sum_{j=1}^{n_p} \tilde{a}_{ij}^k u_j + \sum_{i=1}^{n_p} u_i (\sum_{\substack{j=1 \\ j \neq i}}^{n_p} -\tilde{a}_{ji}^k + a_{ii}^k) v_j \\
&\qquad = \sum_{i=1}^{n_p} \sum_{j=1}^{n_p} \tilde{a}_{ij}^k v_i u_j - \sum_{i=1}^{n_p} \sum_{j=1}^{n_p} \tilde{a}_{ij}^k v_i u_j + 2 \sum_{i=1}^{n_p} a_{ii} v_i u_i \\
&\qquad = \sum_{i \in s_B} v_i u_i n_i^k
\end{aligned}$$

where we exchanged the dummy indices i and j in the second term on the second-to-last line. □

Corollary 3 (Discrete energy conservation**).** *If m_i and a_{ij}^k satisfy conditions C-1 and C-2, then*

$$\sum_{i=1}^{n_p} m_i u_i \delta^k u_i = \sum_{i \in s_B} \frac{1}{2} u_i^2 n_i^k \tag{7}$$

Proof. Equation (7) is obtained by setting $v = u$ in equation (6). □