A CONSERVATIVE MESH-FREE SCHEME FOR CONSERVATION LAWS

QIQI WANG *, EDMOND KWAN-YU CHIU †, ANTONY JAMESON †, AND RUI HU †

Abstract. We present a mesh-free, finite-volume-like scheme for solving partial differential equations. We first derive a conservative and stable formulation of mesh free first order derivatives. This formulation is then generalized to allow flexible flux schemes, such as upwinding. The generalized scheme is analogous to finite volume schemes and is locally conservative. An algorithm is constructed for calculating the coefficients in our mesh free scheme. A numerical example with 2D inviscous advection equation demonstrates the stability and convergence of our scheme.

Key words. Conservation law, Advection equation, Mesh free scheme, Finite difference, Finite volume

AMS subject classifications. 35L65, 65M06, 65M08, 65M12, 65M50

1. Introduction. Despite significant improvement in technology and software tools, efficient generation of high quality meshes has remained the frequent bottleneck in scientific computing, especially when domain boundaries are characterized by non-trivial geometry.

To circumvent mesh generation, many have developed various classes of meshless algorithms, a large number of which falls within the spectrum between applying variational formulations in mesh-free settings and employing point collocation strategies to approximate quantities of interest. In the former category, for example, Nayroles et al. [1] developed the Diffuse Element (DE) method. Belytschko et al. [2] extended DE to Element Free Galerkin (EFG) method. These methods still 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. Sridar and Balakrishan[11], and Katz and Jameson[12] also developed meshless methods that resemble traditional finite difference methods by using Taylor series.

In addition to pure meshless schemes, many hybrid schemes that use meshless methods and traditional spatial discretization in different parts of the domain have been investigated. Belytschko et al. [13] coupled finite element method with EFG. Kirshman and Liu [14], Koh et al. [15] and Luo et al. [16] coupled meshless schemes near geometry surfaces with Cartesian grid methods in the rest of the domain. Kamatsuchi [17] performed large-scale viscous calculations using a meshless method in a near-wall subgrid that supplemented sub-divided Cartesian meshes through immersed boundary methods. Katz et al. [18] also used a meshless method as an interface between two different meshes.

*Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, U.S.A.

†Department of Aeronautics and Astronautics, Stanford University, Stanford, CA 94305, U.S.A.

The work mentioned above demonstrated the great potential in applying meshless methods to tackling complex problems in scientific computing. However, meshless methods also face a different set of challenges, the most fundamental of which, perhaps, amounts to the lack of formal conservation. Because of the local nature of the point collocation procedure, many existing mesh-free schemes do not preserve conservation at the discrete level except in very limited situations (such as with a uniform point distribution, which defeats the purpose of meshless schemes). Compared to some mesh-based approaches, the lack of conservation hinders efficiency as one cannot compute reciprocal fluxes 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 memetic properties at the discrete level, but also provide the option to incorporate existing schemes for estimating fluxes when solving conservation laws. Accordingly, the rest of the paper is organized into a theoretical part and a implementation part. The theoretical part starts from the definition of the discrete derivative operator for our meshless scheme in section 2, along with the reciprocity and consistency conditions that it should satisfy. Based on those conditions, we prove in sections 3 and 4, respectively, the resulting global and local conservation properties of the scheme. These properties lead to analogs of finite volume schemes and a generalization in section 5 that allows flexible flux schemes. Section 6 outlines the scheme's discrete geometric properties, which lead to the procedure in section 7 for generating the necessary coefficients for the scheme. The remaining sections contain results from testing this procedure on the advection equation using both the default and upwinding flux construction, followed by concluding remarks.

2. Differentiation operator. We discretize a complex domain using a cloud of points x_i, $i = 1, \ldots, N$. The point cloud contains points both on the boundary and in the interior of the domain. Each point i has a number of neighboring points, denoted by the set s_i. s_i does not include point i itself, and the reciprocal condition $j \in s_i \Leftrightarrow i \in s_j$ is satisfied. The discrete first derivative is defined as

$$m_i \partial^k \phi_i \approx m_i \delta^k \phi_i = a_{ii}^k \phi_i + \sum_{j \in s_i} a_{ij}^k \phi_j \tag{2.1}$$

where $k = 1, 2, 3$ is the spatial dimension; ∂^k and δ^k are the analytic and discrete first derivative operators in the kth spatial coordinate. Here, m_i can represent some volume associated with each node, while the coefficients a_{ij} for the node pairs (i, j) then have corresponding dimensions of area (we shall justify this characterization in section 4). Note that although we perform our analysis in 3D, they are also applicable in 2D, where $k = 1, 2$. We enforce the following two conditions on a_{ij} and m_i:

C-1. *Reciprocity of coefficients*:

$$\begin{aligned} a_{ij}^k &= -a_{ji}^k, && i \neq j \\ a_{ii}^k &= 0, && i \text{ interior point} \\ a_{ii}^k &= \frac{1}{2} n_i^k, && i \text{ boundary point} \end{aligned}$$

where n_i^k is the kth coordinate of the outward facing boundary normal times the boundary area occupied by the boundary point.

C-2. *Consistency of order L*:

$$a_{ii}^k p(x_i) + \sum_{j \in s_i} a_{ij}^k p(x_j) = m_i \partial^k p(x_i)$$

for all multivariate polynomials p of total order L, where x_i is the coordinate of the ith point.

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

3. Global conservation and mimetic properties. The analytic first derivative operator ∂^k has the following properties:

- Conservation

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

- Integration by parts

$$\int_\Omega \psi \partial^k \phi dx = \int_{\partial\Omega} \psi \phi n^k ds - \int_\Omega \phi \partial^k \psi dx$$

- Enegy conservation

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

This section proves that the discrete operator has corresponding properties, if it satisfies the conditions in Section 1.

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

$$\sum_{i=1}^N m_i \delta^k \phi_i = \sum_{i \in s_B} \phi_i n_i^k \tag{3.1}$$

where s_B is the collection of all boundary points.

Proof. We can write the discrete first derivative as

$$m_i \partial^k \phi_i \approx m_i \delta^k \phi_i = \sum_{j=1}^N \tilde{a}_{ij}^k \phi_j \tag{3.2}$$

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

$$\tilde{a}_{ij}^k = -\tilde{a}_{ji}^k \qquad \textit{only if } i \neq j \tag{3.3}$$

$$\sum_{\substack{j=1 \\ j \neq i}}^N \tilde{a}_{ji}^k = a_{ii}^k \tag{3.4}$$

and also

$$\sum_{j=1}^N \tilde{a}_{ji}^k = 2a_{ii}^k \tag{3.5}$$

Incorporate Equation (3.2) into the left hand side of Equation (3.1) and use Equation (3.3), we have

$$\begin{aligned}\sum_{i=1}^{N} m_i \delta^k \phi_i &= \sum_{i=1}^{N}\sum_{j=1}^{N} \tilde{a}_{ij}^k \phi_j = \sum_{j=1}^{N}(\sum_{i=1}^{N} \tilde{a}_{ij}^k)\phi_j \\ &= \sum_{j=1}^{N} 2a_{jj}\phi_j \\ &= \sum_{i=s_B} n_i^k \phi_i \end{aligned} \tag{3.6}$$

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*

$$\sum_{i=1}^{N} m_i \psi_i \delta^k \phi_i + \sum_{i=1}^{N} m_i \phi_i \delta^k \psi_i = \sum_{i \in s_B} \psi_i \phi_i n_i^k \tag{3.7}$$

Proof. Substitute Equation (3.2) into the left hand side of Equation (3.7), we have

$$\begin{aligned}&\sum_{i=1}^{N} m_i \psi_i \delta^k \phi_i + \sum_{i=1}^{N} m_i \phi_i \delta^k \psi_i \\ &\quad= \sum_{i=1}^{N} \psi_i \sum_{j=1}^{N} \tilde{a}_{ij}^k \phi_j + \sum_{i=1}^{N} \phi_i \sum_{j=1}^{N} \tilde{a}_{ij}^k \psi_j \\ &\quad= \sum_{i=1}^{N} \psi_i \sum_{j=1}^{N} \tilde{a}_{ij}^k \phi_j + \sum_{i=1}^{N} \phi_i (\sum_{\substack{j=1 \\ j\neq i}}^{N} -\tilde{a}_{ji}^k + a_{ii}^k)\psi_j \\ &\quad= \sum_{i=1}^{N} \psi_i \sum_{j=1}^{N} \tilde{a}_{ij}^k \phi_j + \sum_{i=1}^{N} \phi_i (\sum_{\substack{j=1 \\ j\neq i}}^{N} -\tilde{a}_{ji}^k - a_{ii}^k + 2a_{ii}^k)\psi_j \\ &\quad= \sum_{i=1}^{N} \psi_i \sum_{j=1}^{N} \tilde{a}_{ij}^k \phi_j - \sum_{i=1}^{N} \phi_i \sum_{j=1}^{N} \tilde{a}_{ji} \psi_j + 2\sum_{i=1}^{N} \phi_i a_{ii} \psi_i \\ &\quad= \sum_{i=1}^{N}\sum_{j=1}^{N} \tilde{a}_{ij}^k \psi_i \phi_j - \sum_{i=1}^{N}\sum_{j=1}^{N} \tilde{a}_{ij}^k \psi_i \phi_j + 2\sum_{i=1}^{N} a_{ii} \psi_i \phi_i \\ &\quad= \sum_{i \in s_B} \psi_i \phi_i n_i^k \end{aligned} \tag{3.8}$$

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} satisfies Conditions C-1 and C-2, then*

$$\sum_{i=1}^{N} m_i \phi_i \delta^k \phi_i = \sum_{i \in s_B} \frac{\phi_i^2}{2} n_i^k \tag{3.9}$$

Proof. Equation (3.9) is obtained by setting $\psi = \phi$ in Equation (3.7). □

4. Local conservation. In addition to global conservation, it is also desirable for a scheme to preserve local conservation, i.e.

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

at the discrete level. We shall prove that our scheme does so indeed.

THEOREM 4 (Local Discrete Conservation). *If a_{ij} and m_i satisfy Conditions C-1 and C-2, then, defining $n_{f_j} = 2a_{ij}$ to be some virtual face area vectors for interior point pairs, the following conditions hold:*

1. *At each node i,*

$$\begin{aligned} m_i \delta^k \phi_i &= \sum_{j \in s_i} \phi_{f_j} n^k_{f_j} + \phi_i n^k_i && i \in s_B \\ m_i \delta^k \phi_i &= \sum_{j \in s_i} \phi_{f_j} n^k_{f_j} && i \notin s_B \end{aligned} \tag{4.1}$$

2. *In addition, m_i represent some virtual volume at node i that is enclosed by some virtual faces between node i and node j (plus boundary faces for $i \in s_B$), which are denoted by $\vec{n}_{f_j}$ with function values ϕ_{f_j}.*

To faciliate the proof, we introduce the following corollary.

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

$$\nabla \cdot \vec{p} = \sum_{k=1}^{3} \partial^k p^k = 0$$

, then the following condition holds

$$\begin{aligned} \vec{p}(x_i) \cdot \vec{n_i} + \sum_{j \in s_i} \vec{p}_{f_j} \cdot \vec{n}_{f_j} &= 0 && i \in s_B \\ \sum_{j \in s_i} \vec{p}_{f_j} \cdot \vec{n}_{f_j} &= 0 && i \notin s_B \end{aligned} \tag{4.2}$$

We shall proof the above theorem and corollary together. As one will see, Equation (4.1) directly leads to Equation (4.2), which in turn leads to condition 2 above required to complete the proof of Theorem 4.

Proof. Applying Condition C-2 to $p \equiv 1$ leads to $a^k_{ii} + \sum_{j \in s_i} a^k_{ij} = 0$. Multiplying this by ϕ_i and adding the result to the definition of the first derivative operator (2.1), we have

$$\begin{aligned} m_i \delta^k \phi_i &= a^k_{ii} \phi_i + \sum_{j \in s_i} a^k_{ij} \phi_j + a^k_{ii} \phi_i + \sum_{j \in s_i} a^k_{ij} \phi_i \\ &= 2a^k_{ii} \phi_i + \sum_{j \in s_i} a^k_{ij} (\phi_i + \phi_j) \\ &= 2a^k_{ii} \phi_i + \sum_{j \in s_i} n^k_{ij} \frac{(\phi_i + \phi_j)}{2} \end{aligned} \tag{4.3}$$

For interior points, $a_{ii} = 0$. For boundary points, $a_{ii} = \frac{n_i}{2}$. If we let $\phi_{f_j} = \frac{\phi_i + \phi_j}{2}$, then we obtain equation (4.1).

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

$$\begin{aligned} m_i \partial^k p_i &= \sum_{j \in s_i} p_{f_j} n^k_{f_j} + p_i n^k_i && i \in s_B \\ m_i \partial^k p_i &= \sum_{j \in s_i} p_{f_j} n^k_{f_j} && i \notin s_B \end{aligned} \tag{4.4}$$

If $\vec{p}$ is divergence free, summing over Equation (4.4) applied to each component of $\vec{p}$ results in Equation (4.2).

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

First, let $\phi = x^k$ (recall k denotes spatial dimension). Equation (4.1) becomes $\sum_{j \in s_i} n_{f_j} \frac{x^k_i + x^k_j}{2} = m_i$ for an interior point i, and $\sum_{j \in s_i} n_{f_j} \frac{x^k_i + x^k_j}{2} + n^k_i x^k_i = m_i$ for a boundary point.

Then, Corollary 5 applied to $p \equiv 1$ yields $\sum_{j \in s_i} n_{f_j} = 0$ for an interior point i and $\sum_{j \in s_i} n_{f_j} + n^k_i = 0$ 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 normals and volumes. The scheme is locally conservative. □

Theorem 4 justifies the geometric interpretation of the coefficients a_{ij} and m_i as analogs of face areas and volumes. This geometric interpretation also allow us to generalize our scheme, as shown in the next section.

5. Generalization. In addition to the geometric interpretation of the coefficients a_{ij} and m_i, $a^k_{ij} = -a^k_{ji}$ in Condition C-1 guarantees that the face area and normal are consistent as seen by volumes i and j. A set of n^k_i, m_i and a^k_{ij} (such as one generated by the algorithm in Section 7), can still respresent boundary normals, cell volumes and interface normals. However, with the above geometric information, instead of using the central average flux $\phi_f = \frac{\phi_i + \phi_j}{2}$, we can generalize out scheme to use more sophisticated flux while preserving conservation.

The following formula generalizes Equation (4.3):

$$m_i \delta^k_f \phi_i = 2 a^k_{ii} \phi_i + \sum_{j \in s_i} 2 a^k_{ij} F_{ij} \tag{5.1}$$

where the interface flux F_{ij} can be a function of ϕ_i, ϕ_j and the derivative of ϕ and the i-th and j-th points. This general formulation also preserves the global conservation property of the original scheme (2.1), but loses the integration by parts and kinetic energy conservation properties.

An example of the generalized derivative operator is the upwinding flux scheme. In this scheme, the interface flux F_{ij} is defined as

$$F_{ij} = \begin{cases} \phi_i + \dfrac{x^k_j - x^k_i}{2} \delta^k_1 \phi_i, & a_{ij} < 0 \\ \phi_j + \dfrac{x^k_i - x^k_j}{2} \delta^k_1 \phi_j, & a_{ij} > 0 \end{cases} \tag{5.2}$$

where δ_1^k is a first order accurate reconstruction of the derivative in the k-th spatial dimension. In the numerical examples detailed in this paper, it is constructed as

$$\delta_1^k \phi_i = \sum_{j \in s_i} a_{1ij}^k (\phi_j - \phi_i)$$

where a_{1ij}^k are chosen by solving

$$\min \sum_{j \in s_i} (a_{1ij}^k)^2 \quad \text{s.t.} \quad \sum_{j \in s_i} a_{1ij}^k (x_j^{k'} - x_i^{k'}) = \delta_{kk'}$$

for each i and k, where $k' = 1, 2, 3$.

This upwinding interface flux F_{ij} is second order accurate, making the derivative approximation (5.1) first order accurate. Our numerical results in Section 10 demonstrate the desirable performance of this scheme.

6. Global divergence theorem and geometric conservation law. Before we present our algorithm to compute a_{ij} and m_i, we discuss some extra global properties of our scheme. Resulting from conditions C-1 and C-2, these properties parallel Equation (4.4) and Corollary 5. They provide important insight into constructing the algorithm in Section 7 for generating the coefficients.

THEOREM 6 (Discrete divergence theorem). *If a_{ij} and m_i satisfies Conditions C-1 and C-2, then the following condition holds for all multivariate polynomials p of total order $2L$:*

$$\sum_{i \in s_B} p(x_i) n_i^k = \sum_{i=1}^{N} m_i \partial^k p(x_i) \tag{6.1}$$

Proof. It is sufficient to prove (6.1) for all multivariate monomials p of order less or equal to $2L$. Let $p = p_1 p_2$, where both p_1 and p_2 are monomials or order less or equal to L, and thus satisfy Condition C-2:

$$a_{ii}^k p_1(x_i) + \sum_{j \in s_i} a_{ij}^k p_1(x_j) = m_i \partial^k p_1(x_i) \tag{6.2}$$

$$a_{ii}^k p_2(x_i) + \sum_{j \in s_i} a_{ij}^k p_2(x_j) = m_i \partial^k p_2(x_i) \tag{6.3}$$

Multiply Equation (6.2) by $p_2(x_i)$, and Equation (6.3) by $p_1(x_i)$; then add them together and sum over $i = 1, \ldots, N$. Using the fact that

$$\sum_{(i,j) \in E} a_{ij}^k \big(p_1(x_i) p_2(x_j) + p_1(x_j) p_2(x_i)\big) = 0$$

derived from $a_{ij} + a_{ji} = 0$ in Condition C-1, we get

$$\sum_{i=1}^{N} 2 a_{ii}^k p_1(x_i) p_2(x_i) = \sum_{i=1}^{N} m_i \partial^k \big(p_1(x_i) p_2(x_i)\big)$$

Using the formula for a_{ii} in Condition C-1, we get Equation (6.1) for $p = p_1 p_2$. ∎

In this proof, we took linear combinations of the constraints in condition C-2, canceled out all a_{ij}'s, and got Equation (6.1). Therefore, this theorem shows that Conditions C-1 and C-2, as linear constraints for a_{ij}, are **overdetermined**. In other words, these conditions can not be satisfied simultaneously unless m_i satisfies Equation (6.1).

The following corollary shows that equation (6.1) as linear constraints for m_i are also overdetermined. A solution for (6.1) exists only if the boundary normals n_i^k are chosen appropriately.

COROLLARY 7 (Geometric conservation law). *If a_{ij} and m_i satisfies Conditions C-1 and C-2, and the vector valued multivariate polynomials $\vec{p}$ of order $2L$ satisfies the divergence free condition*

$$\nabla \cdot \vec{p} = \sum_{k=1}^{3} \partial^k p^k = 0$$

then the following condition holds

$$\sum_{i \in s_B} \vec{p}(x_i) \cdot \vec{n}_i = 0 \tag{6.4}$$

Proof. Equation (6.4) is obtained by summing Equation (6.1) over $k = 1, 2, 3$ and use the divergence free condition. □

7. Operator construction. As we saw in the previous section, Theorem 6 and Corollary 7 means that n_i^k must be chosen first so that Equation (6.4), which is a necessary condition for the existence of solution to Equation (6.1), is satified. Then, one can generate m_i satisfying Equation (6.1), which is a necessary condition for the existence of a_{ij}^k that satisfies Conditions C-1 and C-2.

The following algorithm constructs $\vec{n}_i = (n_i^1, n_i^2, n_i^3)$, m_i and $\vec{a}_{ij} = (a_{ij}^1, a_{ij}^2, a_{ij}^3)$ that satisfies Conditions C-1 and C-2.

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 (6.4).
3. With initial estimates of $m_i = 0$, project m_i into the linear manifold that satisfies (6.1).
4. Solve a constrained least squares problem for $\vec{a}_{ij}$ to enforce Conditions 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 $\{(i,j) | j \in s_i\}$.

The projections in steps 2 and 3 involves a small number of constraints when L is small, and are solved with QR and singular value decompositions constraint matrices, respectively. Specifically, let $\mathbf{n}$ be a column vector that contains $\vec{n}_i$ for all boundary points i, the geometric conservation law (6.4) can be written as

$$G^{\mathrm{T}}\mathbf{n} = 0\,. \tag{7.1}$$

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

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

$$\mathbf{n} = \mathbf{n}_0 - QQ^{\mathrm{T}}\mathbf{n}_0,$$

where $QR = G$ is the thin QR decomposition of G. The projected $\mathbf{n}$ satisfies the linear equation (7.1), which is equivalent to the geometric conservation law (6.4).

To enforce the discrete divergence theorem (6.1), we denote $\mathbf{m}$ as a column vector that contains m_i for all points, and write Equation (6.1) as

$$D^{\mathrm{T}}\mathbf{m} = E^{\mathrm{T}}\mathbf{n} \tag{7.2}$$

The matrices D and E have the same number of columns, which is equal to the number of linearly independent multivariate polynomials of maximum order $2L$. Each column of D contains the divergence of one of these polynomials at all points; the corresponding column of E contains the value of the polynomial at all boundary points. When L is small, both D and E are thin matrices.

In order to compute $\mathbf{m}$ that satisfies Equation (7.2), we perform the thin Singular Value Decomposition $D = USV^{\mathbf{T}}$. Equation (7.2) becomes

$$SU^{\mathrm{T}}\mathbf{m} = V^{\mathrm{T}}E^{\mathrm{T}}\mathbf{n}$$

Due to the geometric conservation law (6.4), the matrix D is singular. The number of zero singular values of D is equal to the number of columns of matrix G. If $\mathbf{n}$ satisfies (7.1), the rows of $V^{\mathrm{T}}E^{\mathrm{T}}\mathbf{n}$ corresponding to the zero singular values are 0. Let b_1 be the its rows of $V^{\mathrm{T}}E^{\mathrm{T}}\mathbf{n}$ corresponding to the nonzero singular values, U_1 be the columns of U corresponding to the nonzero singular values, and S_1 be the square submatrix of S corresponding to the nonzero singular values, Equation (7.2) becomes

$$S_1U_1^{\mathbf{T}}\mathbf{m} = b_1 \ ,$$

which can be satisfied by

$$\mathbf{m} = U\, S_1^{-1}b_1 \ .$$

The least squares in step 4 involves many constraints and is solved using Minimum residual iterations. We denote $\mathbf{a}$ as a column vector that contains $\vec{a}_{ij}$ for all neighborhood pairs. For each neighborhood pair (i, j), either $\vec{a}_{ij}$ or $\vec{a}_{ji}$ are stored, such that Condition C-1 ($\vec{a}_{ij} = -\vec{a}_{ji}$) is automatically satisfied. We write Condition C-2 in the linear form

$$C^{\mathrm{T}}\mathbf{a} = \mathbf{d} \ ,$$

where $C^{\mathrm{T}}\mathbf{a}$ contains the term $\sum_{j\in s_i} a_{ij}^k p(x_j)$, and $\mathbf{d}$ contains both the $a_{ii}^k p(x_i)$ and $m_i\partial^k p(x_i)$ terms.

This system is then solved using LSQR, first developed by Paige and Saunders [19]. LSQR is capable of handling matrices of arbitrary ranks and dimensions. Note that although the system is singular due to the discrete divergence theorem (6.1), it has a compatible right hand side constructed by choosing m_i and $\vec{n}_i$ that satisfies Equation (7.2).

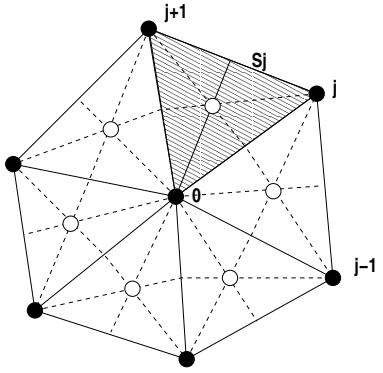

FIG. 8.1. *Section of regular triangular mesh for nodal finite-volume scheme with primal nodes (filled) and median-dual nodes (hollow)*

8. Existing schemes satisfying C-1 and C-2. To further motivate that our current scheme is a natural mesh-free analog of conservative finite-volume schemes, we shall give an example of an existing finite volume scheme satisfying Condition C-1 and Condition C-2 with $L = 1$.

Consider a nodal finite volume scheme on the median dual of a regular triangular mesh, a representative portion of which is denoted in figure 8.

In each triangle, Gauss theorem gives

$$S\frac{\partial u}{\partial x} = \int u dy \simeq \sum_k u_j \Delta y_j \tag{8.1}$$

This result is exact for polynomials of order 1.

For a conservation law

$$\frac{\partial \phi}{\partial t} + \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} = 0 \tag{8.2}$$

The following approximation of the integral form

$$\begin{aligned}(\sum_j S)\frac{\partial \phi}{\partial t}\Big|_0 &= -\frac{1}{2}\sum_k \{(f_{j+1} + f_j)(y_{j+1} - y_j) - (g_{j+1} + g_j)(x_{j+1} - x_j)\} \\ &= -\frac{1}{2}\sum_k \{(f_j)(y_{j+1} - y_{j-1}) - (g_j)(x_{j+1} - x_{j-1})\}\end{aligned} \tag{8.3}$$

is also exact for a polynomial of order 1 with $f = c_1\phi, g = c_2\phi$, i.e. if the governing equations are homogeneous of order 1. This is indeed the case for several conservation laws, including the advection equation and Euler Equations for inviscid, compressible flows. Jameson and Mavripilis [20] have previously successfully obtained numerical solutions to the Euler Equations using this scheme.

However, it also turns out that some commonly-used schemes do not satisfy Conditions C-1 and C-2 with $L \leq 1$. Without diving into details, we point out that finite difference / volume schemes constructed in a fashion similar to the one above does not satisfy Condition C-2 for $L = 1$ when used on a structured, but non-uniform, quadrilateral mesh.

9. Advection equation. Consider the equation

$$\frac{\partial \phi}{\partial t} + \vec{u} \cdot \nabla \phi = 0 \tag{9.1}$$

with boundary condition $\phi(x,t) = \phi_B(x,t)$ at the inlet part of the boundary $\{x \in \partial\Omega \mid \vec{n} \cdot \vec{u} < 0\}$. The advection velocity $\vec{u} = (u^1, u^2, u^3)$ is a constant. This equation is discretized as [1]

$$\frac{d\phi_i}{dt} + u^k \delta^k \phi_i = \begin{cases} 0 & i \notin s_B \\ \dfrac{\phi_B - \phi_i}{m_i} [u_i^n]_- & i \in s_B \end{cases} \tag{9.2}$$

where $u_i^n = n_i^k u^k$; $[n_i^n]_- = -\min(0, n_i^n)$ denotes the negative part. The penalty term in the discretization (9.2) is consistent with the continuous boundary condition.

The stability of this discretization can be proven using Corollary 3. We multiply Equation (9.2) by $m_i \phi_i$, and sum over all i. Using equation (3.9), we get

$$\frac{d\mathbf{E}}{dt} = -\sum_{i \in s_B} \left(-\frac{\phi_i^2}{2} u_i^n - (\phi_i^2 - \phi_i \phi_B)[u_i^n]_- \right)$$

where the energy is defined as

$$\mathbf{E} = \sum_{i=1}^{N} \frac{1}{2} m_i \phi_i^2$$

By splitting u_i^n into positive and negative parts $u_i^n = [u_i^n]_+ - [u_i^n]_-$, we further get

$$\frac{d\mathbf{E}}{dt} = \frac{1}{2} \sum_{i \in s_B} \left(-\phi_i^2 [u_i^n]_+ - (\phi_i - \phi_B)^2 [u_i^n]_- + \phi_B^2 [u_i^n]_- \right)$$

The first two terms, the energy convected out of the domain and the penalty term on the boundary, cannot increase the total energy; the third term, the energy convected into the domain, dependents only on the boundary condition. Therefore, the energy can not increase exponentially, and the scheme (9.2) is stable.

However, because $a_{ii} = 0$ for all the interior nodes, estimates of the derivative only depend on the function values at neighbor points. This in turn may lead to grid-to-grid oscillations due to decoupling of points in the domain, depending on the final point connectivity. The stability of both schemes and the oscillatory nature of this particular flux are shown numerically in Section 10.

Alternatively, we can discretize Equation (9.1) with the upwinding derivative δ_f^k discussed in Section 5:

$$\frac{d\phi_i}{dt} + u^k \delta_f^k \phi_i = \begin{cases} 0 & i \notin s_B \\ \dfrac{\phi_B - \phi_i}{m_i} [u_i^n]_- & i \in s_B \end{cases} \tag{9.3}$$

Although this scheme looses the kinetic energy conserving property of the central flux scheme (9.2), the global and local conservation properties are still satisfied. By introducing numerical dissipation in the form of upwinding, this scheme removes the grid-by-grid oscillation of the central flux scheme. The stability and convergence of this scheme is demonstrated numerically in Section 10.

[1] Subscript k follows Einstein notation.

10. Numerical results. This section demonstrate our schemes (9.2) and (9.3) by solving Equation (9.1) in the 2D domain $\Omega = \{x \mid 1 < \|x\|_2 < 5\}$. Four sets of point cloud are used, with number of points N being 150, 506, 1,885 and 7,182, respectively.

10.1. Point cloud and neighborhood sets. The point cloud is the union of a Cartesian grid in the interior and evenly spaced points on the boundary. To generate

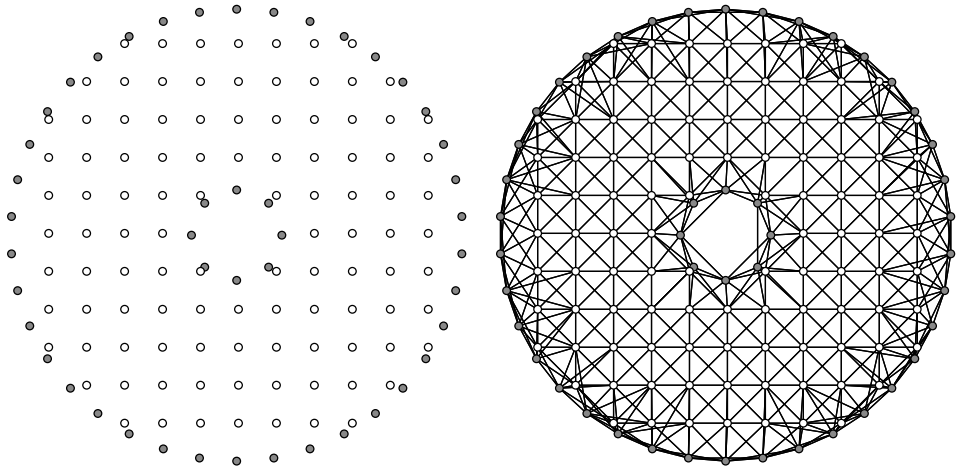

FIG. 10.1. *The point cloud and the neighborhood pairs with* 150 *points (the coarsest point cloud). Interior points are the open circles, and interior points are the filled circles. A line connecting two points indicates that they are mutual neighbors.*

the neighborhood set s_i for each point i, we first set s_i to be the 8 nearest points to i. The reciprocal condition is then enforced by setting $s_i = s_i \cup \{j \mid i \in s_j\}$. The point cloud and neighborhood sets for $N = 150$ is plotted in Figure (10.1). To make the neighbor search algorithm efficient, we perform a Cartesian subdivision of the domain, and search the neighbors of each point only in its neighboring subdivisions. The resulting process of point cloud and neighborhood sets generation is very efficient and has computational complexity of $O(N)$.

10.2. Construction of n_i^k, m_i and a_{ij}^k. We generate the operator with polynomial order $L = 1$. An initial estimate of n_i^k for each boundary point i is obtained by averaging the normals of the adjacent two boundary edges. n_i^k are then projected into the linear subspace that satisfies the geometric conservation law (6.4) using QR decomposition. We then set the initial guess for m_i to 0, and project them to the linear manifold that satisfies the divergence theorem (6.1) using SVD. The detailed algorithm is described in Section 7. Both geometric conservation law and divergence theorem are satisfied with accuracy better than 10^{-11} on all four point clouds.

After n_i^k and m_i are computed to satisfy the geometric conservation law and the discrete divergence theorem, Conditions C-1 and C-2 compatible. we then solve the system for a_{ij}^k using LSQR. On the finest point cloud with $N = 7,182$, Condition C-2 is enforced to accuracy of 10^{-6} for $p \equiv 1$, and 10^{-5} for $p = x$ and $p = y$.

10.3. Solution of the advection equation. The advection equation (9.1) is solved with advection velocity $(u^1, u^2) = (-\sqrt{2}/2, \sqrt{2}/2)$. The initial and boundary

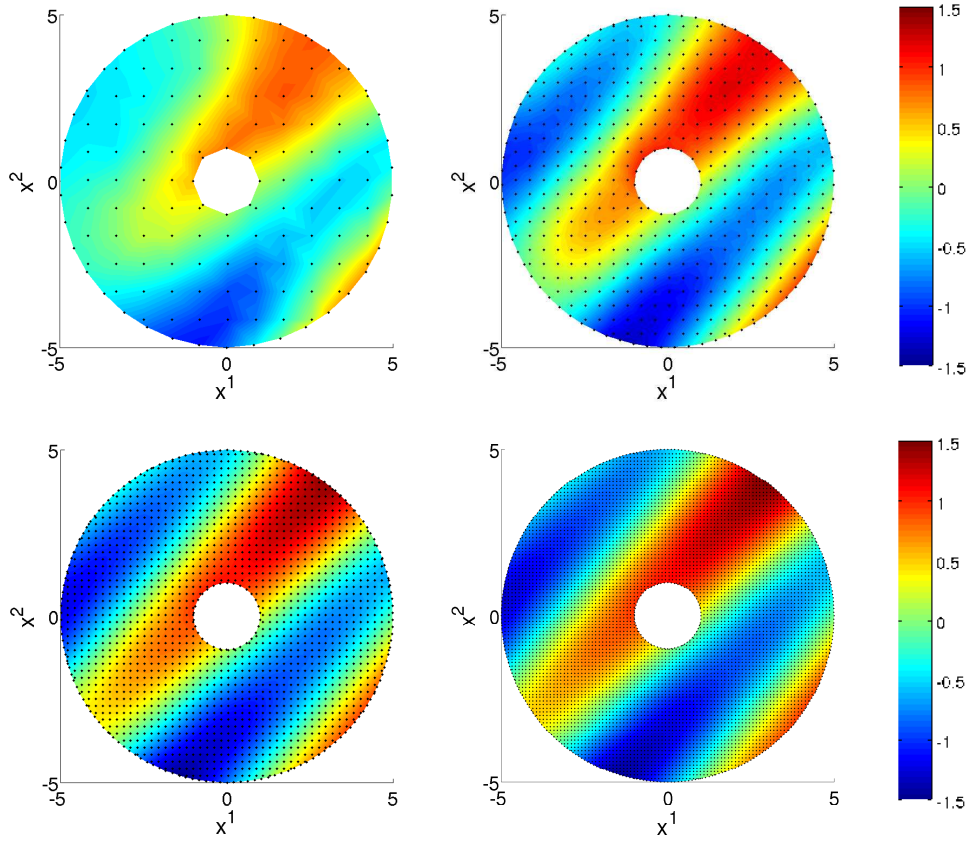

FIG. 10.2. *Solution of the advection equation at* $t = 100$ *using the upwinding scheme (9.3). The upper left, upper right, lower left and lower right plots correspond to point clouds of size* $N = 150$, 506, $1,885$ *and* $7,182$, *respectively. The black dots represent the points used in the solution.*

conditions are set to

$$\phi(x,t) = \frac{\sqrt{2}}{20}(x+y) + 1 \,, \qquad\qquad t = 0 \,;$$
$$\phi(x,t) = \frac{\sqrt{2}}{20}(x+y) + \cos[t - 5 - u^k x^k]_+ \,, \quad x \in \partial\Omega, \;\; n^k u^k < 0 \,.$$

The exact solution for this initial-boundary-value problem is

$$\phi(x,t) = \frac{\sqrt{2}}{20}(x+y) + \cos[t - 5 - u^k x^k]_+ \,.$$

The numerical solution is computed using two schemes: (9.2) and (9.3). Four point clouds of different sizes are used for each scheme. The L_∞ and L_2 norms of the numerical error in the solution is computed against the analytic solution.

Figures 10.3 and 10.3 shows the numerical solution at $t = 100$ and the numerical error of the upwinding flux scheme (9.3). We observe that the convergence rate of this scheme is between first and second order accurate, although the official rate of approximation is only first order. The numerical solution by the central flux scheme

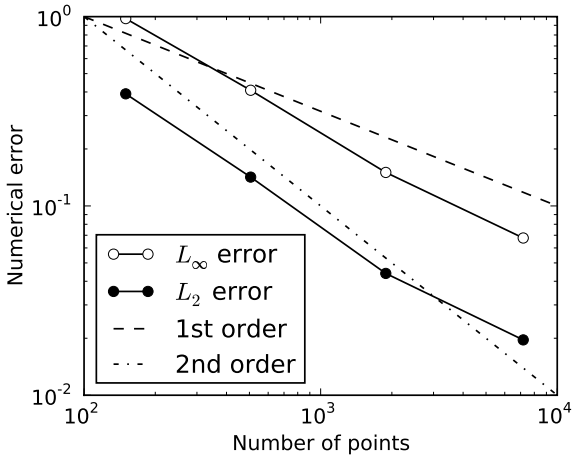

FIG. 10.3. *The numerical error of the upwinding scheme (9.3) as a function of the point cloud size N.*

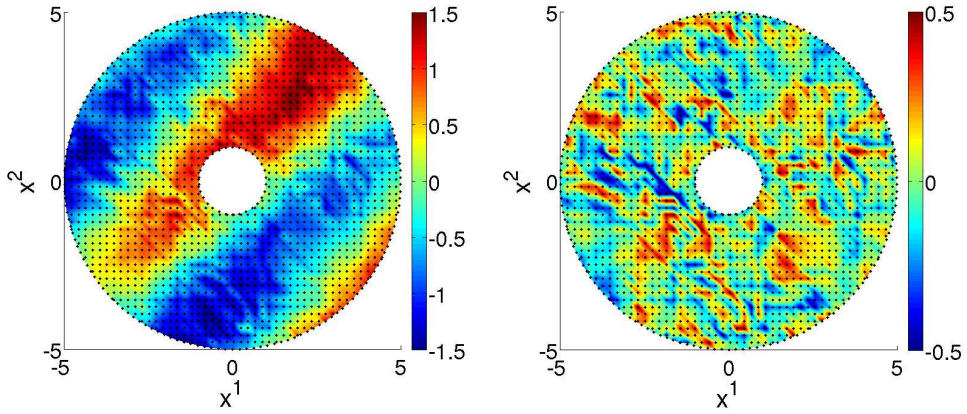

FIG. 10.4. *Solution and numerical error of the advection equation at $t = 100$ using the central flux scheme (9.2), with a point cloud of size.* $1,885$. *The left plot shows the numerical solution, and the right plot shows the difference between the numerical and the analytic solution. The black dots represent the point cloud used in the solution.*

(9.2), however, suffers from severe grid-by-grid oscillation. Figure 10.3 shows this oscillation on a point cloud of size $1,885$. Figure 10.3 further shows that this oscillation does not improve as the point cloud refines.

11. Conclusion. We have formulated a mesh free derivative operator by enforcing two conditions C-1 and C-2. These conditions guarantee that our operator is discretely conservative and has certain mimetic properties, such as summation by parts and kinetic energy conservation. Based on its local conservation properties, we further derived a generalization of the mesh free scheme based on finite-volume-like flux formulation. The upwinding flux was given as an example of the generalized

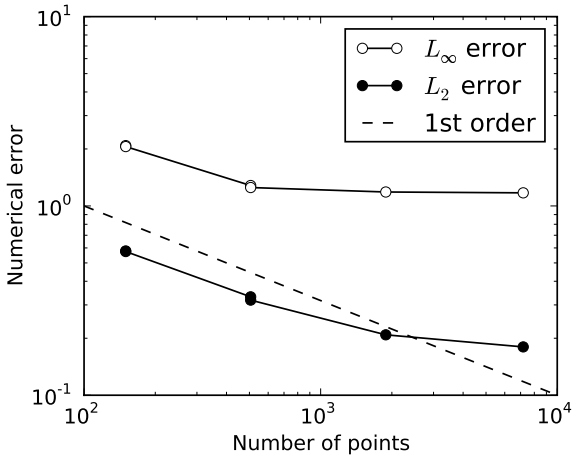

FIG. 10.5. *The numerical error of the central scheme (9.2) as a function of the point cloud size* N.

scheme. In addition, we analyzed the necessary conditions for which Conditions C-1 and C-2 can be satisfied. An algorithm is constructed to enforce these conditions and compute the mesh free operator. We use the obtained operator on a two dimensional inviscous advection equation. Numerical results show that the discretization is stable with both the original and the generalized schemes. Both schemes are numerically stable. While the central flux scheme suffers from grid-by-grid oscillations, the generalized scheme using upwinding flux calculates smooth solutions and converges as the point cloud refines.

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 Aerodpace 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, Prasad, 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] Belytschko, T., Organ, D., and Krongauz, Y., "A Coupled Finite Element-Element-free Galerkin Method," *Computational Mechanics*, Vol. 17, No. 3, 1995, pp. 186–195.

[14] Kirshman, D. J. and Liu, F., "A gridless boundary condition method for the solution of the Euler equations on embedded Cartesian meshes with multigrid," *Journal of Computational Physics*, Vol. 201, No. 1, 2004, pp. 119 – 147.

[15] Koh, E., Tsai, H. M., and Liu, F., "Euler Solution Using Cartesian Grid with a Gridless Least-Squares Boudnary Treatment," *AIAA Journal*, Vol. 43, No. 2, 2005, pp. 246 – 255.

[16] Luo, H., Baum, J. D., and Löhner, R., "A hybrid Cartesian grid and gridless method for compressible flows," *Journal of Computational Physics*, Vol. 214, No. 2, 2006, pp. 618–632.

[17] Kamatsuchi, T., "Turbulent Flow Simulation around Complex Geometries with Cartesian Grid Method," AIAA Paper 2007-1459, 45th AIAA Aerodpace Sciences Meeting and Exhibit, Reno, NV, USA, 8-11 Jan. 2007.

[18] Katz, A., Jameson, A., and M, W. A., "A Multi-Solver Scheme for Viscous Flows Using Adaptive Cartesian Grids and Meshless Grid Communication," AIAA Paper 2009-768, 47th AIAA Aerodpace Sciences Meeting and Exhibit, Reno, NV, USA, 3-8 Jan. 2009.

[19] Paige, C. C. and Saunders, M. A., "LSQR: An Algorithm for Sparse Linear Equations and Sparse Least Squares," *ACM Transactions on Mathematical Software*, Vol. 8, No. 1, 1982, pp. 43–71.

[20] 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.