48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition
4 - 7 January 2010, Orlando, Florida

AIAA 2010-1269

An Edge-Averaged Semi-meshless Framework for Numerical Solution of Conservation Laws

Edmond Kwan-yu Chiu* and Antony Jameson†
Stanford University, Stanford, CA 94305, USA

Despite the continuous improvement of technology, generation of high-quality meshes around complicated geometry remains a time consuming and even laborious task. In this paper, we propose a semi-meshless computational framework designed to reduce the difficulty in generating the necessary point connectivity for numerically solving conservation laws. The new framework is based on an analytic formulation of conservation laws in a generalized coordinate frame, in which the time derivatives of the conservative variables can be computed using a weighted average of estimates from local stencils. This formulation naturally implies a merit function for generating stencils based on point distances and the orthogonality of the generalized coordinates in the expected stencils. We applied the framework to solving the Euler equations for inviscid flow. Test cases involving airfoils showed that the new computational framework produced numerical solutions with quality similar to their finite volume counterparts.

I. Introduction

Efficient generation of high quality meshes has been a subject of continued interest in various scientific communities. Even with the abundance of available tools and technology, mesh generation often remains the bottleneck in the analysis phase of an engineering design cycle, especially when the designed system involves complicated geometry. In the last decade, many have attempted to circumvent mesh generation by developing meshless algorithms. This led to different classes of meshless methods, a general one of which solves the governing equations in a finite-difference setting using various local interpolation methods to obtain derivatives. For example, Batina,[1] Löhner et al.,[2] and Oñate et al.,[3] used polynomial basis in their work. Sridar and Balakrishan,[4] and Katz and Jameson[5] employed Taylor series. Kansa,[6] Shu et al.[7] and Tota and Wang[8] applied radial basis functions for interpolation.

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. Kirshnman and Liu,[9] Koh et al.[10] and Luo et al.[11] coupled meshless schemes near geometry surfaces with Cartesian grid methods in the far field. Kamatsuchi[12] performed viscous calculations using a meshless method in a near-wall subgrid that supplemented sub-divided Cartesian meshes through immersed boundary methods. Katz et al.[13] also used a meshless method as an interface between two different meshes.

Even though the work mentioned above has demonstrated the potential of meshless methods, there still seems to be doubt on wider application of meshless algorithms. A valid concern is that, even though meshless methods does not require rigid connecitivity in the form of a grid, they still require initial point distributions and ways to generate point clouds from these distributions. To form point clouds, Oñate et al.[3] used a quadrant search technique, while Löhner et al.[2] used an octree search followed by local Delauney triangulation. Both methods further prune clouds to avoid ill-conditioned local least squares matrices. This highlights that, even though meshless connectivity have much less stringent reciprocity requirements, meshless methods face other challenges in preprocessing, particularly in ensuring the quality of the meshless clouds.

*Ph. D. Candidate, Department of Aeronautics and Astronautics, William F. Durand Building, 496 Lomita Mall, Stanford, CA 94305-4035; 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

Few have considered intermediate approaches to generate full connectivity from some limited initial connectivity. Meakin et al.[14] did so in their work on strand grids. Even though their work better fits into the category of mesh generation, it did demonstrate that, if one can generate such initial connectivity at low costs and make clever use of its (albeit limited) structure, this kind of intermediate approach may lead to overall gains in preprocessing efficiency.

In this paper, we propose and validate an intermediate framework that, i) like meshless algorithms, relaxes the requirements on the necessary connectivity for numerically solving conservation laws, ii) discretizes a domain with less initial connectivity information than that provided by traditional meshes, but iii) by using slightly more initial connectivity information than a pure meshless method, suggests a natural quality estimate of the final connectivity that can help reduce quality tests during preprocessing. Current results suggest that this framework can incorporate traditional spatial discretization schemes to produce solutions comparable to those obtained using the same spatial discretization with traditional meshes.

In section II and III, we present the formulation of the framework along with some properties implied by this analytic formulation. In section IV, we detail the algorithm used to produce stencils for approximating spatial derivatives. Section V contains results from applying the current framework to computing steady-state solutions for inviscid compressible flow over airfoils.

II. Formulation

II.A. Motivation

Consider a system of conservation laws in 2-D

$$\frac{\partial \mathbf{w}}{\partial t} + \nabla \cdot \mathbf{f} = 0 \tag{1}$$

One way to advance (in time) the numerical solutions to the discrete version of system (1) is to use the discrete divergence of the spatial flux as an estimate of the time derivative at each solution point of interest.

The use of finite difference schemes on structured curvilinear meshes essentially solves eqn. (1) in a curvilinear coordinate frame. For example, in 2-D, we have

$$\frac{\partial \mathbf{w}'}{\partial t} + \frac{\partial \mathbf{F}}{\partial \xi} + \frac{\partial \mathbf{G}}{\partial \eta} = 0 \tag{2}$$

where (ξ, η) represent the curvilinear coordinates, and $\mathbf{w}'$, $\mathbf{F}$, $\mathbf{G}$ are, respectively, the transformed conservative variables and the generalized fluxes along ξ and η directions. In this case, the mesh metric terms implicitly define the tranformation from the Cartesian to curvilinear coordinates in which the flux divergence is numerically calculated.

On the contrary, many meshless schemes operate on the directly discretized form of

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y} = 0 \tag{3}$$

which is eqn.(1) in Cartesian coordinates. As mentioned before, these schemes approximate the flux derivatives using different local collocation techniques.

In the current framework, we take an intermediate approach that is equivalent to numerically solving the governing equations at each solution point in a *local* transformed coordinate frame. In the formulation, we start from a semi-mesh that defines the location of each solution point, its local η-direction and local connectivity along this direction. To compute flux derivative estimates at each solution point, we also assume that we find neighbor points to complete the definition of the local transformed coordinate frame using the locations of these neighbor points relative to the solution point. Section IV contains an example of such algorithms for obtaining candidate points.

II.B. Detailed Formulation

Because of the focus of our present work on computing external flow around objects, we define the semi-meshes as a set of lines eminating from the object's surfaces. All solution points are distributed on the lines. At each solution point, we define η to be the local tangential direction of the line. The η-connectivity of each

point is then its neighbors on the line. However, the resulting mathematical formulation does not preclude the use of other definitions of semi-meshes, as long as they possess the local connectivity information as stated in section II.A.

Consider a point a_i on a mesh line A. Let us denote the local coordinate along A at a_i by η_{a_i}. If the coordinate direction ξ_{a_i} between a_i and an out-of-line neighbor point is linearly independent of η_{a_i}, the two directions completely define the local transformed coordinates. Then, as long as we can use these solution points along the two coordinate directions to form a stencil, we can approximate derivatives of smooth functions along these directions. In other words, we can use the stencil points to obtain a local approximation of eqn.(1) in 2-D at point a_i, in the local coordinate frame (ξ_{a_i}, η_{a_i}):

$$\frac{\partial \mathbf{w'}_{a_i}}{\partial t} + \frac{\partial \mathbf{F}}{\partial \xi_{a_i}}\Big|_{a_i} + \frac{\partial \mathbf{G}}{\partial \eta_{a_i}}\Big|_{a_i} = 0 \tag{4}$$

where

$$\begin{aligned}
\mathbf{w'}_{a_i} &= J_{xya_i}\mathbf{w} \\
J_{xya_i} &= \frac{\partial x}{\partial \xi_{a_i}}\frac{\partial y}{\partial \eta_{a_i}} - \frac{\partial x}{\partial \eta_{a_i}}\frac{\partial y}{\partial \xi_{a_i}} \\
\mathbf{F} &= \mathbf{f}\frac{\partial y}{\partial \eta_{a_i}} - \mathbf{g}\frac{\partial x}{\partial \eta_{a_i}} \\
\mathbf{G} &= -\mathbf{f}\frac{\partial y}{\partial \xi_{a_i}} + \mathbf{g}\frac{\partial x}{\partial \xi_{a_i}}
\end{aligned}$$

Because the transformed coordinates are local in nature, this formulation relaxes the requirement of one-to-one reciprocity in traditional meshes. To allow even more flexibility, we recognize that although the mesh lines define the direction η_{a_i}, there may be more than one suitable definition of ξ_{a_i} from different neighbor points around a_i. Let S_{a_i} be a set of points around a_i that contains suitable neighbor points to form n stencils. Then, evaluating all of the terms in (4), we can write (4) for each stencil $1 \le k \le n$ and obtain

$$\frac{\partial}{\partial t}\Big(J_{xya_ik}\mathbf{w}_{a_i}\Big) + \Big[\frac{\partial}{\partial \xi_{a_ik}}\Big(\mathbf{f}\frac{\partial y}{\partial \eta} - \mathbf{g}\frac{\partial x}{\partial \eta}\Big)\Big]_{a_i} + \Big[\frac{\partial}{\partial \eta}\Big(-\mathbf{f}\frac{\partial y}{\partial \xi_{a_ik}} + \mathbf{g}\frac{\partial x}{\partial \xi_{a_ik}}\Big)\Big]_{a_i} = 0 \quad \forall\ 1 \le k \le n \tag{5}$$

where "a_ik" represents the evaluation of a quantity at point a_i using information from stencil k.

The last two terms on the left hand side of each of the above n equations then provide a different estimate of the spatial derivatives for each stencil. We denote this estimate by $\mathbf{R}_{a_ik}$, i.e.

$$\mathbf{R}_{a_ik} = -\Big\{\Big[\frac{\partial}{\partial \xi_{a_ik}}\Big(\mathbf{f}\frac{\partial y}{\partial \eta_{a_i}} - \mathbf{g}\frac{\partial x}{\partial \eta_{a_i}}\Big)\Big]_{a_i} + \Big[\frac{\partial}{\partial \eta}\Big(-\mathbf{f}\frac{\partial y}{\partial \xi_{a_ik}} + \mathbf{g}\frac{\partial x}{\partial \xi_{a_ik}}\Big)\Big]_{a_i}\Big\} \tag{6}$$

With stationary solution points, we can move the quantity J_{xya_ik} outside the time derivative. In that case, a natural way to obtain $\frac{\partial \mathbf{w}_{a_i}}{\partial t}$ from all the residuals $\mathbf{R}_{a_ik}$ is to determine $\frac{\partial w_{j_{a_i}}}{\partial t}$ by solving the following least squares problem for each conservative variable w_j:

$$\min \sum_{k=1}^{n} \Big\{\sqrt{\theta_{a_ik}}\Big[\Big(J_{xya_ik}\frac{\partial w_{j_{a_i}}}{\partial t}\Big) - R_{j_{a_ik}}\Big]\Big\}^2 \tag{7}$$

where θ_{a_ik} is the least squares weight for the k-th residual. Constructing each of the normal equations from (7), we can recombine the equations in vector form and write the vector normal equation of (7) as

$$[\sum_k \theta_{ik}(J_{xya_ik})^2]\frac{\partial \mathbf{w}_{a_i}}{\partial t} = \sum_k \theta_{ik}(J_{xya_ik})\mathbf{R}_{a_ik} \tag{8}$$

from which we can obtain the time derivative

$$\frac{\partial \mathbf{w}_{a_i}}{\partial t} = \frac{\sum_k \theta_{a_ik}(J_{xya_ik})\mathbf{R}_{a_ik}}{[\sum_k \theta_{a_ik}(J_{xya_ik})^2]} \tag{9}$$

There are two levels of generalizations one can make to the above derivation. Firstly, as we have mentioned, the derivation of eqn.(6) does not require the use of lines from surfaces as semi-meshes. Equation (6) should accommodate other means of defining local coordinates at each solution point. Secondly, although we arrived at eqn.(9) using finite-difference like notations, the idea of estimating the time derivative by the least squares average of spatial residuals does not limit the choice of final spatial discretization for calculating these residuals. Therefore, this part of the formulation can be applied with other spatial discretization schemes for which one can generate local stencils. Also, the time derivative estimate in (9) may serve as a single-stage value in a multi-stage time-stepping scheme, as we do in section V.

Because the time derivative is a least squares average of values from different stencils, each of which can be viewed as a collection of edges between the solution point and neighbor points, we refer to this method as an edge-averaged semi-meshless (EAS) method. In the following section, we will discuss some properties of eqn.(9) and our choice of the weights $\theta_{a_i k}$ based on these properties.

III. Properties and Weight Selection

Equation (9) possesses a few interesting properties. We have already seen that it provides us with freedom in choosing the final discretization for the system of conservation laws. Here, we will focus on a property that automatically puts emphasis on more orthogonal local coordinate bases. We also discuss the property's implication on the choice of desirable weights for the least squares problem.

III.A. Inherent Emphasis on Stencil Orthogonality

In eqn.(9), in addition to its weighting by the least-square weights $\theta_{a_i k}$, the residual $\mathbf{R}_{a_i k}$ is automatically weighted by the coordinate Jacobian

$$J_{xya_i k} = \frac{\partial x}{\partial \xi_{a_i k}} \frac{\partial y}{\partial \eta_{a_i}} - \frac{\partial x}{\partial \eta_{a_i}} \frac{\partial y}{\partial \xi_{a_i k}} \tag{10}$$

$J_{xya_i k}$ vanishes if the unit vectors along $\xi_{a_i k}$ and η_{a_i} are parallel. Therefore, the current framework automatically weighs the k-th residual estimate in a way proportional to the orthogonality of the k-th local coordinate basis, independent of the least squares weights. In other words, all other things being equal, the framework prefers to compute the time derivative using spatial information from the most orthogonal bases locally available.

In a finite volume mesh consisting of quadrilaterals, $J_{xya_i k}$ represents the volume of a cell. In particular, $J_{xya_i k} = 0$ when a cell is degenerate. In cases involving complicated geometries, obtaining high-quality meshes with the constraint of reciprocal connectivity between points is often difficult. From above, one can see that the current framework ignores the spatial information from these "degenerate cells". Thus, by relaxing the requirement to find one-to-one correspondences in all coordinate directions, the current framework introduces the possibility to locally obtain better stencils that are equivalent to less-skewed mesh cells.

The metric terms also influence the strategy to select the least squre weights. Because the magnitudes of the metric terms increase with the representative spacing between stencil points, so does $J_{xya_i k}$. Therefore, given the same orthogonality, stencil points further away from a_i (i.e. coarser stencils) actually receive heavier weights. We will now detail how one can selecting appropriate weights $\theta_{a_i k}$ to avoid this problem, while preserving the order of accuracy of the underlying spatial estimation.

III.B. Weight Selection

As mentioned in section III.A, the magnitude of $J_{xya_i k}$ increases with the representative stencil length h_k, leading to over-emphasis of coarse stencils. At first glance, setting $\theta_{a_i k} = (J_{xya_i k})^{-\alpha}$, where $\alpha \geq 1$, will negate the undesirable contribution from $J_{xya_i k}$. Unfortunately, the product $\theta_{a_i k} J_{xya_i k}$ in eqn.(9) also means that $\alpha \geq 1$ eliminates the orthogonality prioritization property, making this approach undesirable.

Instead, we proceed by observing that $J_{xya_i k} \sim h_k^{n_d}$, where n_d is the spatial dimension of the problem and h_k is a representative length of the k-th stencil independent of the stencil orthogonality. Thus, a reasonable way to restore the emphasis on stencils involving close-by points is to set

$$\theta_{a_i k} = h_k^{-q}, \qquad q \geq n_d \tag{11}$$

$$h_k = \frac{1}{n_b} \sum_{j=1}^{n_b} \left\| \mathbf{x}_{kb_j} - \mathbf{x}_{a_i} \right\|_2 \tag{12}$$

where n_b is the number of points in the k-th stencil that does not belong to the meshline A (for example, with a 4-point stencil for central differencing in both ξ and η directions, $n_b = 2$), and $\mathbf{x}_{kb_j}$ is the coordinates of the j-th cloud point in the k-th stencil.

One should also consider, however, the resulting effective accuracy of the spatial discretization due to the combination of estimates obtained using multiple stencils. Consider a p-th order spatial discretization. We can express the spatial residual estimate $R_{j_{a_i k}}$ (defined by eqn.(6)) as

$$R_{j_{a_i k}} = R_j + K_p h_k^p + O(h_k^{p+1}) \tag{13}$$

where R_j is the exact spatial residual for the j-th conservative variable and K_p is the leading error constant from the Taylor's expansion of the scheme. Often, K_p does not depend on h. Therefore, combining eqns. (8) and (13), we have

$$\begin{aligned} \frac{1}{\sum_k [\theta_{a_i k} (J_{xya_i k})^2]} \frac{\partial \mathbf{w}_{a_i}}{\partial t} &= \sum_k \left[\theta_{a_i k} J_{xya_i k} (R + K_p h_k^p) \right] \\ &= \sum_k \left(\theta_{a_i k} J_{xya_i k} \right) R + \sum_k \left(\theta_{a_i k} J_{xya_i k} {h_k}^p \right) K_p \end{aligned} \tag{14}$$

With eqn.(11), and $J_{xya_i k} \sim h_k^{n_d}$, the summation in the second term of eqn.(14) becomes

$$\sum_k \left(\theta_{ik} J_{xya_i k} {h_k}^p \right) \sim \sum_k \left({h_k}^{-q} {h_k}^{n_d} {h_k}^p \right) = \sum_k O(h_k^{p+n_d-q}) \tag{15}$$

A reasonable assumption is that the size of h_k does not vary dramatically at a point for moderate values of n (number of stencils). Therefore, to preserve the spatial accuracy, we need to choose

$$\theta_{a_i k} = h_k^{-q}, \qquad q \le n_d \tag{16}$$

Combining eqns.(11) and (16), we have

$$\theta_{a_i k} = h_k^{-q}, \qquad q = n_d \tag{17}$$

Thus, we have shown in this section that, given a set of neighbor points around a solution point a_i and suitably chosen least square weights, the proposed semimeshless scheme locally estimates the time derivatives using spatial information from neighboring stencil points, taking into account the distance and orthogonality of the stencil and maintaining the order of accuracy of the spatial discretization. In the next section, we will illustrate how to select the set of neighbor points by a local cloud search procedure.

IV. Stencil (Edge) Generation

In this section, we present a procedure to generate the computation stencils in the current framework. For better illustration, we show the specific steps for generating stencils for a 2-D finite-volume discretization that is second-order accurate in smooth regions.[15] We use the same discretization in our test cases in section V. Our cloud search procedure bears some similarities to that by Oñate *et al.*[3] in the sense that it tries to ensure some level of orthogonality between point pairs in the stencil. However, in our case, the consideration of orthogonality results directly from the formulation of our computational framework.

The outline of the procedure is as follows: After generating the mesh lines with points, we first perform a cloud search based on local line spacing to locate possible candidate neighbors for forming stencils. Then, a merit function ranks the points based on neighbor distances and orthogonality. Finally, the best points are grouped together to form the stencils.

IV.A. Cloud Search

Given n_{max}, the desired number of stencils at each point a_i, and cloud search parameters $n_{p,min}$ and $n_{p,max}$ that bound the preprocessing cloud size, we first find all the points that are within a specified radius from a_i, i.e.,

$$S_{a_i} = \{ b_j \mid \left\| \mathbf{x}_{b_j} - \mathbf{x}_{a_i} \right\|_2 < r_{a_i},\ b_j \notin A \} \tag{18}$$

where r_{a_i} is the local search radius defined as a multiple of the representative point spacing along the line at a_i. In this step, we filter near-boundary points in a way similar to that adopted by Löhner *et al.*[2]

After forming S_{a_i}, we rank the points in S_{a_i} using a merit function ψ. we design ψ to bear some similarities to the least squares weights $\theta_{a_ik}J_{xya_ik}$ so that we can expect that the stencils formed by the neighbor points with the best merits will also receive the heaviest weights in the semi-meshless formulation. In this work, we use the merit function

$$\psi = \frac{(1 - |\hat{\xi}_{a_ib_j} \cdot \hat{\eta}_{a_i}|)^\alpha}{\|\mathbf{x}_{b_j} - \mathbf{x}_{a_i}\|_2^\beta} \tag{19}$$

where $\hat{\eta}_{a_i}$ is the unit vector along line A at a_i, $\hat{\xi}_{a_ib_j}$ is the unit vector along $\mathbf{x}_{b_j} - \mathbf{x}_{a_i}$ and α and β are user-chosen coefficients for balancing the contribution from orthogonality (the numerator) and distance (the denominator) during the cloud search. In the present work, $\alpha = 1$ and $\beta = q = n_d = 2$. With these parameters, ψ behaves like a product of the stencil's normalized volume and its expected weight and serves as an estimate of the quality of the resulting stencil.

We repeat the search with increasing radius r_{a_i} until either S_a contains at least $n_{p,min}$ out-of-line neighbor points or the algorithm has performed n_{search} iterations. If the cloud already contains more than $n_{p,max}$ points before n_{search} iterations, the algorithm retains the $n_{p,max}$ points with highest values of ψ and terminates. As we saw in section III, EAS favors orthogonality in stencil bases (ξ_{a_ik}, η_{a_i}). Therefore, the user can set an optional preprocessing minimum threshold on the included angle of (ξ_{a_ik}, η_{a_i}) pairs to speed up the cloud search.

IV.B. Stencil Assignment

The stencil assigment is specific to the chosen spatial discretization scheme. For the current finite volume scheme, we set the local stencil for a point to include its immediate neighbors on the line and an out-of-line point pair separated by the line.

We separate S_{a_i} into

$$\begin{aligned} S_{a_i}^+ &= \{b_j \in S_{a_i} \mid \xi_{a_ib_j} \cdot \zeta_{a_i} \geq 0\} \\ S_{a_i}^- &= \{c_l \in S_{a_i} \mid \xi_{a_ic_l} \cdot \zeta_{a_i} < 0\} \end{aligned} \tag{20}$$

where ζ_{a_i} is simply the unit vector in the out-of-plane direction. $S_{a_i}^+$ and $S_{a_i}^-$ thus contains points in S_{a_i} that belongs to each of the respective left and right half planes created by bisecting $\mathbb{R}^2$ by a the line tangent to A at a_i. Pairing up the k-th highest-ranked points in each of the half planes, we obtain the k-th stencil, which consists of the points $\{a_i, a_{i+1}, a_{i-1}, b_{j_k}, c_{l_k}\}$. For efficiency, we ensure that there are enough points on both sides of line A during the cloud search in our implementation.

With the definition of stencil points, we can then calculate θ_{a_ik} and J_{xya_ik} from the location of the stencil points. Figure 1 provides a high-level illustration of the above procedure.

V. Application to Solving the Euler Equations

To test the current framework in a practical scenario, we apply it to obtain steady-state numerical solutions to the 2-D Euler equations

$$\frac{\partial \mathbf{w}}{\partial t} + \frac{\partial \mathbf{f}}{\partial x} + \frac{\partial \mathbf{g}}{\partial y} = 0 \tag{21}$$

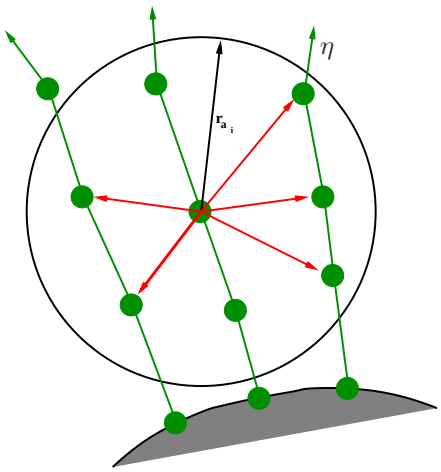
with

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

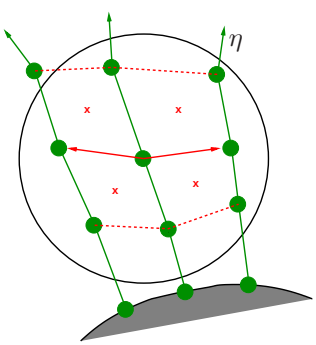
and

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

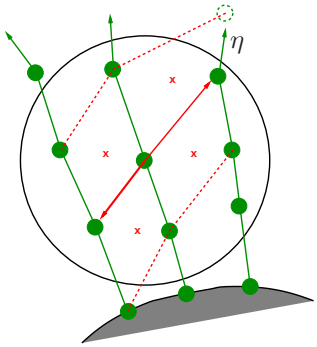
where ρ, u, v, P, E, and H are, respectively, the density, x- and y-components of the flow velocity, pressure, total energy and total enthalpy. We compute flow solutions around airfoils in subsonic and transonic conditions and compare the convergence and accuracy of these solutions with well-validated numerical solutions.



(a) Cloud search around selected point with radius r_{a_i}



(b) Stencil 1



(c) Stencil 2

Figure 1: Cloud search of the semimeshless framework forming 2 stencils around a point.

V.A. Discretization

As mentioned in IV.B, the k-th stencil at point a_i is a quadrilateral formed around a_i with its respective edge midpoints at the midpoint between a_i and each point in the set $\{a_{i-1}, a_{i+1}, b_{j\,k}, c_{l\,k}\}$, where $b_{j\,k}$ and $c_{l\,k}$ is the out-of-line point pair generated by the algorithm described in section IV. Contrary to a traditional finite-volume mesh, overlaps and gaps may exist between these quadrilaterals. Therefore, there is no *a priori* guarantee that the final discretization will be conservative, although the cloud search may generate connectivity that leads to global conservation.

Each computational stencil at a point produces an estimate of the spatial flux derivatives based on the spatial discretization. In the current cases, we compute this estimate using a procedure very similar to the one previously used by Jameson, Schmidt and Turkel[15] to discretize the Euler Equations on quadrilateral meshes.

In each stencil cell, we obtain the spatial flux derivative estimate by an approximation to the integral $\int_V (\mathbf{f}dy - \mathbf{g}dx)$. For example, the x-momentum flux computed using the k-th stencil of each node is

$$R_{2k} = \sum_{m_k=1}^{n_{ek}} \phi_{2\,m_k} = \sum_{m_k=1}^{n_{ek}} (Q_{m_k}\rho u_{m_k} + \Delta y_{m_k} P_{m_k}) \tag{22}$$

where n_{ek} is the number of edges in the k-th stencil cell, ϕ_{m_k} is the directed flux vector along edge m_k, $Q_{m_k} = \Delta y_{m_k} u_{m_k} - \Delta x_{m_k} v_{m_k}$ and $(\Delta x_{m_k}, \Delta y_{m_k})$ is the unnormalized vector along the length of the edge m_k of cell k. The quantities with subscripts "m_k" are averages of values at stencil points separated by edge m_k.

After obtaining the residuals for each stencil, the EAS solution procedure differs from that of a pure finite-volume method: We weigh the residuals from each of the k stencils according to eqn. (9).

As in the case of much previous work that used similar spatial discretizations, we introduce necessary aritificial dissipation to achieve stability. We employ the H-CUSP scheme,[16] which admits constant enthalpy in steady flow, combined with symmetric limited positive (SLIP) reconstruction[16] to retain higher order of accuracy in regions with smooth solutions. Together, these schemes replace the contribution to the spatial residual from each stencil edge by

$$\tilde{\phi}_{m_k} = \phi_{m_k} - \mathbf{d}_{m_k} \tag{23}$$

where

$$\mathbf{d}_{m_k} = \frac{1}{2}\alpha^* c(\mathbf{w}_{m_{k\,R}} - \mathbf{w}_{m_{k\,L}}) + \frac{1}{2}\beta(\phi_{m_{k\,R}} - \phi_{m_{k\,L}}) \tag{24}$$

where the subscipts "L" and "R" denotes limited reconstructed left and right states on both sides of the edge, and α^* and β denotes computed coefficients[16] based on the average state.

We march the solution to a steady state using modified Runge-Kutta schemes[17] with larger stability regions, along with convergence acceleration techniques such as local time stepping, residual smoothing[18] and enthalpy damping[15] on the total residual.

V.B. Results

We now present results from a number of test cases involving flow around airfoils. We compare the results obtained using the EAS solver to those obtained using FLO82, a cell-centered finite volume solver using structured O-meshes. To evaluate the connectivity generation procedure, both solvers use the same point distributions. We use the connectivity of the cell centers in the wall-normal direction from FLO82 as the initial semi-mesh for the EAS solver.

V.B.1. Connectivity Generation Results

In the previous sections, we suggested that the stencil generation procedure does not guarantee global conservation at the discrete level, even with a conservative underlying spatial discretization. However, when we select the best stencil at each point, the algorithm did globally produce reciprocal connectivity. One can partly attribute this to the good quality of the initial point distribution. Nonetheless, it is still encouraging that the current algorithm can potentially preserve conservativeness of the underlying spatial discretization.

To futher test our algorithm, we initiated the cloud search for points around a NACA 0012 airfoil with a semi-mesh that contains a mesh line emanating from the trailing edge. In the region near the trailing edge, the algorithm did locally form stencils that do not allow reciprocity but possess better orthogonality than those composed of immediate neighbors in the structured mesh. In other regions of the domain, the resulting stencils retained reciprocity. One can view these results as the willingness of the current algorithm to sacrifice conservativeness near geometric singularities where mesh generation often becomes difficult.

Figure 2 shows the local cells generated from stencils near the trailing edge of the NACA 0012 airfoil when we applied the current algorithm to both semi-meshes.

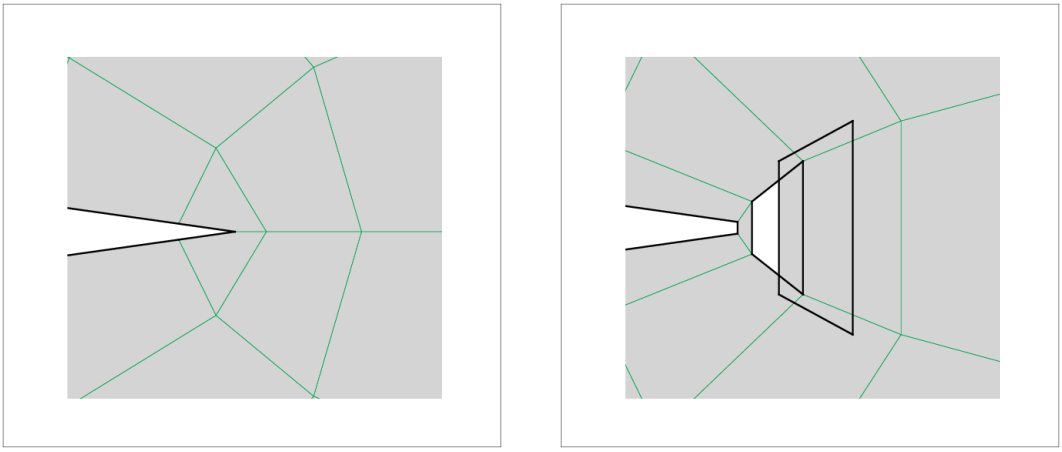
V.B.2. Flow Solutions

Table 1 shows the drag coefficient of a NACA 0012 airfoil computed at $M = 0.3$, $\alpha = 0^\circ$, using the best stencil obtained at each point on the initial semi-mesh.

Surface Points	c_d	
(# Mesh Lines)	EAS	FLO82
40	0.0039	0.0025
80	0.0004	0.0005
160	0.0001	0.0001

Table 1: Drag coefficients for flow over NACA 0012, $M = 0.3$, $\alpha = 0^\circ$

With one stencil at each point, the semi-meshless algorithm reduces to a locally finite volume solution procedure. As expected, results from the current semi-meshless method compare very well with their finite



(a) with no mesh lines from trailing edge (b) with a mesh line from trailing edge

Figure 2: Domain around a NACA 0012 airfoil discretized using the current algorithm, showing the areas covered by final cells (grey), reciprocal connectivity (green) and resulting domain boundaries (black).

volume counterparts for reasonably dense point distributions. The drag convergence also suggests that the discretization is effectively second-order accurate, same as that of the pure finite volume method.

Table 2 lists the lift and drag coefficients computed in other test cases, in which the semi-meshless solver also used the best stencil available at each solution point on initial semi-meshes with 160 mesh lines.

Airfoil	M	α	c_l		c_d	
			EAS	FLO82	EAS	FLO82
NACA 0012	0.3	3.0	0.3822	0.3824	0.0001	0.0001
NACA 0012	0.75	1.0	0.2256	0.2260	0.0026	0.0026
RAE 2822	0.75	3.0	1.1433	1.1452	0.0487	0.0488

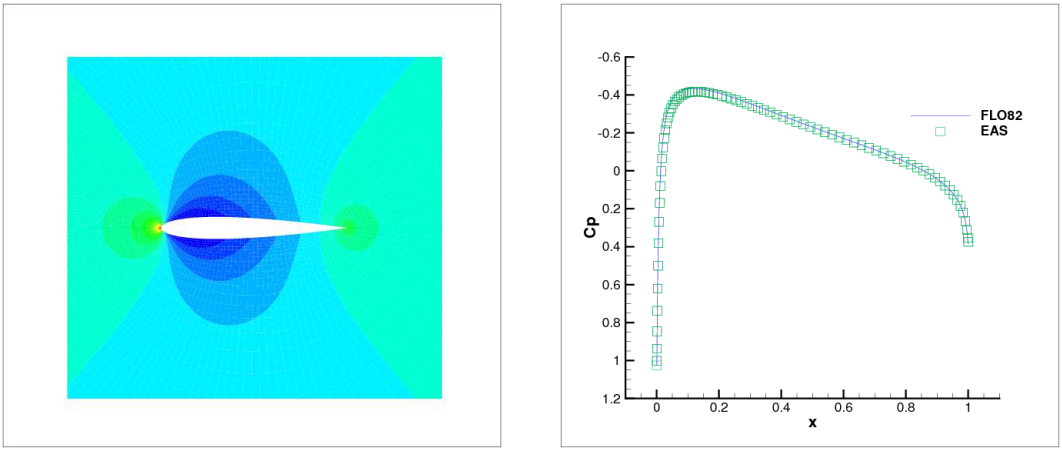
Table 2: Numerical results from subsonic and transonic lifting test cases

One can see that the lift coefficients obtained using the current framework are practically identical to those obtained using FLO82. Figures 3 to 5 show the excellent agreement between the semi-meshless and the finite volume solutions. One may expect this excellent agreement from the fact that the semi-meshless algorithm generated local cells that are very similar to those in the original finite volume mesh. However, we found that the non-reciprocal connectivity in section V.A, which allowed overlaps and gaps in the domain, still resulted in flow solutions comparable to those in the cases with fully reciprocal connectivity.

We also briefly investigated the effects of using more than one local stencil per point in the final discretization. Initial tests performed using the current weighting system and cloud search procedure suggest that increasing the number of stencils per point may improve convergence through the increased support provided by the extra stencils, but in general does not have a positive effect on the stability of the algorithm. This may have resulted from the less superior quality of the additional stencils obtained from these particular semi-meshes. Potential future work in developing alternative weighting systems, cloud search or point generation procedures may address this issue by improving the robustness of the semi-meshless framework.

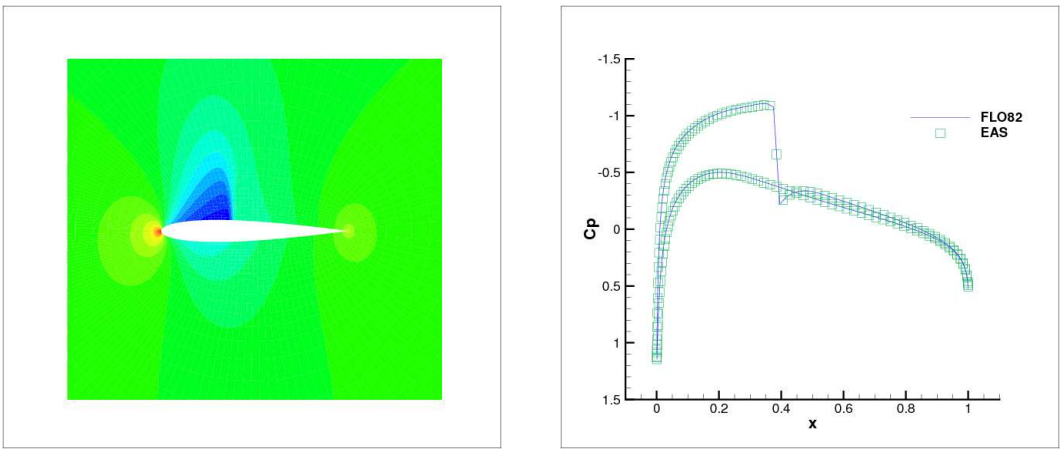
VI. Conclusion

We proposed a new semi-meshless computational framework that relaxes the connectivity generation requirements that often make mesh generation difficult. In 2-D, the new framework generates a user-defined number of stencils by using a cloud search to supplement predefined partial connectivity in one local spatial dimension. Based on local coordinate frames, the formulation considers the distance as well as the orthogonality between points when generating stencils. A general least-squares type formulation for calculating the



(a) pressure contours (b) surface pressure coefficient

Figure 3: Flow over NACA 0012, $M = 0.3$, $\alpha = 0^\circ$



(a) pressure contours (b) surface pressure coefficient

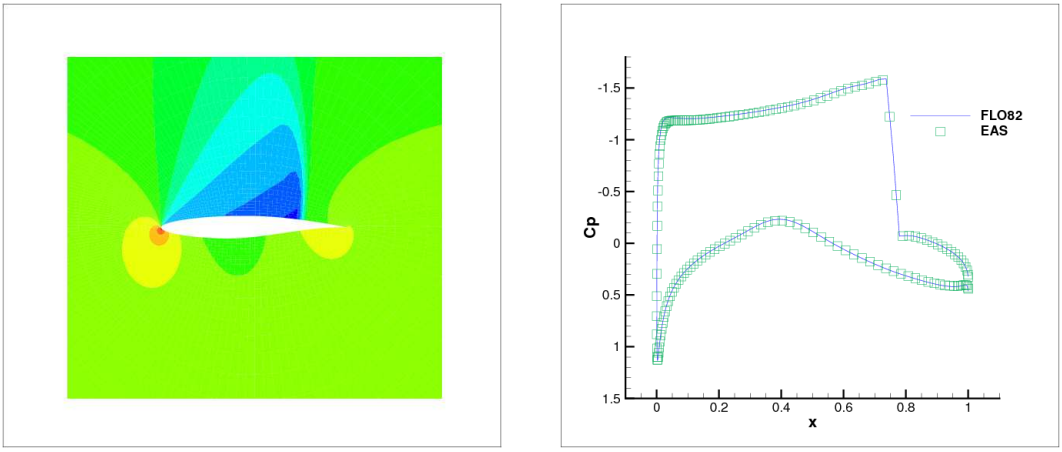
Figure 4: Flow over NACA 0012, $M = 0.75$, $\alpha = 1^\circ$

time derivatives using the stencils provides freedom in the choice of spatial discretization schemes and the number of local stencils.

When equipped with underlyling finite volume spatial discretization, the semi-meshless algorithm generated local stencils very similar to those in existing finite volume meshes. Even though the formulation does not guarantee fully reciprocal connectivity, we observed such connectivity in practice when the algorithm selects the best stencil at each point in reasonable initial point distributions. In tests cases involving inviscid compressible flow over airfoils, the semi-meshless algorithm also produced results comparable to finite volume solutions. These results encourage future work in the development of weighting systems and connectivity generation techniques to improve the robustness of the algorithm.

VII. Acknowledgements

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



(a) pressure contours (b) surface pressure coefficient

Figure 5: Flow over RAE 2822, $M = 0.75$, $\alpha = 3^\circ$

References

[1]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.

[2]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.

[3]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.

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

[5]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.

[6]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.

[7]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.

[8]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.

[9]Kirshman, D. and Liu, F., "Gridless Boundary Condition Treatment for a Non-Body Conforming Mesh," AIAA Paper 2002-3285 32nd AIAA Fluid Dynamics Conference and Exhibit, St. Louis, MO, USA, 24-26 June 2002.

[10]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.

[11]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.

[12]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.

[13]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.

[14]Meakin, R. L., Wissink, A. M., Chan, W. M., Pandya, S. A., and Sitaraman, J., "On Strand Grids for Complex Flows," AIAA Paper 2007-3834, 18th AIAA Computational Fluid Dynamics Conference Miami, FL, USA, 25-28 June 2007.

[15]Jameson, A., Schmidt, W., and Turkel, E., "Numerical Solution of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time Stepping Schemes," AIAA Paper 1981-1259, 14th Fluid and Plasma Dynamics Conference, Palo Alto, CA, USA, 23-25 June 1981.

[16]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.

[17]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.

[18]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.