19th AIAA Computational Fluid Dynamics
22 - 25 June 2009, San Antonio, Texas

AIAA 2009-3534

A Meshless Volume Scheme

Aaron Katz* and Antony Jameson†

Department of Aeronautics and Astronautics, Stanford University, Stanford, CA, 94305, USA

Development of a new meshless technique is described, called the meshless volume scheme. The method is completely meshless, yet retains the low storage and simple flux distribution technique characteristic of finite volume methods. The new method is based on the well-known Taylor series expansion method with least squares. A least squares weighting scheme is introduced that significantly reduces storage and computational requirements compared to other meshless schemes. The scheme is shown to be analogous to finite volume schemes in many ways, but still lacks a discrete telescoping property and is not discretely conservative. The method is applied to the Euler equations in two dimensions. Results are presented which agree well with established methods.

I. Introduction

DESPITE a wide variety of formulations, meshless methods in general have not achieved mainstream use for computational fluid dynamics (CFD). Most practical CFD algorithms are based on finite volume, finite difference, or finite element methods. To date, meshless methods have not truly shown the ability to relieve difficulties in mesh generation associated with mesh-based schemes. In fact, a major difficulty for meshless schemes is obtaining arbitrary clouds of points which lead to stable and accurate computations. Point cloud generation appears to be nearly as difficult as mesh generation itself for complex cases. While progress has been made by Löhner and Oñate[1] and Löhner, Sacco, and Oñate[2] to fill space with points, obtaining optimal point clouds remains difficult. Meshless methods also lack a discrete conservation property, are often more expensive, and often require more storage than traditional approaches. However, the persistent difficulties in mesh generation for complex configurations have left room for meshless schemes as a potential alternative in certain scenarios. This is the justification for this and other meshless work.

Many approaches to meshless discretization have been developed over the past three decades. Batina[3] showed the ability of a meshless method based on a polynomial basis and least squares to compute inviscid and viscous flows in two and three dimensions. A more detailed formulation of a finite point method for a wide variety of problems has been provided in the work of Oñate and others.[4–6] Löhner et al.[7] have applied the finite point method to compressible flow computations. Morinishi[8] assessed issues of accuracy and conservation of meshless solvers, as did Sridar and Balakrishnan.[9] Additionally, Balakrishnan and Praveen[10,11] have formulated an upwind least squares method for the Euler equations. In a different approach, Liu et al.[12] proposed a reproducing kernel particle method, an improvement over the smooth particle hydrodynamics method of Monaghan.[13] In yet a different approach, Belytschko et al.[14] devised an element-free Galerkin method, which was based on the weak forms characteristic of finite element methods. Duarte and Oden[15] have also developed the method of Hp clouds to solve boundary value problems.

Like many previous methods, the development in this paper is based on a least squares fitting of discrete nodal values to obtain the discretization of partial differential equations. However, our method differs from previous approaches in the way we perform the least squares weighting procedure. Instead of weighting the nodal contributions to the least squares gradient procedure based purely on proximity, we perform local constrained optimization procedures at each node to determine a new set of weights. These weights lead to simplified gradient computations, reducing storage and computational requirements as compared to other meshless schemes. We require half the storage in two dimensions and one third the storage in three dimensions as traditional least squares approaches. When applied to the Euler equations, we require half the number of flux computations in both two and three dimensions. The method requires similar storage

*PhD Candidate, presently post-doctoral Researcher, US Army Aeroflightdynamics Directorate (AMRDEC), AIAA Member.
†Thomas V. Jones Professor of Engineering, AIAA Member.

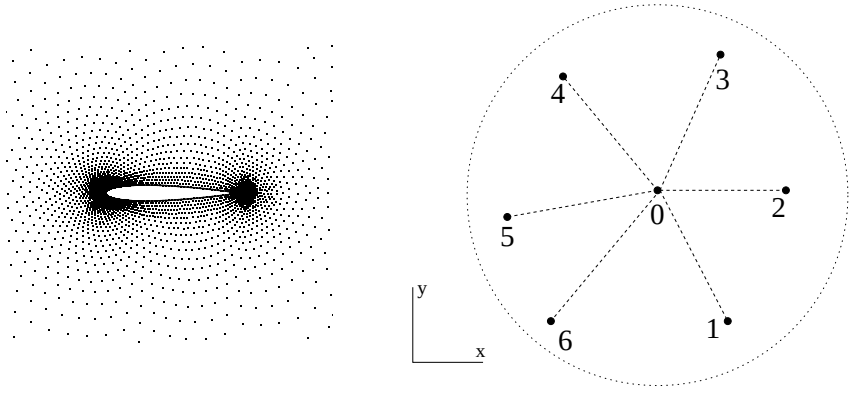
and computational requirements as conventional finite volume schemes. In fact, the implementation of our meshless method is quite similar to finite volume schemes, which is why we have chosen the name, "meshless volume," to describe our scheme.

As this paper is focused on the development of a new meshless agorithm, we do not address the important issue of point cloud generation. In fact, in this work we use meshes to establish local clouds of points for our simulations as a matter of convenience. However, in previous papers, we have implemented meshless schemes for which no underlying mesh was used to obtain the connectivity. For example, we showed that a meshless method could be used to connect overset grid systems in mesh interface regions.[16] Additionally, we have used a meshless method for convergence acceleration in a multigrid fashion, in which only clouds of points were used to represent coarse levels.[17] The meshless volume scheme could be implemented in these scenarios or other situations where a mesh is difficult to obtain. Furthermore, meshless techniques can also prove useful when meshes are available. For example, since the meshless volume scheme is essentially a gradient estimation method, it could be used for gradient reconstruction in finite volume schemes to reduce storage requirements over conventional least squares reconstruction. Consequently, the use of meshes to obtain connectivity in this context does not diminish the utility of the meshless volume scheme for many applications.

This work begins by exploring a commonly used meshless scheme based on local Taylor series expansions. Next, we show how we can eliminate some of the drawbacks of the Taylor approach with the use of a unique least squares weighting procedure. We then apply the new scheme to the Euler equations. Finally, we perform fundamental error studies and show two dimensional Euler results for well-known test cases.

II. Traditional Weighted Least Squares Gradient Procedure

Meshless discretizations of partial differential equations for CFD have revolved around the accurate computation of solution gradients given scattered grid data. One of the most common methods of gradient estimation is based on local Taylor series expansions and least squares. Such methods are not unique to meshless discretizations and have been applied in the context of finite volume gradient reconstruction.[18,19] The Taylor series method requires that for every point in the global domain to be solved, there exist a local cloud of points consisting of nearest neighbors. The global and local clouds of points are shown in Figure 1. As shown in Figure 1(b), for each node 0 in the global domain, a local cloud of n surrounding nodes



(a) Global point cloud. (b) Local point cloud.

Figure 1. Meshless discretization framework.

is identified, establishing the connectivity throughout the domain. In this work we obtain global and local clouds of points from meshes for convenience, but in other works we have obtained clouds with truly meshless techniques.[16,17]

In order to estimate the gradient of a function ϕ in a local cloud, the Taylor method requires that we expand a Taylor series with a given number of terms to each point in the local cloud. For example, a linear approximation leads to the following n equations for the gradients at node 0:

$$\phi_i = \phi_0 + \Delta \boldsymbol{r}_{0i} \cdot \nabla \hat{\phi}_0, \quad i = 1, \cdots, n, \tag{1}$$

where $\Delta \boldsymbol{r}_{0i} = \boldsymbol{r}_i - \boldsymbol{r}_0$, and $\nabla\hat{\phi}_0$ is the estimate of the gradient at node 0 for which we seek. The enforcement of the condition $n > 2$ leads to an overdetermined system for the gradients and the use of least squares techniques. The least squares problem involves finding the minimum of the function

$$f(\nabla\hat{\phi}_0) = \sum_{i=1}^{n} w_{0i}(\Delta \boldsymbol{r}_{0i} \cdot \nabla\hat{\phi}_0 - \Delta\phi_{0i})^2, \tag{2}$$

where $\Delta\phi_{0i} = \phi_i - \phi_0$, and w_{0i} is a weighting function designed to emphasize the contribution of certain points in the local cloud, usually based on proximity to node 0. A common weighting scheme is to weight the contribution of each node in the local cloud by the inverse square of its distance to node 0, with

$$w_{0i} = \frac{1}{|\Delta \boldsymbol{r}_{0i}|^2}. \tag{3}$$

The mimimization of $f(\nabla\hat{\phi}_0)$ leads to the normal equations, which may be expressed as

$$M\nabla\hat{\phi}_0 = P\Delta\boldsymbol{\phi}, \tag{4}$$

where $\Delta\boldsymbol{\phi}^T = \begin{bmatrix}\Delta\phi_{01} & \Delta\phi_{02} & \cdots & \Delta\phi_{0n}\end{bmatrix}$ is the known vector of undivided differences. For this linear case, the 2x2 matrix M and 2xn matrix P take the following forms:

$$M = \begin{bmatrix} \sum_i w_{0i}\Delta x_{0i}^2 & \sum_i w_{0i}\Delta x_{0i}\Delta y_{0i} \\ \sum_i w_{0i}\Delta x_{0i}\Delta y_{0i} & \sum_i w_{0i}\Delta y_{0i}^2 \end{bmatrix}, \quad P = \begin{bmatrix} w_{01}\Delta x_{01} & w_{02}\Delta x_{02} & \cdots & w_{0n}\Delta x_{0n} \\ w_{01}\Delta y_{01} & w_{02}\Delta y_{02} & \cdots & w_{0n}\Delta y_{0n} \end{bmatrix}.$$

The solution of Equation 4 involves the computation of the matrix M^{-1}, which is usually well-conditioned enough to obtain a solution directly from the normal equations. However, since M is only 2x2 for the two-dimensional linear least squares approximation, an analytical solution for the gradient estimates is easily obtained for this case, leading to partial derivative estimates which may be expressed in the following form:

$$\hat{\phi}_{0,x} = \sum_i a_{0i}\Delta\phi_{0i}, \quad \hat{\phi}_{0,y} = \sum_i b_{0i}\Delta\phi_{0i}, \tag{5}$$

where a_{0i} and b_{0i} are the derivative weight coefficients for i^{th} node in the local cloud centered around node 0, defined by

$$a_{0i} = \frac{w_{0i}\Delta x_{0i}\sum_k w_{0k}\Delta y_{0k}^2 - w_{0i}\Delta y_{0i}\sum_k w_{0k}\Delta x_{0k}y_{0k}}{\sum_k w_{0k}\Delta x_{0k}^2 \sum_k w_{0k}\Delta y_{0k}^2 - \left(\sum_k w_{0k}\Delta x_{0k}\Delta y_{0k}\right)^2} \tag{6}$$

$$b_{0i} = \frac{w_{0i}\Delta y_{0i}\sum_k w_{0k}\Delta x_{0k}^2 - w_{0i}\Delta x_{0i}\sum_k w_{0k}\Delta x_{0k}y_{0k}}{\sum_k w_{0k}\Delta x_{0k}^2 \sum_k w_{0k}\Delta y_{0k}^2 - \left(\sum_k w_{0k}\Delta x_{0k}\Delta y_{0k}\right)^2}. \tag{7}$$

Here, the index k runs over the nodes in the local cloud of node 0.

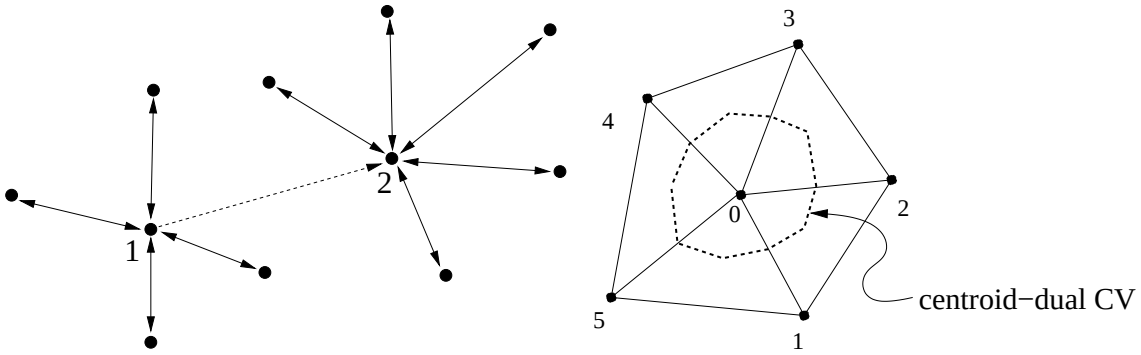
The weight coefficients, a_{0i} and b_{0i} are independent of ϕ and depend only on the nodal positions. As such, they are generally computed in a preprocess step for static point distributions. The partial derivative estimates of Equation 5 may be used to directly estimate derivative terms in partial differential equations for a meshless discretization. For example, the flux derivatives of the Euler equations may be discretized directly with the least squares approach. Alternatively, Equation 5 may be used to reconstruct solutions for linear preserving finite volume schemes. In either case, this method approximates the gradients of any function to first order accuracy in general, and improves to second order accuracy for certain regular point distributions.

In order to assess the properties of the Taylor series least squares approach applied to meshless schemes, it is helpful to compare it with a finite volume formulation when applied to a conservation law,

$$Q_{0,t} + \nabla \cdot \boldsymbol{F}_0 = 0, \tag{8}$$

which has integral form

$$V_0\bar{Q}_{0,t} + \int_{\partial V_0} \boldsymbol{F}(Q, \boldsymbol{n})dA = 0, \tag{9}$$



(a) Directed edge which breaks reciprocity.

(b) Centroid dual control volume on a triangular mesh.

Figure 2. Differences between meshless and finite volume discretizations.

where $\bar{Q}_0$ is the cell average value of Q in cell 0. A common finite volume scheme is formulated in a node-centered manner on a dual mesh, as shown in Figure 2(b). Such a scheme can be shown to be equivalent to a Galerkin weighted finite element discretization on linear triangular elements.[20] Following the derivation of Barth,[18] the finite volume discretization of Equation 9 is

$$V_0 Q_{0,t} + \sum_i \frac{1}{2} \left[\boldsymbol{F}(Q_i, dx_{0i}, dy_{0i}) + \boldsymbol{F}(Q_0, dx_{0i}, dy_{0i}) \right] - D_0^{FV} = 0, \tag{10}$$

where the geometric terms, (dx_{0i}, dy_{0i}), represent the dual area associated with edge $0i$, and D_0^{FV} represents diffusive terms arising from either artificial diffusion or upwind interpretations. Here we focus on the convective terms in the discretization by simply lumping all the diffusive contributions into D_0^{FV}. It is important to note that D_0^{FV} also depends on the geometric terms associated with the edges incident to node 0. Similarly, the meshless discretization of Equation 8 based on the Taylor series least square method is

$$Q_{0,t} + \sum_i \left[\boldsymbol{F}(Q_i, a_{0i}, b_{0i}) - \boldsymbol{F}(Q_0, a_{0i}, b_{0i}) \right] - D_0^{M} = 0, \tag{11}$$

where the geometric terms, (a_{0i}, b_{0i}), represent the derivative weights obtained from the least squares minimization procedure in Equation 5, and D_0^M again represents diffusive terms, which are also functions of the coefficients $.(a_{0i}, b_{0i})$ associated with edges incident to node 0.

It is important to highlight a few key differences between these two discretizations. First, the implementation of the finite volume formulation lends itself naturally to an edge-based flux distribution scheme in which a common flux is computed at an edge and subsequently distributed to the nodes which share the edge. Thus, the scheme is reciprocal in nature since a node on an edge lies in the stencil of the opposing node on the same edge. Such reciprocal relationships are not neccessarily present in the meshless scheme, as shown in Figure 2(a), which contains a directed edge connecting node 2 to node 1, but not node 1 to node 2. Directed edges like this are a result of connectivity that is defined by local clouds on a node-wise basis for meshless schemes. This potential break in reciprocity leads to difficulties in the implementation of the meshless scheme since an edge-based method similar to the finite volume implementation could not be used here without significant complications.

A second observation is that the geometric terms for the finite volume scheme, (dx_{0i}, dy_{0i}), are common to the two nodes sharing an edge. Thus, these terms represent the only geometric information that needs to be stored for that edge. In the meshless scheme, however, the geometric information for the two nodes sharing an edge is different due to the local nature of the least squares procedure at each node (assuming the nodes act reciprocally on the edge). For the edge connecting nodes 0 and i, there will arise geometric terms (a_{0i}, b_{0i}) needed for the discretization at node 0. Similarly, there will arise terms (a_{i0}, b_{i0}) at each edge corresponding to the discretization at each node i in the stencil. Consequently, four pieces of geometric information need to be stored for each reciprocal edge. This is double the finite volume edge storage.

A final observation comes as a consequence of the previous observation. Since the geometric information is different for each node sharing an edge in the meshless scheme, two flux computations are necessary for

each edge, instead of a single flux computation as in the finite volume method. This is a significant drawback for meshless schemes, since flux computation represents a significant portion of solution time of the overall algorithm. In the next section, we show how a simple modification to the Taylor series meshless approach circumvents these difficulties.

III. Least Squares Gradient Procedure with Constrained Weights

In order to overcome many of the shortcomings of the Taylor series least squares approach described in the previous section, we develop a modified meshless approach, which addresses the issue of reciprocal connectivity, while reducing geometric storage and flux computations. Our approach requires similar storage and computational effort as conventional finite volume schemes, but still remains non-conservative at a discrete level. Our approach involves a new weighting scheme for the least squares method described in Equation 2. The weighting scheme requires slightly more computational effort in preprocessing, but results in significantly reduced computation time to resolve the solution over the traditional least squares approach.

Before describing the weighting scheme, it is important to note that we adopt an edge-based data structure. Even though the meshless connectivity is the collection of all local clouds, we may express these local clouds implicitly by a collection of edges if we enforce reciprocal relationships between nodes on an edge. If in the local cloud definition procedure, a directed edge arises, as shown in Figure 2(a), the local cloud connectivity is modified to form a reciprocal edge. Since meshes are used in this work to obtain local clouds, reciprocity is automatically enforced. However, in our previous works, reciprocity was not automatically guaranteed, requiring the modifications just described to form an edge-based connectivity.[16,17] As in the case of finite volume schemes, the edge-based data structure greatly simplifies the solution procedure.

Considering the least squares optimization procedure of Equation 2, there is a certain degree of flexibility in defining the weighting system for each node in the local cloud. While we generally follow the approach in Equation 3 to emphasize nodes which are closer in proximity to node 0, we modify these weights to satisfy certain constraints. These constraints are designed to diagonalize the least squares matrix, M, of Equation 4, which simplifies the computation of the derivative weight coefficients, a_{0i} and b_{0i}. If we select weights, w_{0i}, such that

$$\sum_i w_{0i}\Delta x_{0i}^2 = \sum_i w_{0i}\Delta y_{0i}^2, \quad \sum_i w_{0i}\Delta x_{0i}\Delta y_{0i} = 0, \tag{12}$$

then M becomes diagonal with the diagonal entries being equal in magnitude. Note that for regular point distributions that possess a great degree of symmetry, these constraints are already satisfied by the simple weight definition in Equation 3. Thus for reasonably even point distributions, the satisfaction of these constraints should not require significant modification from inverse square weighting. As point distributions become more irregular, the satifaction of these constraints means finding weights which increasingly depart from the inverse square weighting method.

Praveen[21] has examined extensively the properties of the least squares matrix, M, and has shown that its eigenvalues represent the semi-major axes of an ellipse which completely bounds the nodes in a given local cloud. Upon satisfaction of the constraints in Equation 12, the eigenvalues, which are simply the diagonal elements of M, are equal. This means the connectivity is bounded by a circle in the space transformed by the weighting of radius

$$r_0 = \sqrt{\sum_i w_{0i}\Delta x_{0i}^2} = \sqrt{\sum_i w_{0i}\Delta y_{0i}^2}. \tag{13}$$

Furthermore, upon examination of Equation 6 we see that if the constraints of Equation 12 are satisfied, the derivative weight coefficients become simply

$$a_{0i} = \hat{w}_{0i}\Delta x_{0i}, \quad b_{0i} = \hat{w}_{0i}\Delta y_{0i}, \tag{14}$$

where

$$\hat{w}_{0i} = \frac{w_{0i}}{r_0^2}. \tag{15}$$

Note that the vector formed by the coefficients (a_{0i}, b_{0i}) is simply a scalar multiple of the vector joining the endpoints of the edge, $(\Delta x_{0i}, \Delta y_{0i})$. Thus, both a_{0i} and b_{0i} may be represented with a single weight, $\hat{w}_{0i}$, as long as the nodal coordinates are accessible, which is usually the case. Similarly, both a_{i0} and b_{i0} may also be represented with the edge weight, $\hat{w}_{i0} = w_{i0}/r_i^2$, needed for the discretization of the opposing

node i on the edge. Thus, the only storage necessary for the edge connecting nodes 0 and i are the weights $\hat{w}_{0i}$ and $\hat{w}_{i0}$, which is the same amount of storage used in the finite volume scheme of Equation 10. Actually, the meshless scheme requires less storage since no storage for cell volumes is needed. In three dimensions this scheme would still only require two pieces of information per edge, while the finite volume scheme would require three, $(dx_{0i}, dy_{0i}, dz_{0i})$, plus the volume.

Along with decreased storage, the simplified derivative weights of Equation 14 enable only a single flux evaluation per edge, similar to the finite volume method. Applying the derivative weights of Equation 14 to the meshless discretization of the conservation law in Equation 11, we obtain

$$Q_{0,t} + \sum_i \hat{w}_{0i} \left[\boldsymbol{F}(Q_i, \Delta x_{0i}, \Delta y_{0i}) - \boldsymbol{F}(Q_0, \Delta x_{0i}, \Delta y_{0i}) \right] - D_0^M = 0. \tag{16}$$

Note that the convective flux is only a function of the solution at the edge endpoints and the edge vector itself. Once the flux is computed it will be weighted with $\hat{w}_{0i}$ and distributed to node 0. The same flux for this edge can be used and distributed to node i, with a weight, $\hat{w}_{i0}$. The diffusive terms needed for upwinding and stabilization follow a similar procedure. Because of its similarity in implementation to finite volume schemes, we call this scheme the meshless volume scheme.

We have just shown how the satisfaction of the constraints in Equation 12 locally leads to reduced storage and flux computation over the traditional least squares meshless scheme. We now show how we choose weights, w_{0i}, such that these constraints are satisfied in general. The constraints can be satisfied on arbitrary point distributions as part of a local weight optimization procedure. In the optimization problem, a minimum of $g(\boldsymbol{w})$ is sought, subject to the constraints $h_1(\boldsymbol{w})$ and $h_2(\boldsymbol{w})$, where

$$g(\boldsymbol{w}) = \sum_i (w_{0i} - \tilde{w}_{0i})^2, \quad h_1(\boldsymbol{w}) = \sum_i w_{0i}(\Delta x_i^2 - \Delta y_i^2), \quad h_2(\boldsymbol{w}) = \sum_i w_{0i} \Delta x_i \Delta y_i. \tag{17}$$

Here, $\boldsymbol{w}^T = \begin{bmatrix} w_{01} & w_{02} & \cdots & w_{0n} \end{bmatrix}$ is the vector of weights for which we seek corresponding to each node in the local cloud, and $\tilde{\boldsymbol{w}}^T = \begin{bmatrix} \tilde{w}_{01} & \tilde{w}_{02} & \cdots & \tilde{w}_{0n} \end{bmatrix}$ are the corresponding target weights obtained from any conventional weighting procedure, such as the inverse square procedure:

$$\tilde{w}_{0i} = \frac{1}{|\Delta \boldsymbol{r}_{0i}|^2}. \tag{18}$$

The method of Lagrange multipliers may be invoked here, yielding the following system of equations at each node:

$$\begin{bmatrix} I & U^T \\ U & \boldsymbol{0} \end{bmatrix} \begin{Bmatrix} \boldsymbol{w} \\ \boldsymbol{l} \end{Bmatrix} = \begin{Bmatrix} \tilde{\boldsymbol{w}} \\ \boldsymbol{0} \end{Bmatrix}, \tag{19}$$

with

$$U = \begin{bmatrix} \Delta x_{01}^2 - \Delta y_{01}^2 & \Delta x_{02}^2 - \Delta y_{02}^2 & \cdots & \Delta x_{0n}^2 - \Delta y_{0n}^2 \\ \Delta x_{01} \Delta y_{01} & \Delta x_{02} \Delta y_{02} & \cdots & \Delta x_{0n} \Delta y_{0n} \end{bmatrix},$$

where $\boldsymbol{l} = (l_1, l_2)$ is the vector of Lagrange multipliers. Since this system of equations is solved on a local node by node basis, matrix sizes are kept small and efficiency is maintained. This procedure appears to be a small price to pay for the gains in memory and solution efficiency already described. In practice, it has been observed that the optimized weight vector, $\boldsymbol{w}$, does not differ much from the conventional weights, $\tilde{\boldsymbol{w}}$. In other words, for reasonably even point distributions, the derivative weight coefficients are nearly parallel with their edges anyway, so invoking the optimization procedure amounts to merely tweaking the weights in order to satisfy the above constraints.

In order to obtain a conservative scheme, the weights, w_{0i}, would need further constraints, which seem difficult or impossible to satisfy in general. First, the two weights needed for the edge connecting nodes 0 and i, w_{0i} and w_{i0}, would have to be equal in magnitude. If such weights existed, the procedure to find them would involve the solution of a large sparse matrix with dimension equal to the number of edges in the domain, which might cost as much as the flow solution itself. Second, the weights would have to satisfy $\sum_i \hat{w}_{0i} \Delta x_{0i} = \sum_i \hat{w}_{0i} \Delta y_{0i} = 0$ at each node, which would mean the weights actually represented a closed volume, implying the existence of a mesh. However, these constraints appears to be too restrictive in general, over-constraining the weight computation. Both these constraints are automatically satisfied for point clouds

possessing a great deal of symmetry and regularity. Thus, discrete conservation is obtained for these cases, but would likely degrade for increasing cloud irregularity.

In summary, we have shown how the traditional Taylor series least squares method of gradient estimation may be modified with a particular weighting scheme to significantly reduce storage and the number of flux computations. The price for the increased efficiency is a local weight optimization problem with small matrix sizes. Underlying this procedure, we have adopted an edge-based data structure to represent local meshless clouds of points in a reciprocal manner. This gradient estimation technique may be used in a variety of contexts. Next we show it can be applied directly to the Euler equations to compute inviscid flows using a meshless procedure.

IV. Application of the Meshless Volume Method to the Euler Equations

In this section, we directly apply the meshless volume method to the Euler equations in two dimensions. The implementation is facilitated by the use an edge-based data structure, similar to finite volume schemes. We show how the simple geometric information computed from Equation 14 at each edge is sufficient for the entire discretization procedure. The Euler equations in strong conservation law form are

$$\frac{\partial Q}{\partial t} + \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} = 0, \tag{20}$$

where the vector of conserved variables and Euler fluxes are

$$Q = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{pmatrix}, \quad f = \begin{pmatrix} \rho u \\ \rho u^2 + P \\ \rho uv \\ \rho uH \end{pmatrix}, \quad g = \begin{pmatrix} \rho v \\ \rho vu \\ \rho v^2 + P \\ \rho vH \end{pmatrix}.$$

In the above notation, ρ, u, v, P, E, and $H = E + \frac{P}{\rho}$ are the density, Cartesian velocity components, pressure, total energy, and total enthalpy. The Euler equations are completed by the ideal gas law,

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

To obtain a stable meshless discretization, we seek to satisfy a local extremum diminishing (LED) property, as articulated by Jameson.[22] We do this by augmenting the purely convective discretization with artificial diffusion terms which enforce positivity, at least at local extrema. In this manner, a local minimum cannot decrease and a local maximum cannot increase unbounded with time, which is the essential principle of LED schemes. The LED principle leads to the class of total variation diminishing (TVD) schemes in one dimension proposed by Harten.[23] Applying the derivative weights obtained in Equation 14, we can construct a stable scheme of the form

$$Q_{0,t} + \sum_i [\hat{w}_{0i}(F_{li} - F_{l0}) - |\hat{w}_{0i}|d_{0i}] = 0, \tag{21}$$

where $F_l = \Delta x_{0i} f + \Delta y_{0i} g$ is a directed flux in the direction of the edge connecting nodes 0 and i, and d_{0i} is an artificial diffusion term. The design of d_{0i} is critical to the accuracy and convergence characteristics of the meshless scheme. A highly accurate, yet efficient formulation is the convective upwind split pressure (CUSP) scheme of Jameson.[22] The CUSP scheme produces shocks with a single interior point in one dimension, while minimizing computational cost. The CUSP scheme may be formulated to admit solutions with constant stagnation enthalpy, a defining characteristic of steady inviscid flows with uniform freestream.[24] Following the CUSP scheme for constant stagnation enthalpy, we may express the diffusive flux as

$$d_{0i} = \alpha^* c(Q_{hi} - Q_{h0}) + \beta(F_{li} - F_{l0}), \tag{22}$$

with speed of sound, c, and enthalpy variables, $Q_h = (\rho, \rho u, \rho v, \rho H)^T$. Here, α^* and β are dimensionless parameters controlling the amount of diffusion to be added and may be computed following the analysis of Jameson.[25]

We may reduce the amount of artificial diffusion to be added by replacing d_{0i} with d_{LR},

$$d_{LR} = \alpha^* c(Q_{hR} - Q_{hL}) + \beta(F_{lR} - F_{lL}), \tag{23}$$

which is a function of left and right reconstructed states. A simple method to obtain the reconstructed states is the symmetric limited positive (SLIP) scheme of Jameson,[22] which leads to high order accuracy in smooth regions and reverts to the strictly LED scheme in the vicinity of discontinuities to avoid oscillations. Here we adapt the SLIP scheme to our meshless method by defining the left and right states as

$$Q_L = Q_0 + \frac{1}{4}s(\Delta Q^+, \Delta Q^-)(\Delta Q^+ + \Delta Q^-), \quad Q_R = Q_i - \frac{1}{4}s(\Delta Q^+, \Delta Q^-)(\Delta Q^+ + \Delta Q^-). \tag{24}$$

Here, ΔQ^+ and ΔQ^- are estimates of the change in the solution on either side of the edge connecting nodes

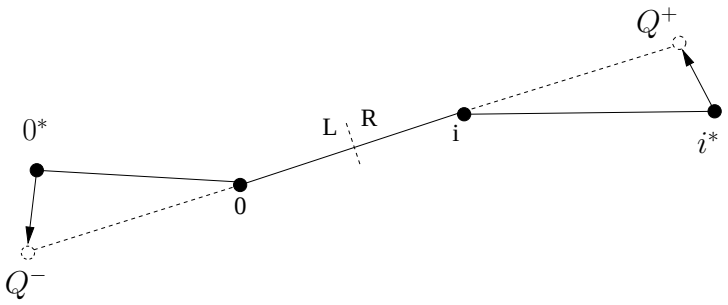


Figure 3. Edge reconstruction procedure to obtain left and right states.

0 and i, as shown in Figure 3. First, points 0^* and i^*, the points in the local clouds of 0 and i which are most closely aligned with the edge $0i$ are identified. Estimates with $\Delta Q^+ = Q_{i^*} - Q_i$ and $\Delta Q^- = Q_0 - Q_{0^*}$ may be used. However an increase in accuracy is observed by making corrections of the form

$$\Delta Q^+ = Q_{i^*} - Q_i + l_i \cdot \nabla Q_{i^*}, \quad \Delta Q^- = Q_0 - Q_{0^*} - l_0 \cdot \nabla Q_{0^*}, \tag{25}$$

where the gradients of Q are computed using the same least squares coefficients as the base meshless volume scheme, and l_0 and l_i are the position vectors from 0^* and i^* to the psuedo-points past the edge. The ΔQ estimates obtained in this way have lead to sharp capturing of discontinuities.

For the enforcement of boundary conditions, we prefer to use ghost points while using the same interior scheme for points coinciding with boundaries. At domain boundaries, interior points are reflected across tangent boundary planes to form ghost points, as shown in Figure 4. The ghost points serve two purposes. First, they balance the local clouds of the boundary points, improving the condition of the least squares computation and the weight optimization procedure. Second, solution values at ghost points may be set to enforce flow tangency or far field conditions. With the injection of proper conditions at the boundary, no modification of the interior scheme is necessary at the boundary points. Integration to steady state was accomplished with the modified Runge-Kutta approach of Jameson[26] with local time stepping, enthalpy damping, and implicit residual smoothing. Additionally we have implemented the multicloud algorithm of Katz and Jameson,[17] enhancing the convergence to steady state.

The savings in geometric information storage by using the meshless volume Euler scheme is quantified in Table 1. Along with the meshless volume (MV) scheme, shown in the table are the traditional Taylor meshless (Tay) scheme and nodal finite volume schemes on simplicial volumes for both least squares (FV-LS) and Green-Gauss (FV-GG) linear reconstruction. The geometric storage needed for the nodal coordinates, volumes, flux distribution, and reconstruction procedures are shown. Flux distribution includes least squares gradient weights for the meshless schemes or face areas for the finite volume schemes. If a "0" appears in the reconstruction column, it means that the same coefficients that are used for flux distribution are also used for reconstruction, and no additional storage is counted. This is the case for three of the schemes, while the finite volume scheme with least squares reconstruction requires additional storage. The total number of geometric variables to be stored, shown in the final column, is the sum of the previous column values, where the number of edges, n_e, is expressed in terms of the number of points, n_p, in a typical mesh. The number of edges exceeds the number of points by a factor of roughly three in two dimensions and roughly seven in

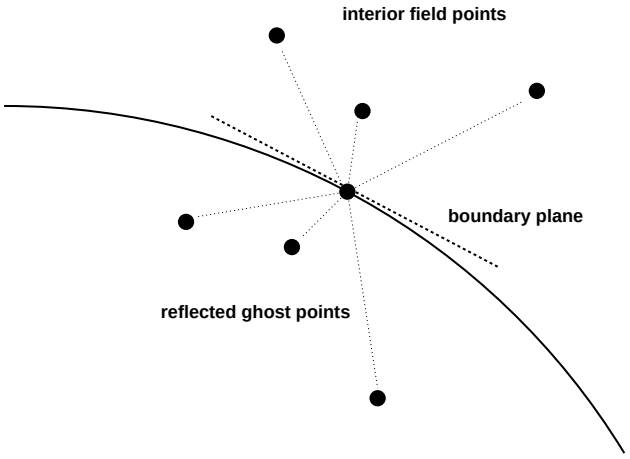


Figure 4. Reflection of interior points to produce ghost nodes.

three dimensions.[27] As the table indicates, the geometric storage needed for the meshless volume scheme is the smallest, being slightly smaller than finite volume with Green-Gauss reconstruction. It is nearly a factor of two more efficient than the traditional meshless approach, and nearly a factor of three smaller than finite volume with least squares reconstruction in both two and three dimensions. Aside from geometric storage, the remaining storage needed for the solution procedure is expected to be about the same for all four schemes.

Table 1. Geometric storage comparison of various schemes. ($n_e \approx 3n_p$ in two dimensions, $n_e \approx 7n_p$ in three dimensions.)

		coordinates	volume	flux distribution	reconstruction	total
2D	MV	$2n_p$	0	$2n_e$	0	$8n_p$
	Tay	$2n_p$	0	$4n_e$	0	$14n_p$
	FV-LS	$2n_p$	n_p	$2n_e$	$4ne$	$21n_p$
	FV-GG	$2n_p$	n_p	$2n_e$	0	$9n_p$
3D	MV	$3n_p$	0	$3n_e$	0	$24n_p$
	Tay	$3n_p$	0	$6n_e$	0	$45n_p$
	FV-LS	$3n_p$	n_p	$3n_e$	$6ne$	$67n_p$
	FV-GG	$3n_p$	n_p	$3n_e$	0	$25n_p$

While more difficult to quantify, the CPU time spent per iteration for the meshless volume scheme should be approximately equal to the time spent for the finite volume schemes since the number of flux computations and other solution procedures are similar. The traditional Taylor approach is expected to be more expensive, since two flux computations per edge are needed.

V. Results in Two Dimensions

In order to characterize the error of the meshless volume scheme, we performed a fundamental grid convergence study. The study consisted of prescribing a known function with known gradient over a domain, and comparing the computed gradient with the exact gradient. We applied the function,

$$f(x, y) = \sin x \sin y,$$

the exact gradient of which is easily found analytically. We then computed the gradient of this function using the meshless volume technique on a variety of point distributions, shown in Figure 5. The nodes and

connectivity were obtained from structured and triangular meshes which were both regular and perturbed in nature. We computed the L2 norm of the error between the exact and computed gradients. As shown in Figure 5(e), the regular distibutions led to gradient estimates which were second order accurate, while the perturbed meshes led to estimates which were first order accurate. This is the expected result, common to linear preserving schemes in general, which often lose one order of accuracy as meshes become distorted. We would expect similar behavior from most finite volume, finite element, or finite difference schemes with linear representations. Even with the new least squares weighting scheme we have proposed, the meshless volume method of gradient estimation formally does not lose accuracy compared to the traditional method. Thus, the scheme appears suitable for direct application to the Euler equations for a meshless discretization or for other gradient estimation applications, such as linear reconstruction.

With regard to our meshless discretization of the Euler equations, we first present three airfoil cases, which should give zero drag, since the flow is inviscid and we prescribe conditions which lead to shock free results. We tested three airfoils at shock-free operating conditions, as shown in Table 2. Two of the cases, the NACA and RAE airfoils, were tested at subsonic conditions, while the KORN airfoil was tested in a transonic regime. A series of three increasingly refined meshes were used to assess the convergence to zero drag. In all three cases, nearly zero drag was obtained with 160 points on the airfoil surface. It appears that error due to the artificial diffusion remains low as the mesh in refined as indicated by the rapid convergence to zero drag for all cases. Additionally, the highly sensitive transonic KORN airfoil case produced zero drag, which is the theoretical behavior of this airfoil at the shown operating condition.

Table 2. Drag convergence for inviscid test cases

	NACA 0012 $M = 0.5$, $\alpha = 3^o$		KORN Airfoil $M = 0.75$, $\alpha = 0^o$		RAE 2822 $M = 0.5$, $\alpha = 3^o$	
Surface points	Total points	C_d	Total points	C_d	Total points	C_d
40	526	0.0082	474	0.0036	457	0.0104
80	1591	0.0009	1919	0.0013	1580	0.0018
160	5443	0.0001	7810	0.0000	5733	0.0003

Figures 6-8 show the surface pressure, pressure contours, mach contours, and the meshless point distribution for three well known inviscid test cases. We show both subsonic and transonic results for the NACA 0012 airfoil, along the with transonic KORN airfoil case discussed above. The local cloud definitions for these cases were extracted from a triangulation created with the Delaundo package.[28] The airfoil surfaces contained 160 points, while the interior domains contained roughly 5000 points. Also provided with each test case is a comparison of lift and drag with a structured finite volume solver, which also makes use of CUSP and SLIP algorithms. The comparison to finite volume results is also made in the plots of surface pressure, with the dots showing the meshless result, and the solid line showing the finite volume result. The subsonic case of Figure 6 shows nearly zero drag. In the transonic case of Figure 7, the shock location and magnitude coincide well with the finite volume results. In the case of the KORN airfoil of Figure 8, the upper surface is virtually shock free at its design point, indicating a high level of accuracy. The KORN airfoil case is particularly encouraging, since the shock-free result is highly sensitive and difficult to obtain for many schemes. The results for all cases tested agree remarkably well with established finite volume results in spite of a loss of formal conservation at the discrete level.

VI. Conclusions

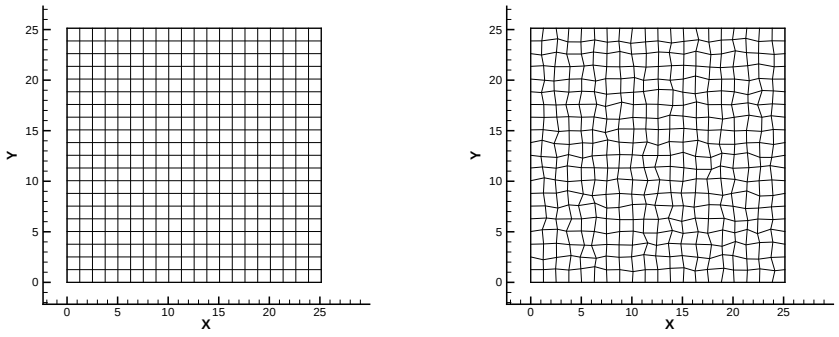
We have developed a meshless technique, which significantly reduces storage and flux computations as compared with traditional least squares meshless approaches. We have done this by adopting an edge-based connectivity to describe local clouds, and by introducing a weighting scheme which simplifies the least squares procedure. The cost of computing the new weights is relatively low, considering the reduced memory and fewer flux computations needed to resolve the solution. We have shown similarities between the meshless scheme and finite volume schemes in two dimensions. Both schemes are similar in implementation. However, since geometric terms of the meshless scheme do not form a closed volume, discrete conservation appears difficult to prove.

We have applied the new scheme directly to the Euler equations in two dimensions. We have used an LED-based method with artificial diffusion derived from CUSP and SLIP concepts to accurately resolve flow features. We use an explicit method with a truly meshless multigrid-like operator to rapidly achieve steady state. The results agree well with established methods in terms of solution time and accuracy for all cases tested despite the non-conservation of the scheme.

In this work we have shown improvement over many existing meshless schemes in terms of efficiency in memory and computation. In doing this, we have shown that accuracy does not appear to suffer for all cases tested. The widespread use of this or other meshless schemes will depend on the creation of point and cloud generation techniques which can be shown to relieve mesh generation difficulties. While our previous works have demostrated this in certain specific scenarios,[16,17] widespread use of meshless schemes depends on more general approaches to point generation and cloud definition. Future work in meshless schemes will focus on these areas. Providing that point and cloud generation prove successful, the application of the meshless volume scheme could prove to be a highly efficient and easily implemented scheme useful for many CFD problems.

VII. Acknowledgements

Development was funded by a National Defense Science and Engineering Graduate (NDSEG) fellowship. The NDSEG fellowship was administered through the High Performance Computing Modernization Office of the Department of Defense. The second author has benefited greatly from the long term and continuing support of the AFOSR Computational Mathematics Program directed by Dr. Fariba Fahroo.

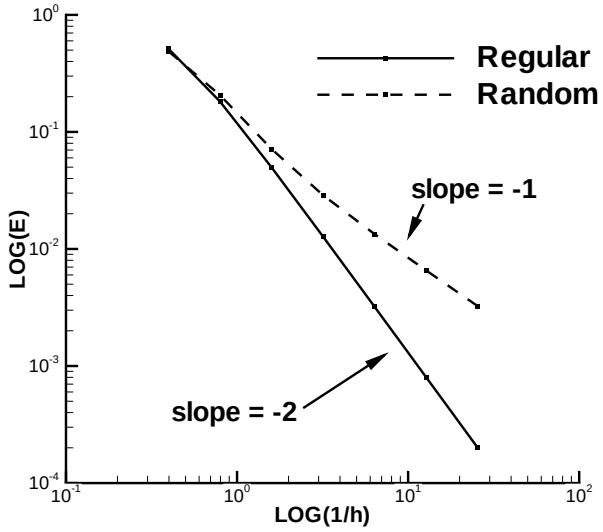


(a) Regular structured

(b) Random structured

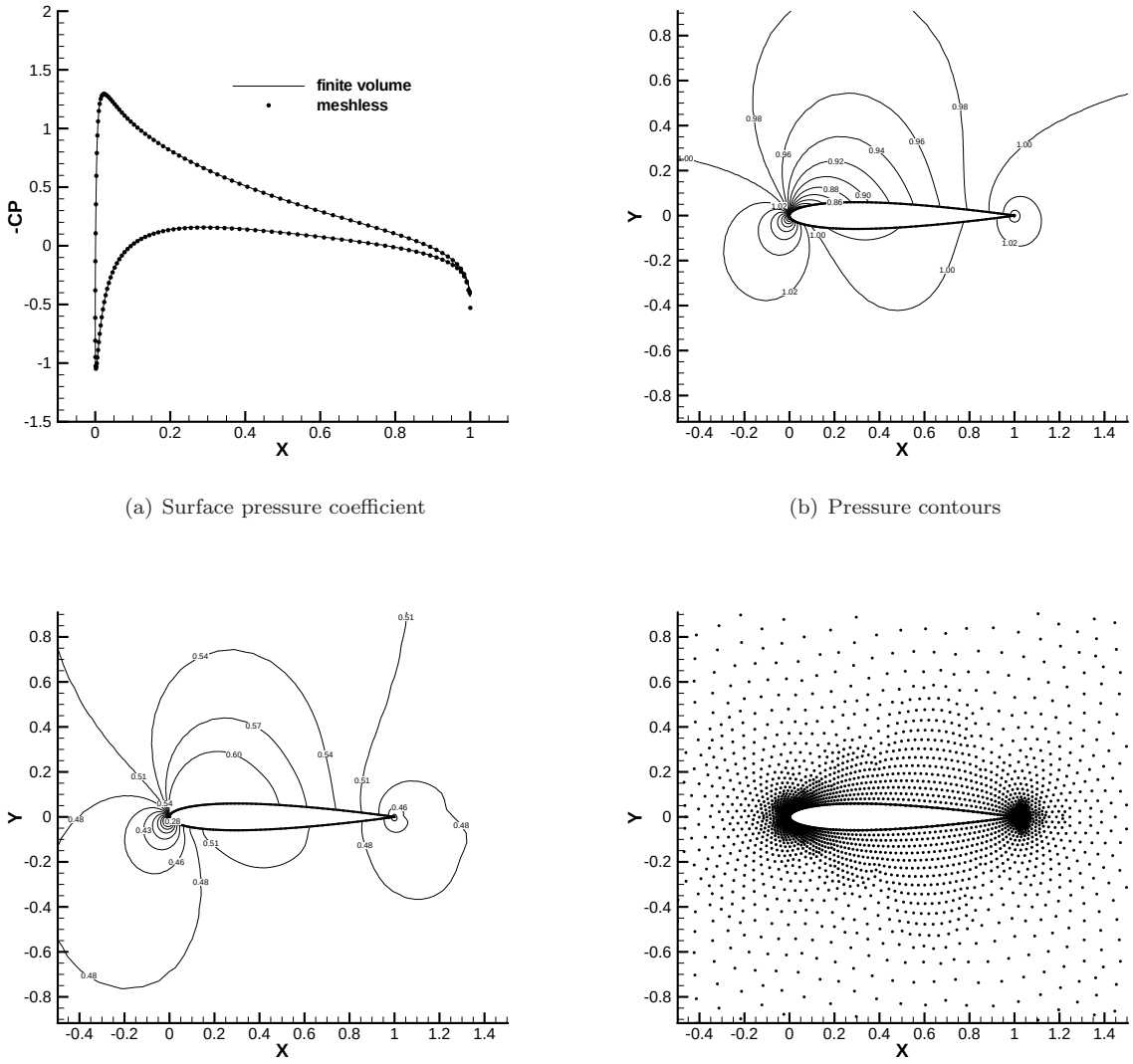
(c) Regular triangular

(d) Random triangular



(e) Grid Convergence

Figure 5. Grid Convergence study.



(a) Surface pressure coefficient

(b) Pressure contours

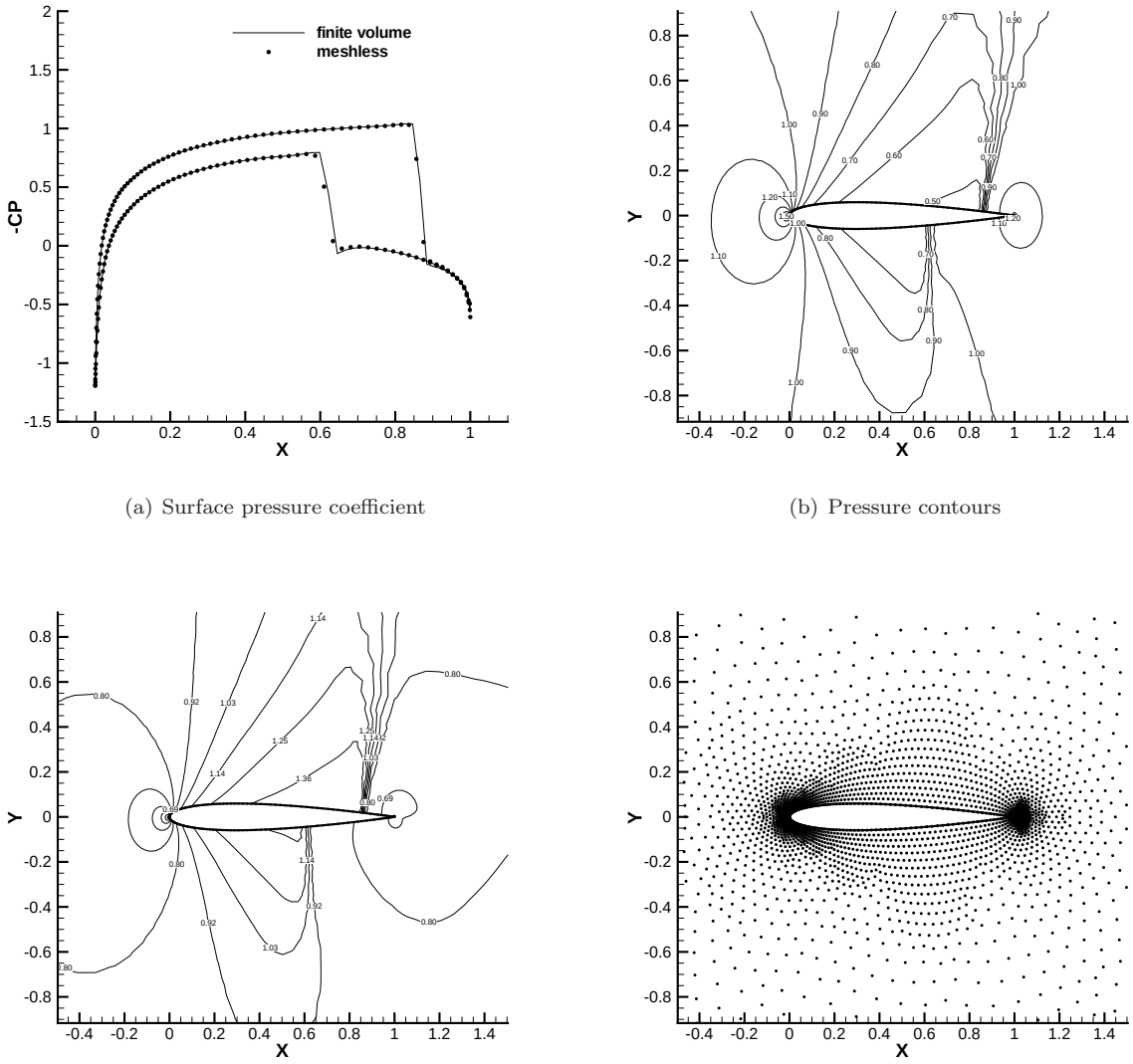
(c) Mach contours

(d) Point distribution

Figure 6. Flow over NACA 0012, $M = 0.50$, $\alpha = 3.0^o$, **Meshless volume scheme.**

Table 3. Lift and drag coefficients, NACA 0012 airfoil, $M = 0.5$, $\alpha = 3.0^o$.

	c_l	c_d
FV	0.4313	0.0000
MV	0.4321	0.0001



(a) Surface pressure coefficient

(b) Pressure contours

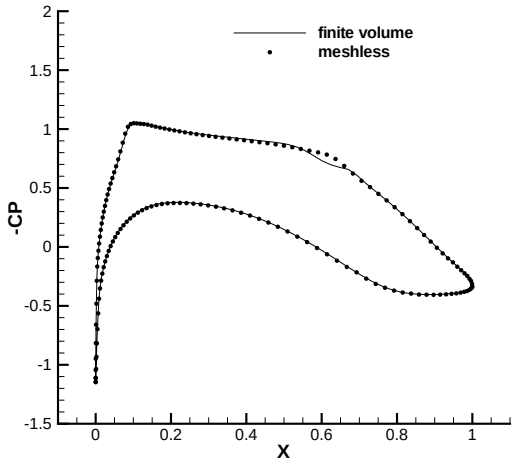
(c) Mach contours

(d) Point distribution

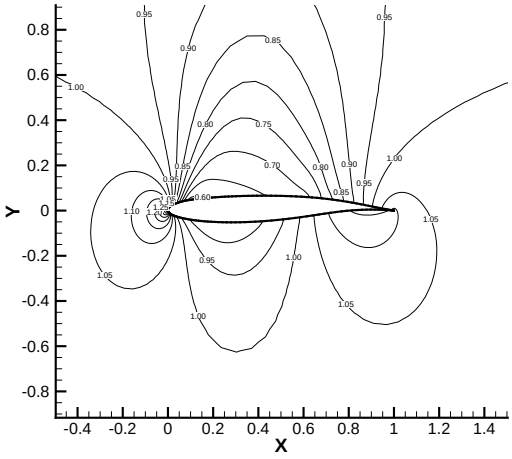
Figure 7. Flow over NACA 0012, $M = 0.85$**,** $\alpha = 1.0^o$**, Meshless volume scheme.**

Table 4. Lift and drag coefficients, NACA 0012 airfoil, $M = 0.85$**,** $\alpha = 1.0^o$**.**

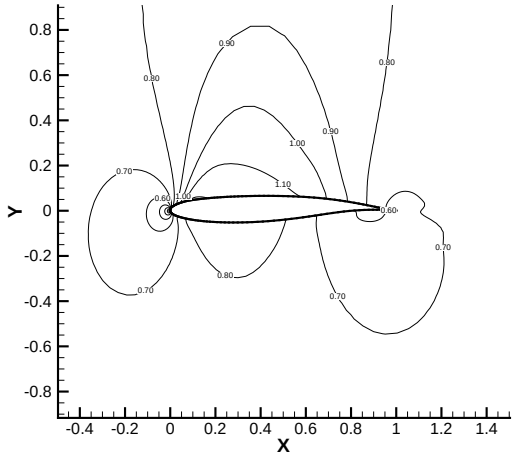
	c_l	c_d
FV	0.3891	0.0582
MV	0.3923	0.0572



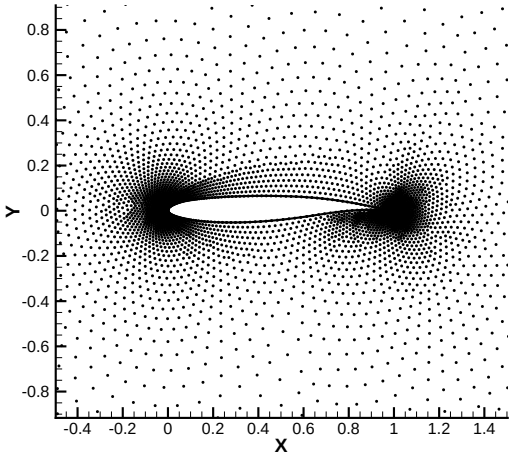
(a) Surface pressure coefficient



(b) Pressure contours



(c) Mach contours



(d) Point distribution

Figure 8. Flow over KORN airfoil, $M = 0.75$, $\alpha = 0.0^o$, Meshless volume scheme.

Table 5. Lift and drag coefficients, KORN airfoil, $M = 0.75$, $\alpha = 0.0^o$.

	c_l	c_d
FV	0.6308	0.0000
MV	0.6353	0.0000

References

[1]Löhner, R. and Oñate, E., "An Advancing Front Point Generation Technique," *Communications in Numerical Methods in Engineering*, Vol. 14, 1998, pp. 1097–1108.

[2]Löhner, R., Sacco, C., and Oñate, E., "A General Advancing Front Technique for Filling Space with Arbitrary Objects," *Int. J. Numer. Meth. Engng.*, Vol. 61, 2004, pp. 1977–1991.

[3]Batina, J. T., "A Gridless Euler/Navier-Stokes Solution Algorithm for Complex Aircraft Applications," *AIAA paper* 1993-0333, AIAA 31st Aerospace Sciences Meeting and Exhibit, Reno, NV, January 1993.

[4]Oñate, E., Idelsohn, S., Zienkiewicz, O. C., Taylor, R. L., and Sacco, C., "A Stabilized Finite Point Method for Analysis of Fluid Mechanics Problems," *Comput. Methods Appl. Mech. Engrg.*, Vol. 139, 1996, pp. 315–346.

[5]Oñate, E., Idelsohn, S., Zienkiewicz, O. C., 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.

[6]Oñate, E. and Idelsohn, S., "A Mesh-free Finite Point Method for Advective-diffusive Transport and Fluid Flow Problems," *Computational Mechanics*, Vol. 21, 1998, pp. 283–292.

[7]Löhner, R., Sacco, C., Oñate, E., and Idelsohn, S., "A Finite Point Method for Compressible Flow," *Int. J. Numer. Meth. Engng.*, Vol. 53, 2002, pp. 1765–1779.

[8]Morinishi, K., "Effective Accuracy and Conservation Consistency of Gridless Type Solver," *Computational Fluid Dynamics 2000: Proceedings of the First International Conference on Computational Fluid Dynamics*, edited by N. Satofuka, Springer-Verlag, 2000, pp. 325–330.

[9]Sridar, D. and Balakrishnan, N., "An Upwind Finite Difference Scheme for Meshless Solvers," *Journal of Computational Physics*, Vol. 189, 2003, pp. 1–29.

[10]Praveen, C., , and Balakrishnan, N., "New least squares based finite difference scheme for compressible flows," Tech. rep., Proceedings of the 8th Asian Congress of Fluid Mechanics, Shenzen, China, December 1999.

[11]Balakrishnan, N. and Praveen, C., "A new upwind least squares finite difference scheme (LSFD-U) for Euler equations of gas dynamics," *Finite Volumes for Complex Applications, vol. II*, edited by F. Benkhaldoun, R. Vilsmeier, and D. Hänel, Hermes Science Publications, 1999, p. 331.

[12]Liu, W. K., Jun, S., and Zhang, Y. F., "Reproducing Kernel Particle Methods," *International Journal for Numerical Methods in Fluids*, Vol. 20, 1995, pp. 1081–1106.

[13]Monaghan, J. J. and Gingold, R. A., "Shock Simulation by the Particle Method SPH," *Journal of Computational Physics*, Vol. 52, 1983, pp. 374–389.

[14]Belytschko, T., Lu, Y. Y., and Gu, L., "Element-Free Galerkin Methods," *International Journal for Numerical Methods in Engineering*, Vol. 37, 1994, pp. 229–256.

[15]Duarte, C. and Oden, J., "Hp clouds - a meshless method to solve boundary-value problems," Technical Report 95-05, Texas Institute for Computational and Applied Mathematics, Austin, TX, 1995.

[16]Katz, A. and Jameson, A., "Edge-based Meshless Methods for Compressible Viscous Flow with Applications to Overset Grids," *AIAA paper* 2008-3989, AIAA 38th Fluid Dynamics Conference, Seattle, WA, June 2008.

[17]Katz, A. and Jameson, A., "Multicloud: Multigrid convergence with a meshless operator," *Journal of Computational Physics*, Vol. 228, 2009, pp. 5237–5250.

[18]Barth, T. J., "A 3d Upwind Euler Solver for Unstructured Meshes," *AIAA paper* 1991-1548, AIAA 10th Computational Fluid Dynamics Conference, Honolulu, HI, June 1991.

[19]Delanaye, M., "Polynomial Reconstruction Finite Volume Schemes for the Compressible Euler and Navier-Stokes Equations on Unstructured and Adaptive Grids," Phd thesis, Universite de liege, 1998.

[20]Jameson, A., Baker, T. J., and Weatherill, N. P., "Calculation of Inviscid Transonic Flow over a Complete Aircraft," *AIAA paper* 1986-0103, AIAA 24th Aerospace Sciences Meeting, Reno, NV, January 1986.

[21]Praveen, C., "Some Results on the Least Squares Formula," *FM Report* 2003-FM-10, IISc, 2001.

[22]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.

[23]Harten, A., "High Resolution Schemes for Hyperbolic Conservation Laws," *Journal of Computational Physics*, Vol. 49, 1983, pp. 357–393.

[24]Liepmann, H. W. and Roshko, A., *Elements of Gasdynamics*, Dover Publications, Inc., 1957.

[25]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.

[26]Jameson, A., Schmidt, W., and Turkel, E., "Numerical Solutions of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time-Stepping Schemes," *AIAA paper* 1981-1259, AIAA 14th Fluid and Plasma Dynamic Conference, Palo Alto, CA, June 1981.

[27]Mavriplis, D. J., "Unstructured Mesh Discretizations and Solvers for Computational Aerodynamics," *AIAA paper* 2007-3955, AIAA 18th Computational Fluid Dynamics Conference, Miami, FL, June 2007.

[28]Müller, J., "On Triangles and Flow," Phd thesis, The University of Michigan, 1996.