Multidimensional Reconstruction Schemes of Finite-Volume, Finite-Element, and/or Residual-Distribution

Hiroaki Nishikawa

September 10, 2008

1 Residual-Distribution

The residual-distribution(RD) method is based on the nodal solutions, and can be applied to any types of grids (structured, unstructured, or even mixed ones). It stores the solution data at nodes as the node-centered finite-volume(FV) method, but evaluates the residual in a different way: RD evaluates the residual at each node via the cell-residuals (defined over cells that share the node) while FV evaluates the residual directly over the dual-volume around each node.

With the solution values stored at nodes, we make a loop over elements, $T \in \{T\}$, and evaluate the cell-residual as

$$\mathbf{\Phi}^T = \int_T (\mathbf{f}_x + \mathbf{g}_y)\, dxdy, \tag{1}$$

where $\mathbf{f}_x + \mathbf{g}_y = 0$ is the equations we wish to solve. If this vanishes (for a chosen quadrature rule), we are happy with the current nodal solutions, but if it doesn't, we request changes to the nodes of the element, $\{i_T\}$, by distributing the fraction of the non-zero cell-residual, $\mathbf{\Phi}_i^T$:

$$\delta \mathbf{U}_i = \delta \mathbf{U}_i + \mathbf{\Phi}_i^T, \quad i \in \{i_T\}, \tag{2}$$

where $\mathbf{\Phi}_i^T$ is determined by the distribution matrix $\mathcal{B}_i^T$,

$$\mathbf{\Phi}_i^T = \mathcal{B}_i^T \mathbf{\Phi}^T. \tag{3}$$

(See Figure 1). The property of the scheme largely depends upon the choice of $\mathcal{B}_i^T$. After completing the loop over elements, we end up with the following semi-discrete equation,

$$S_j \frac{d\mathbf{U}_j}{dt} = -\sum_{T \in \{T_j\}} \mathcal{B}_j^T \mathbf{\Phi}_i^T, \tag{4}$$

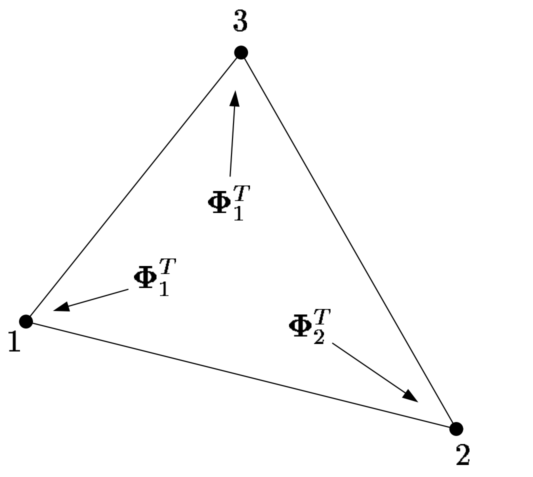

Figure 1: Distribution of a non-zero cell-residual to the set of vertices $\{i_T\} = \{1,2,3\}$.

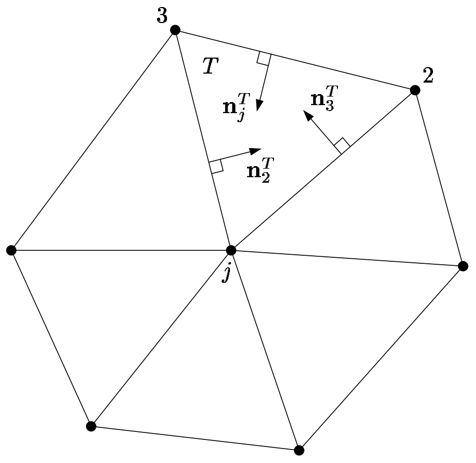

Figure 2: Definition of scaled inward normals, which sum up to zero over the cell T: $\mathbf{n}_j^T + \mathbf{n}_2^T + \mathbf{n}_3^T = \mathbf{0}$.

where S_j is the measure of the dual-volume, which we integrate in time to reach the steady state.

Conservation is ensured by the condition,

$$\sum_{i\in\{i_T\}} \mathbf{\Phi}_i^T = \mathbf{\Phi}^T, \tag{5}$$

i.e., the local updates sum up to the cell-residual within the element. This guarantees that the global sum of the updates depends only on the boundary data. Alternatively, conservation can be ensured by

$$\sum_{i\in\{i_T\}} \mathbf{\Phi}_i^T = \mathbf{0}. \tag{6}$$

The least-squares (residual-minimization) scheme, which is defined by

$$\mathcal{B}_i^T = -\frac{\partial \mathbf{\Phi}^T}{\partial \mathbf{U}_i}, \tag{7}$$

has this property.

2 Finite-Volume as Residual-Distribution

Consider a node-centered finite-volume method which stores the solution values at nodes and evaluates their time rate of change by integrating the governing equations over the dual-control volume. This results in the following semi-discrete equation at each node j,

$$S_j \frac{d\mathbf{U}_j}{dt} = -\sum_{k \in \{k_j\}} \left[\mathcal{F}_n(\mathbf{U}_j, \mathbf{U}_k, \mathbf{n}_{jk}^L) + \mathcal{F}_n(\mathbf{U}_j, \mathbf{U}_k, \mathbf{n}_{jk}^R) \right], \tag{8}$$

where $\{k_j\}$ denotes a set of immediate neighbors of the node j, and $\mathcal{F}_n(\mathbf{U}_j, \mathbf{U}_k, \mathbf{n}_{jk}^{L,R})$ is a numerical flux vector evaluated in the direction of $\mathbf{n}_{jk}^{L,R}$ between the two states $\mathbf{U}_j$ and $\mathbf{U}_k$ (See Figure 3). Note that we have assumed that the method evaluates the numerical flux twice per each edge (for the left and right normals), or that the numerical flux along the edge can be split into two parts in the form above.

This can be rewritten as a sum over the cells that share the node j,

$$S_j \frac{d\mathbf{U}_j}{dt} = -\sum_{T \in \{T_j\}} \left[\mathcal{F}_n(\mathbf{U}_j, \mathbf{U}_l, \mathbf{n}_{jl}^T) + \mathcal{F}_n(\mathbf{U}_j, \mathbf{U}_r, \mathbf{n}_{jr}^T) \right], \tag{9}$$

where the nodes of the cell T have been defined counter-clockwise by $\{j, r, l\}$ (See Figure 4). In this form, the FV scheme can be coded as a two-loop method: a loop over cells to evaluate updates, $\mathbf{\Phi}_j^T$, to the nodes, followed by a loop over nodes to actually update the nodal solution values. This is nothing but residual-distribution:

$$S_j \frac{d\mathbf{U}_j}{dt} = -\sum_{T \in \{T_j\}} \mathbf{\Phi}_j^T, \tag{10}$$

where

$$\mathbf{\Phi}_j^T = \mathcal{F}_n(\mathbf{U}_j, \mathbf{U}_l, \mathbf{n}_{jl}^T) + \mathcal{F}_n(\mathbf{U}_j, \mathbf{U}_r, \mathbf{n}_{jr}^T). \tag{11}$$

Note that we have

$$\sum_{i \in \{i_T\}} \mathbf{\Phi}_i^T = \mathbf{0}, \tag{12}$$

because the fluxes in the opposite direction cancel each other, e.g., $\mathcal{F}_n(\mathbf{U}_1, \mathbf{U}_2, \mathbf{n}_{12}) + \mathcal{F}_n(\mathbf{U}_1, \mathbf{U}_2, \mathbf{n}_{21}) = 0$. In a way, this vanishing sum is a requirement for the finite-volume scheme to be conservative. Then, this scheme is not really residual-distribution in the sense that there exists no

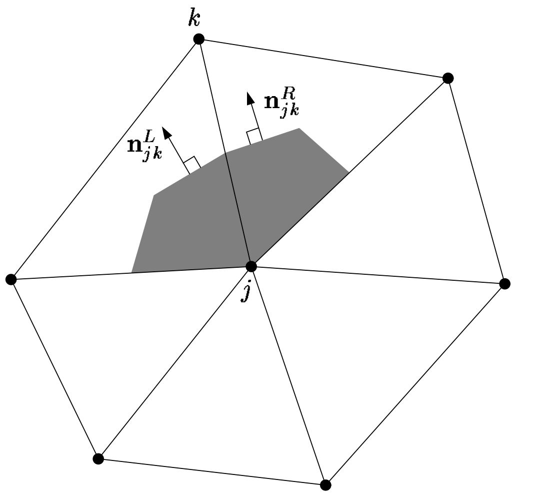

Figure 3: Definition of normals in the edge-wise implementation of finite-volume schemes

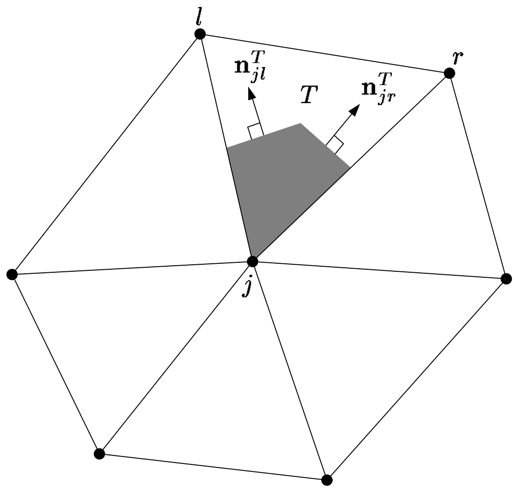

Figure 4: Definition of normals in the cell-wise implementation of finite-volume

cell-residuals which drive the solution updates. If we wish to make it residual-distribution, we may add $\frac{1}{2}(\mathbf{f}_i, \mathbf{g}_i) \cdot \mathbf{n}_i^T$, i.e.,

$$\mathbf{\Phi}_j^T = \frac{1}{2}(\mathbf{f}_i, \mathbf{g}_i) \cdot \mathbf{n}_i^T + \mathcal{F}_n(\mathbf{U}_j, \mathbf{U}_l, \mathbf{n}_{jl}^T) + \mathcal{F}_n(\mathbf{U}_j, \mathbf{U}_r, \mathbf{n}_{jr}^T), \tag{13}$$

so that the updates sum up to a cell-residual:

$$\sum_{i \in \{i_T\}} \mathbf{\Phi}_i^T = \frac{1}{2} \sum_{i \in \{i_T\}} (\mathbf{f}_i, \mathbf{g}_i) \cdot \mathbf{n}_i^T. \tag{14}$$

But note that the added extra term will vanish when the updates are summed over the neighboring cells of node j,

$$\sum_{T \in \{T_j\}} \frac{1}{2}(\mathbf{f}_j, \mathbf{g}_j) \cdot \mathbf{n}_j^T = 0. \tag{15}$$

Therefore, it does not affect the solution update at the node j.

Taking this cell-wise implementation viewpoint of the finite-volume schemes, we can devise a new scheme, for example, by redefining the partial cell-residual in the form,

$$\mathbf{\Phi}_j^T = \left(\mathcal{F}_x^T, \mathcal{F}_y^T\right) \cdot \left(\mathbf{n}_{jl}^T + \mathbf{n}_{jr}^T\right), \tag{16}$$

where $\mathcal{F}_x^T$ and $\mathcal{F}_y^T$ are numerical fluxes evaluated at the centroid of the cell T, based on two states which may be taken along any arbitrary direction and the orthogonal direction. This is the form used by Gnoffo in his multidimensional reconstruction scheme ($\mathcal{F}_x^T \to \mathbf{f}_{x'}$ and $\mathcal{F}_y^T \to \mathbf{f}_{y'}$ in his original notation). Note that this can also be written in the following form:

$$\mathbf{\Phi}_j^T = -\frac{1}{2}\left(\mathcal{F}_x^T, \mathcal{F}_y^T\right) \cdot \mathbf{n}_j^T, \tag{17}$$

where $\mathbf{n}_j^T$ is now the scaled inward normal vector of the side opposite to the node j within the cell T (Figure 2). Conservation is guaranteed in the sense that the updates sum up to zero within the cell:

$$\sum_{i \in \{i_T\}} \mathbf{\Phi}_i^T = -\frac{1}{2}\left(\mathcal{F}_x^T, \mathcal{F}_y^T\right) \cdot \sum_{i \in \{i_T\}} \mathbf{n}_i^T = \mathbf{0}. \tag{18}$$

Gnoffo devised ways to evaluate the cell-wise fluxes $\mathcal{F}_x^T$ and $\mathcal{F}_y^T$ based on a multidimensional consideration: evaluate $\mathcal{F}_x^T$ along the directoin of local density gradient (or velocity); $\mathcal{F}_y^T$ along the orthogonal direction.

3 Finite-Volume as Finite-Element

It is well known that the finite-volume scheme with the straightforward central flux is equivalent to the Galerkin finite-element scheme. In the same way, the cell-wise finite-volume schemes in the previous section can be formulated as a finite-element scheme.

Let ψ_j be the linear basis function (i.e., $\psi_j = 1$ at the node j, and $\psi_j = 0$ at immediate neighbors). Then, weight the governing equations by ψ_j and integrate them over the entire domain:

$$\int_\Omega \psi_j(\mathbf{f}_x + \mathbf{g}_y)\, dxdy = 0. \tag{19}$$

Because ψ_j has a compact support, this reduces to the integral over the cells that share the node j, i.e., $\{T_j\}$,

$$\int_{\{T_j\}} \psi_j(\mathbf{f}_x + \mathbf{g}_y)\, dxdy = 0. \tag{20}$$

We then integrate this by parts.

$$\oint_{\partial\{T_j\}} \psi_j(\mathbf{f}dy - \mathbf{g}dx) - \int_{\{T_j\}} \nabla\psi_j \cdot (\mathbf{f}, \mathbf{g})\, dxdy = 0. \tag{21}$$

The first term vanishes because ψ_j vanishes on the boundary of $\{T_j\}$. Also, note that $\nabla\psi_j$ is constant over the cell T:

$$\nabla\psi_j = \frac{\mathbf{n}_j^T}{2S_T}. \tag{22}$$

Hence, we can write it as

$$-\sum_{T\in\{T_j\}} \frac{\mathbf{n}_j^T}{2S_T}\cdot\left[\int_T (\mathbf{f},\mathbf{g})\,dxdy\right] = 0. \tag{23}$$

Therefore, this can be thought of as residual-distribuion with

$$\boldsymbol{\Phi}_j^T = -\frac{1}{2S_T}\left[\int_T (\mathbf{f},\mathbf{g})\,dxdy\right]\cdot\mathbf{n}_j^T. \tag{24}$$

Various schemes can be devised, depending on how the volume integral of the flux vectors is evaluated. For example, if we assume the linear variation of the flux vectors over the cell, the straightforward integration yields

$$\boldsymbol{\Phi}_j^T = -\frac{1}{2}\left(\frac{1}{3}\sum_{i\in\{i_T\}}(\mathbf{f}_i,\mathbf{g}_i)\right)\cdot\mathbf{n}_j^T. \tag{25}$$

It is easy to show that this is equivalent to the standard finite-volume scheme with central flux and the standard Galerkin scheme. Alternatively, we may use a quadrature with three midpoints

$$\boldsymbol{\Phi}_j^T = -\frac{1}{2}\left(\frac{1}{3}\sum_{i\in\{i_m\}}(\mathbf{f}_i,\mathbf{g}_i)\right)\cdot\mathbf{n}_j^T. \tag{26}$$

or even with just one point at the centroid,

$$\boldsymbol{\Phi}_j^T = -\frac{1}{2}(\mathbf{f}_c,\mathbf{g}_c)^T\cdot\mathbf{n}_j^T, \tag{27}$$

which is still second-order accurate. In the latter case, replacing the physical fluxes by numerical ones, we obtain the finite-volume scheme of the form (17), i.e.,

$$\boldsymbol{\Phi}_j^T = -\frac{1}{2}\left(\mathcal{F}_x^T(\mathbf{n}^x),\mathcal{F}_y^T(\mathbf{n}^y)\right)\cdot\mathbf{n}_j^T. \tag{28}$$

Whichever the quadrature is chosen for the discretization, the property of the resulting scheme is determined by the way we evaluate the flux vectors at each quadrature point: central, upwind, or multidimensional.

4 Inflow Parameters

For a given direction indicated by the vector $\boldsymbol{\lambda}$, the inflow parameter, denoted by k_i, is defined for each node i within the cell T by

$$k_i = \frac{1}{2}\boldsymbol{\lambda}\cdot\mathbf{n}_i^T. \tag{29}$$

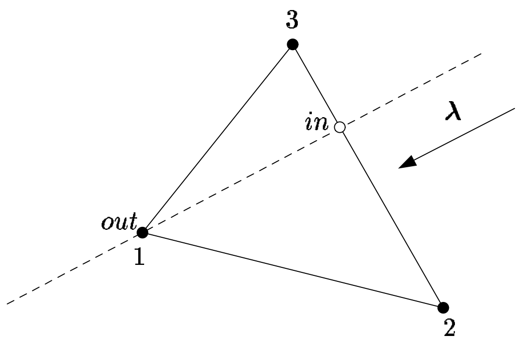

Figure 5: One target case: $k_1 > 0$, $k_2 < 0$, $k_3 < 0$.

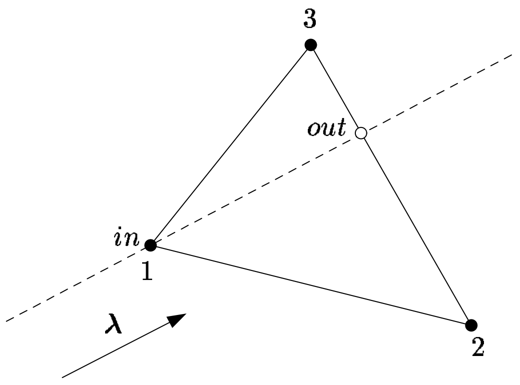

Figure 6: Two target case: $k_1 < 0$, $k_2 > 0$, $k_3 > 0$.

Obviously, the inflow parameters sum up to zero over the cell:

$$\sum_{i \in \{i_T\}} k_i = \frac{1}{2} \boldsymbol{\lambda} \cdot \sum_{i \in \{i_T\}} \mathbf{n}_i^T = \mathbf{0}. \tag{30}$$

This means that there are only two different cases (in 2D): either one or two parameters are positive. The former is called the 1-target case (Figure 5) while the latter is called the 2-target case (Figure 6). Hence, the downstream and upstream nodes can be identified by the positive and negative k_i's respectively:

$$k_i^+ = \frac{1}{2}(k_i + |k_i|) = \max(0, k_i), \tag{31}$$

$$k_i^- = \frac{1}{2}(k_i - |k_i|) = \min(0, k_i). \tag{32}$$

For example, if k_1^+ is nonzero, the node 1 is a downstream node. Note that k_i's have the following properties:

$$k_i = k_i^+ + k_i^-, \quad \forall i, \tag{33}$$

$$|k_i| = k_i^+ - k_i^-, \quad \forall i, \tag{34}$$

$$\sum_{i \in \{i_T\}} k_i^+ = - \sum_{i \in \{i_T\}} k_i^- = \frac{1}{2} \sum_{i \in \{i_T\}} |k_i|, \tag{35}$$

$$\sum_{i \in \{i_T\}} k_i \mathbf{x_i} = \boldsymbol{\lambda} S_T, \tag{36}$$

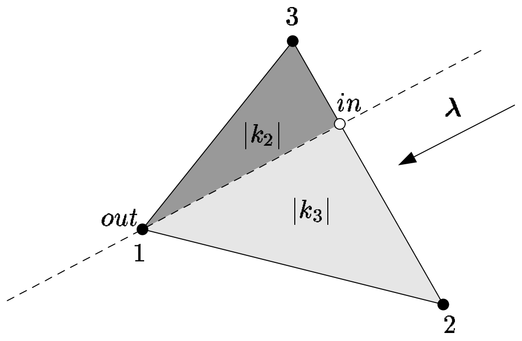

Figure 7: One target case: the two subtriangles have areas of $|k_2|$ and $|k_3|$.

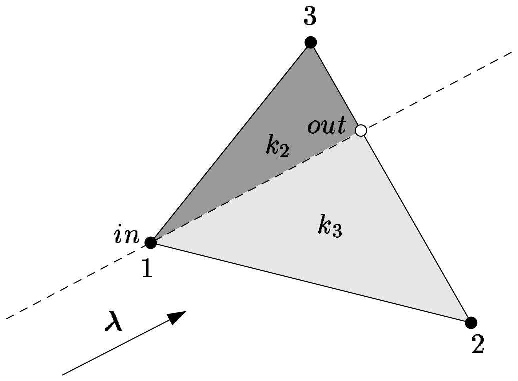

Figure 8: Two target case: the two subtriangles have areas of k_2 and k_3.

The last identity is obvious because the inflow parameter is closely related to the directional derivative taken along the vector $\boldsymbol{\lambda}$: for any variable, u, that varies linearly over the cell,

$$\int_T \boldsymbol{\lambda} \cdot \nabla u \, dxdy = \boldsymbol{\lambda} \cdot \left(\int_T \nabla u \, dxdy \right) = \boldsymbol{\lambda} \cdot \left(\frac{1}{2} \sum_{i \in \{i_T\}} u_i \mathbf{n}_i^T \right) = \sum_{i \in \{i_T\}} k_i u_i^T. \tag{37}$$

Using these positive and negative inflow parameters, we can express the locations of the inflow and outflow points in terms of the nodal coordinates,

$$\mathbf{x}_{in} = \sum_{i \in \{i_T\}} \delta_i^- \mathbf{x}_i, \quad \mathbf{x}_{out} = \sum_{i \in \{i_T\}} \delta_i^+ \mathbf{x}_i, \tag{38}$$

where

$$\delta_i^+ = \frac{k_i^+}{\displaystyle\sum_{i \in \{i_T\}} k_i^+}, \quad \delta_i^- = \frac{k_i^-}{\displaystyle\sum_{i \in \{i_T\}} k_i^-}. \tag{39}$$

Note that this is valid in both one-target and two-target cases, and that these are in fact in the form of the area-weighted averages whenever the sum is taken over two nodes: $\mathbf{x}_{in}$ in Figure 7; and $\mathbf{x}_{out}$ in Figure 8). In the same way, we can interpolate the solution $\mathbf{U}$ at these inflow and outflow locations,

$$\mathbf{U}_{in} = \mathbf{U}(\mathbf{x}_{in}) = \sum_{i \in \{i_T\}} \delta_i^- \mathbf{U}_i, \quad \mathbf{U}_{out} = \mathbf{U}(\mathbf{x}_{out}) = \sum_{i \in \{i_T\}} \delta_i^+ \mathbf{U}_i. \tag{40}$$

Also, note that the vector pointing from the inflow point to the outflow point is given by

$$\mathbf{x}_{out} - \mathbf{x}_{in} = \frac{S_T}{\displaystyle\sum_{i \in \{i_T\}} k_i^+} \boldsymbol{\lambda}, \tag{41}$$

where S_T is the cell area, and so its distance is given simply by its magnitude,

$$|\mathbf{x}_{out} - \mathbf{x}_{in}| = \frac{S_T}{\displaystyle\sum_{i \in \{i_T\}} k_i^+} |\boldsymbol{\lambda}|. \tag{42}$$

5 Multidimensional Reconstruction Schemes (Gnoffo's Method)

The standard numerical flux function is typically a function of two states, $\mathbf{U}_L$ and $\mathbf{U}_R$, and the normal direction $\mathbf{n}$:

$$\mathcal{F}_n = \mathcal{F}_n(\mathbf{U}_L, \mathbf{U}_R, \mathbf{n}). \tag{43}$$

Here, we consider the use of this standard numerical flux in the centroid discretization scheme:

$$\mathbf{\Phi}_j^T = -\frac{1}{2} \left(\mathcal{F}_x^T, \mathcal{F}_y^T \right) \cdot \mathbf{n}_j^T. \tag{44}$$

The numerical flux vector may be evaluated in arbitrary directions, say ξ and η, and rotated back to the standard coordinate directions x and y:

$$(\mathcal{F}_x^T, \mathcal{F}_y^T) = (\mathcal{F}_\xi^T \xi_x + \mathcal{F}_\eta^T \eta_x, \, \mathcal{F}_\xi^T \xi_y + \mathcal{F}_\eta^T \eta_y), \tag{45}$$

where ξ_x, ξ_y, η_x, and η_y are the x- and y-components of the unit vectors in ξ and η directions (see Figure 9). For example, Gnoffo takes the $\mathbf{n}_\xi$ along the density gradient (or the velocity if it is small). Then, the two states to be given to the numerical flux function may be defined along the chosen normal direction, and the required flux vector at the centroid can be completely determined. This particular scheme of Gnoffo can be implemented as residual-distribution(i.e., element-wise): for each cell $T \in \{T\}$, proceed as follows.

1. Set the normal direction along the density gradient (evaluated, for example, by the Green-Gauss formula):

$$\mathbf{n}_\xi = \frac{\nabla \rho}{|\nabla \rho|}. \tag{46}$$

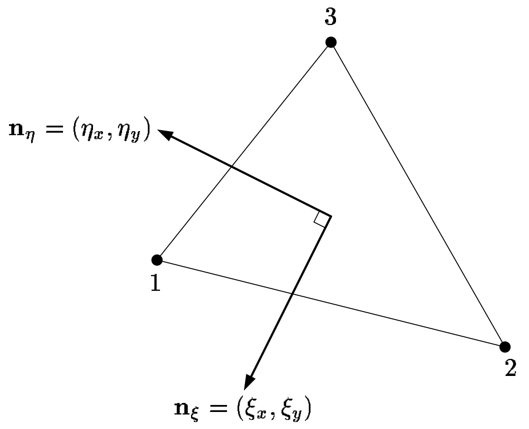
3
1
2
$\mathbf{n}_\eta = (\eta_x, \eta_y)$
$\mathbf{n}_\xi = (\xi_x, \xi_y)$

Figure 9: Directions for numerical flux evaluation.

2. Define the inflow parameters and δ_i's for all nodes $i \in \{i_T\} = \{1, 2, 3\}$:

$$k_i = \frac{1}{2} \mathbf{n}_\xi \cdot \mathbf{n}_i^T, \tag{47}$$

$$\delta_i^+ = \frac{k_i^+}{\sum_{i \in \{i_T\}} k_i^+}, \quad \delta_i^- = \frac{k_i^-}{\sum_{i \in \{i_T\}} k_i^-}. \tag{48}$$

3. Define the left and right states as the inflow and outflow states:

$$\mathbf{U}_L = \mathbf{U}_{in} = \sum_{i \in \{i_T\}} \delta_i^- \mathbf{U}_i, \quad \mathbf{U}_R = \mathbf{U}_{out} = \sum_{i \in \{i_T\}} \delta_i^+ \mathbf{U}_i, \tag{49}$$

and also the gradients if necessary,

$$\nabla \mathbf{U}_L = \nabla \mathbf{U}_{in} = \sum_{i \in \{i_T\}} \delta_i^- (\nabla \mathbf{U}_i), \quad \nabla \mathbf{U}_R = \nabla \mathbf{U}_{out} = \sum_{i \in \{i_T\}} \delta_i^+ (\nabla \mathbf{U}_i), \tag{50}$$

where $\nabla \mathbf{U}_i$'s are assumed to be available at nodes (e.g., via least-squares reconstruction).

4. Compute the numerical flux along $\mathbf{n}_\xi$ by the standard numerical flux function: for 1st-order accuracy

$$\mathcal{F}_\xi^T = \mathcal{F}_n(\mathbf{U}_L, \mathbf{U}_R, \mathbf{n}_\xi), \tag{51}$$

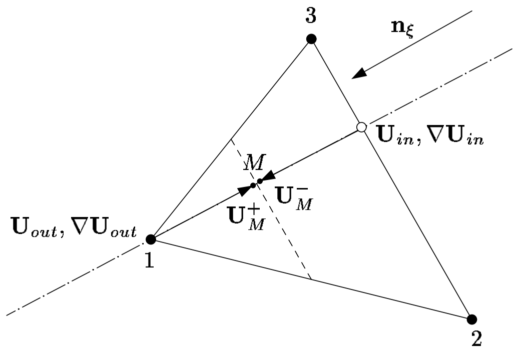
3
$\mathbf{n}_\xi$
$\mathbf{U}_{in}, \nabla \mathbf{U}_{in}$
M
$\mathbf{U}_M^-$
$\mathbf{U}_M^+$
$\mathbf{U}_{out}, \nabla \mathbf{U}_{out}$
1
2

Figure 10: Extrapolation of the solution value at the midpoint of the inflow and outflow points.

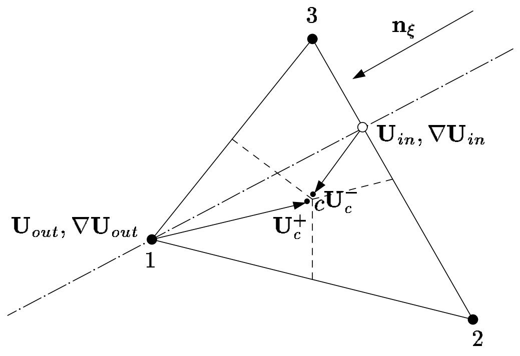
3
$\mathbf{n}_\xi$
$\mathbf{U}_{in}, \nabla \mathbf{U}_{in}$
c
$\mathbf{U}_c^-$
$\mathbf{U}_c^+$
$\mathbf{U}_{out}, \nabla \mathbf{U}_{out}$
1
2

Figure 11: Extrapolation of the solution value at the centroid from the inflow and outflow points.

or, for 2nd-order accuracy,

$$\mathcal{F}_\xi^T = \mathcal{F}_n(\mathbf{U}_M^-, \mathbf{U}_M^+, \mathbf{n}_\xi), \tag{52}$$

where $\mathbf{U}_M^-$ and $\mathbf{U}_M^+$ are extrapolated at the midpoint between the inflow and outflow points from the left and right states (See Figure 10).

5. Repeat the same for the orthogonal direction $\mathbf{n}_\eta$ and obtain $\mathcal{F}_\eta$.
6. Rotate the flux vectors:

$$(\mathcal{F}_x^T, \mathcal{F}_y^T) = (\mathcal{F}_\xi^T \xi_x + \mathcal{F}_\eta^T \eta_x, \, \mathcal{F}_\xi^T \xi_y + \mathcal{F}_\eta^T \eta_y). \tag{53}$$

7. Evaluate the update to each node:

$$\mathbf{\Phi}_i^T = -\frac{1}{2} \left(\mathcal{F}_x^T, \mathcal{F}_y^T\right) \cdot \mathbf{n}_i^T, \quad i = 1, 2, 3. \tag{54}$$

Note that the resulting scheme is not strictly 2nd-order accurate because the fluxes are in general not evaluated at the centroid. (It should be more accurate than the 1st-order schemes without reconstruction, though.) To be strictly 2nd-order accurate, the left and right states for the numeical flux function must be evaluated at the centroid: evaluate the flux (52) by the following two states extrapolated at the centroid (See Figure 11),

$$\mathbf{U}_c^- = \mathbf{U}_{in} + (\nabla \mathbf{U}_{in}) \cdot (\mathbf{x}_c - \mathbf{x}_{in}), \tag{55}$$
$$\mathbf{U}_c^+ = \mathbf{U}_{out} + (\nabla \mathbf{U}_{out}) \cdot (\mathbf{x}_c - \mathbf{x}_{out}). \tag{56}$$

6 Multidimensional Schemes II

In this section, we develop a class of schemes based on genuinely multidimensional dissipation. Unlike the previous schemes, which are based on a one-dimensional numerical flux function, the schemes we develop here are based on genuinely multidimensional residual-distribution schemes that are capable of resolving 'gaps' among three states. In the fnite-volume point of view, this means that we employ residual-distribution schemes to provide a proper dissipation by incorporating multidimensional physics directly into the interface flux (not using one-dimensional flux functions).

In one-dimension, it is easy to show that the first-order finite-volume upwind scheme and the residual-distribution upwind scheme are identical for linear hyperbolic conservation laws. This is well-known by the equivalence of the flux-difference splitting (residual-distribution) and flux-vector splitting methods. But for second-order schemes, there exists no such equivalence any more. For second-order accuracy, the finite-volume schemes focus on the accuracy of the interface flux while the residual-distribution schemes focus on the accuracy of flux-differences (cell-residuals) at a steady state. This comes from the fact that the interface flux loses its meaning in the residual-distribution schemes. This does not change anything in the first-order scheme but does bring a significant impact in the second-order scheme. Here, we attempt to bring residual-distribution schemes back into the finite-volume framework: identify its role as a dissipation and develop a class of new finite-volume schemes with residual-distribution schemes as a means to provide a multidimensional dissipation.

6.1 RD and FV in One Dimension

Consider one-dimensional conservation laws,

$$\frac{d\mathbf{u}}{dt} + \mathbf{f}_x = 0. \tag{57}$$

The standard finite-volume discretization results in

$$\Delta x \, \frac{d\mathbf{U}_i}{dt} = -\left[\mathbf{F}_{j+1/2} - \mathbf{F}_{j-1/2}\right], \tag{58}$$

where $\mathbf{U}_j$ is the cell average of $\mathbf{u}$ over the dual-volume centered at j, and the physical flux $\mathbf{f}$ has been replaced by a numerical flux $\mathbf{F}$. In Godunov's method, the flux is determined by solving the Riemann problem across the interface, and an upwind scheme is devised. The numerical flux can then be expressed in any of the following forms:

$$\begin{aligned} \mathbf{F}_{j+1/2} &= \mathbf{f}_j + \Delta\mathbf{f}_R^- && (59)\\ &= \mathbf{f}_{j+1} - \Delta\mathbf{f}_R^+, && (60) \end{aligned}$$

where the subscript R denotes the cell $[x_j, x_{j+1}]$, $\Delta \mathbf{f}_R^-$ and $\Delta \mathbf{f}_R^+$ are the changes in the flux created by the left-going and right-going waves respectively. We may also take the arithmetic average of the two,

$$\mathbf{F}_{j+1/2} = \frac{1}{2}(\mathbf{f}_j + \mathbf{f}_{j+1}) - \frac{1}{2}|\Delta \mathbf{f}_R|, \tag{61}$$

which is the most common form employed in practice to implement an upwind scheme. Suppose, instead, we use (59) for the interface on the right and (60) for the interface on the left.

$$\Delta x \frac{d\mathbf{U}_i}{dt} = -\left[(\mathbf{f}_j + \Delta \mathbf{f}_R^-) - (\mathbf{f}_j - \Delta \mathbf{f}_L^+)\right], \tag{62}$$

which reduces to

$$\Delta x \frac{d\mathbf{U}_i}{dt} = -\left[\Delta \mathbf{f}_R^- + \Delta \mathbf{f}_L^+\right], \tag{63}$$

where the subscript L denotes the cell on the left of j, i.e., $[x_{j-1}, x_j]$. In this form, the method can be interpreted differently. We start with the flux balance over the cell,

$$\int_{x_j}^{x_{j+1}} \mathbf{f}_x \, dx = \Delta \mathbf{f}, \tag{64}$$

which we decompose into the left-going and right-going waves,

$$\Delta \mathbf{f} = \Delta \mathbf{f}^+ + \Delta \mathbf{f}^-, \tag{65}$$

and distribute them to the left and right respectively,

$$\delta \mathbf{U}_j \leftarrow \delta \mathbf{U}_j + \Delta \mathbf{f}^-, \tag{66}$$

$$\delta \mathbf{U}_{j+1} \leftarrow \delta \mathbf{U}_{j+1} + \Delta \mathbf{f}^+. \tag{67}$$

At every node, collecting $\Delta \mathbf{f}^+$ from the left cell and $\Delta \mathbf{f}^-$ from the right cell, we arrive at the equation of the form (63). This is called flux-difference splitting, fluctuation-splitting, or residual-distribution. We have just shown that they are all equivalent to the standard finite-volume upwind scheme.

6.2 RD and FV in Two Dimensions

TBW.