A Framework for Discontinuous Fluctuation Distribution

Matthew Hubbard
Consider the scalar conservation law

\[ u_t + \nabla \cdot \vec{f} = 0 \quad \text{or} \quad u_t + \vec{\lambda} \cdot \nabla u = 0 \]

on a two-dimensional domain \( \Omega \).

- \( \vec{\lambda} = \left( \frac{\partial f}{\partial u}, \frac{\partial g}{\partial u} \right)^T \) is the advection velocity for \( \vec{f} = (f, g)^T \).
- \( u(x, y, 0) \) is specified.
- \( u(x, y, t) \) is specified on inflow boundaries.
- Additional diffusion/source terms will be ignored.

The main focus here will be on producing steady state solutions.
A range of methods exist which discretise the integral form of the equation, *e.g.*

- Finite volumes
- Finite elements
- Discontinuous Galerkin
- Fluctuation Distribution

These offer alternative approaches for representing the solution and distributing the information supplied by the equations.
Integrating the conservation law gives

\[ u_t + \nabla \cdot \vec{f} = 0 \quad \rightarrow \quad \int_{\Omega} u_t \, d\Omega + \int_{\Omega} \nabla \cdot \vec{f} \, d\Omega = 0 \]

- Assume that \( u \) is continuous.
- Attempt to integrate the equations exactly.
- Distribute the integrals between the unknowns.
- Conservation follows from the Gauss divergence theorem.
The Fluctuation

The fluctuation on a triangular cell ($\triangle$) is given by

$$\phi_c = - \int_{\triangle} \nabla \cdot \vec{f} \, d\Omega = - \int_{\triangle} \lambda \cdot \nabla u \, d\Omega = \oint_{\partial\triangle} u \lambda \cdot \vec{n} \, d\Gamma$$

- $\vec{n}$ gives the inward pointing normal to the cell boundary.
- It has been assumed that $u$ is stored at mesh nodes and varies linearly within each cell.
- In simple cases $\phi_c$ can be evaluated exactly using an appropriate conservative linearisation.
The aim is to solve the equations given by

\[ \vec{\lambda} \cdot \vec{\nabla} u = 0 \quad \rightarrow \quad \sum_{j \in U \Delta_i} \alpha_i^j (\phi_c)_j = 0 \quad \forall \text{ nodes } i \]

This is done iteratively, driving the \((\phi_c)_j\) to zero, by

- **distributing** each fluctuation \((\phi_c)_j\) to its adjacent nodes.
- **carefully choosing** the distribution coefficients \(\alpha_i^j\).
- **applying** a simple pseudo-time-stepping algorithm:

\[ u_i^{n+1} = u_i^n + \frac{\Delta t}{S_i} \sum_{j \in U \Delta_i} \alpha_i^j (\phi_c)_j \]
Ideally any fluctuation distribution scheme satisfies the following properties.

- **Conservative** – for discontinuity capturing.
- **Positive** – to prohibit unphysical oscillations.
- **Linearity Preserving** – for accuracy.
- **Continuous** – for convergence to steady state.
- **Compact** – for efficiency (and parallelism).
- **Upwind** – for physical realism.
Some Standard Scalar Schemes

The commonly used schemes are all conservative, continuous, compact and upwind.

- **N scheme**: positive but not linearity preserving, so it produces highly diffusive approximations.

- **LDA scheme**: linearity preserving but not positive, so it allows unphysical oscillations.

- **PSI scheme**: has all of the desired properties and can be constructed by limiting the distribution coefficients of the N scheme.
Forcing continuity across cell edges can be restrictive.

- It is difficult to change the representation locally, within mesh cells, since it has a knock-on effect on neighbours. This interferes with
  - conservation, particularly for nonlinear systems.
  - \( h \) - and \( p \) - adaptivity.
  - the “limiting” of high order schemes for positivity.
  - the creation of time-dependent schemes.
- Discontinuous flow cannot be represented exactly.
If $u$ is allowed to be discontinuous then

$$\int_{\Omega} \nabla \cdot \vec{f} \, d\Omega = \sum_{j=1}^{N_c} \int_{\Delta_j} \nabla \cdot \vec{f} \, d\Omega + \sum_{j=1}^{N_e} \int_{j} \nabla \cdot \vec{f} \, d\Omega$$

Discontinuous Galerkin would

- split the edge integrals into contributions from either side of the edge.
- associate these with their corresponding cells.

It is possible to consider these edge integrals separately.
Consider a mesh edge to be the limit of a rectangle as its width tends to zero, so

\[
\phi_e = - \lim_{\epsilon \to 0} \int_{\square_\epsilon} \nabla \cdot \vec{f} \, d\Omega
\]

\[
= \lim_{\epsilon \to 0} \int_{\partial \square_\epsilon} \vec{f} \cdot \vec{n} \, d\Gamma
\]

\[
= \int [\vec{f} \cdot \vec{n}] \, d\Gamma
\]

This is simply the integral along the edge of the flux difference across it.

Many finite volume schemes use the flux difference in the definition of the numerical flux.
Assuming that a conservative linearisation exists, integrating exactly using appropriate quadrature gives

\[ \phi_e = -\frac{1}{2} \sum_{k=1}^{N_q} w_k \tilde{\lambda}_k \cdot \tilde{n} [\tilde{u}_k], \]

which can be rewritten in terms of \( \hat{\lambda} \neq \tilde{\lambda} \) as

\[ \phi_e = \frac{1}{2} \hat{\lambda}_{12} \cdot \tilde{n} (u_1 - u_2) + \frac{1}{2} \hat{\lambda}_{43} \cdot \tilde{n} (u_4 - u_3) \]

Using Simpson’s rule gives

\[ \hat{\lambda}_{12} = \frac{1}{3} \left( \tilde{\lambda}_1 + \tilde{\lambda}_2 + \frac{\tilde{\lambda}_3 + \tilde{\lambda}_4}{2} \right) \quad \hat{\lambda}_{43} = \frac{1}{3} \left( \tilde{\lambda}_3 + \tilde{\lambda}_4 + \frac{\tilde{\lambda}_1 + \tilde{\lambda}_2}{2} \right) \]
A Positive Distribution

Upwinding gives the following positive distribution.

\[
S_1 u_1 \rightarrow S_1 u_1 + 3\Delta t \left[ \lambda_{12} \cdot \vec{n} \right]^-(u_1 - u_2)/2
\]

\[
S_2 u_2 \rightarrow S_2 u_2 + 3\Delta t \left[ \lambda_{12} \cdot \vec{n} \right]^+(u_1 - u_2)/2
\]

\[
S_3 u_3 \rightarrow S_3 u_3 + 3\Delta t \left[ \lambda_{43} \cdot \vec{n} \right]^+(u_4 - u_3)/2
\]

\[
S_4 u_4 \rightarrow S_4 u_4 + 3\Delta t \left[ \lambda_{43} \cdot \vec{n} \right]^-(u_4 - u_3)/2
\]

*Linearity preservation can be imposed but it’s not clear that this improves the approximation.*
Consider the limit of a scheme which mixes the usual triangles with rectangles of zero width. (Additional polygonal cells at vertices make no contribution.)

This leads to

\[ (u^j_i)^{n+1} = (u^j_i)^n + \frac{3\Delta t}{S_j} \left( \alpha^j_i (\phi_c)_j + \alpha^{k_1}_i (\phi_e)_{k_1} + \alpha^{k_2}_i (\phi_e)_{k_2} \right) \]

i.e. the cell and edge fluctuations are distributed to the local cell vertices.
Circular advection of the square wave profile with continuous (left) and discontinuous (right) PSI schemes.
Consider the two-dimensional inviscid Burgers’ equation

\[ u_t + \nabla \cdot \vec{f} = 0 \quad \rightarrow \quad u_t + \left( \frac{u^2}{2} \right)_x + u_y = 0 \]

with \( u(x, y, t) \) specified at inflow boundaries of a domain \( \Omega \).

The same procedure can be followed, except that

- \( \vec{\lambda} = (u, 1)^T \) now depends on the solution.
- appropriate linearisations must be carried out to retain conservation.
A Burgers’ equation solution (left) and computational mesh (right).
Burgers’ equation test case with continuous (left) and discontinuous (right) PSI schemes.
Consider the system of conservation laws
\[ \dot{U} + \nabla \cdot \vec{F} = 0 \quad \text{or} \quad \dot{U} + \tilde{A} \cdot \nabla U = 0 \]
on a domain \( \Omega \), where \( \tilde{A} \) represents the flux Jacobians.

Given an appropriate linearisation,
\[
\Phi_c = - \int_{\triangle} \nabla \cdot \vec{F} \, d\Omega = \oint_{\partial\triangle} \vec{F} \cdot \vec{n} \, d\Gamma \\
= - \int_{\triangle} \tilde{A} \cdot \nabla U \, d\Omega = -S_\triangle \tilde{A} \cdot \tilde{\nabla} U
\]

The fluctuation can be rewritten using a similarity transformation, i.e. \( \Phi_c = -S_\triangle \tilde{R} \tilde{A}' \tilde{R}^{-1} \cdot \tilde{\nabla} U \).
The cell-base fluctuation can be distributed in exactly the same manner as before, using one of

- wave decomposition models,
- matrix distribution schemes,

to deal with the nonlinear system.

- Both of these have been used successfully but the results shown are for the elliptic-hyperbolic wave decomposition of Roe and Mesaros (1995).
A Discontinuous Scheme

The edge fluctuations are

\[ \Phi_e = \int [\vec{F} \cdot \vec{n}] \, d\Gamma \]

\[ = \frac{1}{2} \hat{A}_{12} \cdot \vec{n} (\underline{U}_1 - \underline{U}_2) + \frac{1}{2} \hat{A}_{43} \cdot \vec{n} (\underline{U}_4 - \underline{U}_3) \]

This can be distributed directly or decomposed using Roe’s flux difference splitting, via

\[ \frac{\partial \vec{F}}{\partial \underline{U}} \cdot \vec{n} = \hat{A} \cdot \vec{n} = \hat{R} \hat{\Lambda} \hat{R}^{-1} \]

Upwinding is then applied through the Roe-averaged matrix \( \hat{\Lambda} \) and positivity can be maintained.
Constricted channel flow mesh and geometry. Flow is left to right, modelled using the Euler equations. The figures show the flow density.
Supercritical channel flow ($M_\infty = 2.0$) with continuous (left) and discontinuous (right) schemes.
Subcritical channel flow \((M_\infty = 0.5)\) with continuous (left) and discontinuous (right) schemes.
Transcritical channel flow ($M_\infty = 0.7$) with continuous (left) and discontinuous (right) schemes.
This scheme is
- conservative, positive, linearity preserving, compact, upwind and continuous (when the distribution is).

It is not, however,
- easy to converge to the steady state.
- time-accurate.
  - It can, though, be applied to the unsteady case and it no longer has to be upwind in time.
- as robust as the flux-based schemes (yet).
The Future

- Higher orders of accuracy (with positivity)
- Time-dependent problems
- Source terms
- $h$- and $p$-refinement
- Mesh movement to align with discontinuities
- Higher order derivatives
- Spherical polar coordinates
- Three space dimensions