A Semi-Lagrangian Discontinuous Galerkin (SLDG) Conservative Transport Scheme on the Cubed-Sphere

Ram Nair

Computational and Information Systems Laboratory (CISL)
National Center for Atmospheric Research (NCAR), Boulder, CO, USA.

Wei Guo

Department of Mathematics, University of Houston, TX, USA.

[PDEs-2012 Workshop, INI, Cambridge, UK. September 24th, 2012]
Motivation: Transport on the Spectral Element Cubed-Sphere Grid

- Required for high-order Galerkin methods such as spectral element (SE) and discontinuous Galerkin (DG) methods.
- Cubed-Sphere surface is partitioned into quadrilateral elements, each element has $N_g \times N_g$ quadrature points.
- Grid lines are highly nonuniform and non-orthogonal.
- Discontinuity at cube-face edges, vertices (8 weak singular points).

**Challenges:** Conservative transport with monotonicity (at least positivity),
Highly efficient, capable of multi-tracers (O(100)) transport?

**Possible Remedy:** Finite-volume semi-Lagrangian type transport. Eg. CSLAM in CAM (Erath et al., 2012).
However, this requires two grid systems, native SE grid and overlaid FV grid. Grid-to-grid conservative remapping is needed.
High-order Transport schemes: Numerical Method

1. Eulerian approach:
   - Fixed numerical mesh in space.
   - Arbitrary order of accuracy in space and time.
   - CFL restriction for stability with explicit time-stepping method.
     - e.g. $\text{CFL} \approx \frac{1}{2k+1}$ for RKDG when polynomial space of degree $k$ is used.

2. Semi-Lagrangian approach:
   - Combination of Eulerian approach and Lagrangian approach.
   - Fixed numerical mesh in space.
   - Numerical solution is updated by following trajectories.
   - Allowing very large CFL number.

Our Scheme:

A dimension-splitting semi-Lagrangian (SL) discontinuous Galerkin (DG) method on cubed-sphere geometry ($SL + DG = SLDG$).

- Our target is to employ SLDG with $\text{CFL} \approx 1$, this will not adversely affect parallel efficiency.
RKDG Weak Formulation: 1D Case

1D Scalar Transport Equation in Conservative (flux) Form:

\[ \phi_t + (u(x, t) \phi)_x = 0, \quad \text{in} \quad [a, b] \times (0, T), \]

- Domain discretization \([a, b] = \bigcup I_j = \bigcup [x_{j-1/2}, x_{j+1/2}]\).

- For regular RKDG (Runge-Kutta DG), the weak form is used for updating solution at \(t = t^{n+1}\)

\[
\frac{d}{dt} \int_{I_j} \phi_h(t) \psi dx - \int_{I_j} u(x, t) \phi_h \psi_x + [u(x, t) \phi_h \psi]^{x_{j+1/2}}_{x_{j-1/2}} = 0, \quad \forall \psi \in V^k,
\]

where \(\phi_h\) is the approximate solution, \(\psi\) is the test function and \(V^k\) is a vector space of polynomials up to degree \(k\).

- An approximate Riemann solver is used to resolve the flux discontinuity at the element edges \(I_j = [x_{j-1/2}, x_{j+1/2}]\)
For nodal RKDG method the basis set which spans $V^k$ is constructed using Lagrangian-Legendre polynomials $h_i(x)$ with roots at Gauss-Lobatto-Legendre (GLL) points.

The numerical solution is approximated by

$$\phi_h(x) = \sum_{i=0}^{k} \phi_i h_i(x).$$

Semi-discrete form is

$$\frac{d\phi_h}{dt} = L(\phi_i_h)$$
SLDG Weak Formulation: 1D Case

- SLDG 2D (linear transport) formulation Restelli et al. (JCP, 2006)
- Solving the Vlaso-Poisson system by Strang-split approach (Rossmanith & Seal (2011), Qui & Shu (2012))
- Theoretical analysis and link to the Lagrange-Galerkin method, Qui & Shu (2012)

Weak Formulation:

- In order to update the solution at \( t = t^{n+1} \), we let the test function satisfies the following adjoint problem:
  \[
  \begin{cases}
  \psi_t + u(x, t)\psi_x = 0, \\
  \psi(t^{n+1}) = \Psi(x).
  \end{cases}
  \]

- With the following equation, a weak SLDG form (Qiu & Shu, 2012) can be derived.

\[
\frac{d}{dt} \int_{l(t)_j} \phi(x, t)\psi(x, t)dx = 0
\]

\[
\int_{l_j^*} \phi^{n+1}\Psi dx = \int_{l_j^*} \phi(x, t^n)\psi(x, t^n)dx,
\]

where \( l_j^* = [x_j^{* - \frac{1}{2}}, x_j^{* + \frac{1}{2}}] \) with \( (x_j^{* \pm \frac{1}{2}}, t^n) \) are the feet of trajectories emanating from \( (x_j \pm \frac{1}{2}, t^{n+1}) \).
SLDG 1D Case Implementation

Evaluation of the R.H.S of \( \int_{I_j} \phi_{h}^{n+1} \psi dx = \int_{I_j^*} \phi_{h}(x, t^n) \psi(x, t^n) dx \):

1. Detect the cell boundary on \( I_j^* = \bigcup \ell I_{j, \ell}^* \) (\( \ell \) is the index of the sub-interval) by tracing trajectories backward in time.
2. Locate \((k + 1)\) GLL points (red circles) over \( I_{j, \ell}^* \) denoted as \( x_{j, \ell, m}^* \) (\( m \) is the index for GLL points).
3. Trace trajectories forward in time from \( x_{j, \ell, m}^* \) to \( x_{j, \ell, m} \) (green circles).
4. Evaluate the R.H.S. by

\[
\int_{I_j^*} \phi_{h}(x, t^n) \psi(x, t^n) dx = \sum_{\ell} \left( \sum_{m} w_{m} \phi_{h}(x_{j, \ell, m}^*, t^n) \Psi(x_{j, \ell, m}) \Gamma(I_{j, \ell}^*) \right).
\]

where \( w_{m} \) GLL quadrature weights, \( \Gamma(I_{j, \ell}^*) \) width of the element \( I_{j, \ell}^* \).
SLDG: Trajectories and Bound-Preserving Filters

- Trace of characteristics forward or backward in time: Taylor series method or Runge-Kutta method can be used for a given velocity field.
- Here we use 4th order Runge-Kutta method to solve:

\[ \frac{dx(t)}{dt} = u(x(t), t). \]

The bound-preserving (BP) Filter (or Positivity-Preservation):

- Extremely important for tracer transport ("Zero Tolerance" for negative values)
- The BP filter replaces the DG polynomial solution \( p_j(x) \) in each cell by the positive-definite solution \( \tilde{p}_j(x) \), without violating conservation

\[ \tilde{p}_j(x) = \theta p_j(x) + (1 - \theta)\bar{p}_j, \quad \theta = \min \left\{ \left| \frac{\tilde{p}_j}{m' - \bar{p}_j} \right|, 1 \right\}. \]


- \( m' \) is the local minimum within the local cell \( I_j \).
- \( \bar{p}_j \) is the cell average of the DG numerical solution.
- Local to the element, very efficient
Extension to 2D Case by Strang-Splitting

- 2D transport equation:

\[ \phi_t + (u(x, y, t)\phi)_x + (v(x, y, t)\phi)_y = 0 \]

- Split the 2D equation into two 1D equations (Strang-splitting \((\Delta t)^2\) accurate)

\[
\begin{align*}
\phi_t + (u(x, y, t)\phi)_x &= 0, \\
\phi_t + (v(x, y, t)\phi)_y &= 0
\end{align*}
\]

- Apply SLDG dimension-by-dimension manner

- In each of the rectangular cell, locate \((k + 1)\) GLL points in both \(x\) and \(y\) directions.

- Trajectories: \(dx/dt = u(x, y_0, t)\) and \(dy/dt = v(x_0, y, t)\)
Deformation Flow Test on 2D Cartesian (Blossey & Durran (2008))

- Deformation flow test of cosine bell on the 2D Cartesian domain.
- Initial distribution deforms into a crescent shape as it moves on the domain, and returns to the initial position when the flow reverses.
- SLDG $P^3$ (4 GLL points in each direction) is used in the simulation. The numerical solution is computed by $60 \times 60$ elements in $[-1, 1]^2$ domain.
- We choose CFL $\approx 0.95$, which is 6 times larger than that of RKDG.
- The BP filter optionally used
SLDG: Deformation Flow Test on 2D Cartesian without BP

- Initial (Final) Field (SLDG)
- Field at T/4 (SLDG)
- Field at 3T/4 (SLDG)
- Field at T/2 (SLDG)
SLDG: Deformation Flow Test on 2D Cartesian with BP

Initial (Final) Field (SLDG+BP)

Field at T/4 (SLDG+BP)

Field at 3T/4 (SLDG+BP)

Field at T/2 (SLDG+BP)

E_1 = 8.14e-5
E_2 = 2.72e-4
E_Inf = 3.00e-2
Extending SLDG to the Cubed-Sphere Geometry

The Cubed-Sphere Grid [Sadourny (1972); Rancic et al. (1996)]

- Quasi-uniform rectangular mesh, logically square cells, well suited for DG, SE methods
- Free of polar singularities. Non-orthogonal grid lines and discontinuous edges.
- Tensor form transport Eqn. (Nair et al. (2005)). The metric term $\sqrt{g}$ is known.

\[
\frac{\partial}{\partial t}(\sqrt{g} \phi) + \frac{\partial}{\partial x^1}(\sqrt{g} u^1 \phi) + \frac{\partial}{\partial x^2}(\sqrt{g} u^2 \phi) = 0; \quad x^1, x^2 \in [-\pi/4, \pi/4] \quad \Rightarrow
\]

\[
\frac{\partial \psi}{\partial t} + \frac{\partial F_1(\psi)}{\partial x^1} + \frac{\partial F_2(\psi)}{\partial x^2} = 0; \quad (F_1, F_2) : \text{Contravariant Fluxes}, \quad \psi = \sqrt{g} \phi.
\]
For SLDG, extend the gnomonic grid-lines $x^1$ or $x^2$ beyond the edges, to create 1D halo region.

Trajectories are computed along $x^2$ or $x^1$ direction from contravariant wind (RK4)

$$
\frac{dx^1}{dt} = u^1(x^1, x^2, t), \quad \frac{dx^2}{dt} = u^2(x^1, x^2, t)
$$
SLDG: Solid-Body Rotation of a Cosine-bell (Williamson et al. 1992)

SLDG: Advection of a Cosine-Bell

- SLDG \( P^3 \) is used for simulation on a \( 20 \times 20 \times 6 \) element grid.
- Time step \( \Delta t = 1 \) hour, it takes 12 model days to complete a full rotation. Flow orientation is set to NE direction (or \( \alpha = \pi/4 \)).
- The \( \ell_1, \ell_2 \) errors are comparable with that of RKDG. However, \( \Delta t \) chosen was 5 times larger than that of RKDG.
- The \( \ell_\infty \) error is larger for SLDG due to the splitting-error at the corners of the cube.
SLDG: Cosine-Bell Test. Time traces of $\ell_1$, $\ell_2$, $\ell_\infty$ Errors

SLDG: Advection of a Cosine-Bell

(a) SLDG, $\alpha=\pi/4$

(b) SLDG+BP, $\alpha=\pi/4$

(c) SLDG, $\alpha=0$

(d) SLDG+BP, $\alpha=0$

Lauritzen et al. (2010)
SLDG vs. RKDG (Cosine-Bell Test)  Error & Efficiency Comparison

Table: 1. Comparison between RKDG and SLDG (30 × 30 × 6 and $\alpha = 0$)

<table>
<thead>
<tr>
<th>Scheme</th>
<th>time step</th>
<th>CPU time</th>
<th>$L_2$ error</th>
<th>$L_\infty$ error</th>
</tr>
</thead>
<tbody>
<tr>
<td>RKDG $P^3$</td>
<td>1440 sec</td>
<td>9.76 sec</td>
<td>1.14e−02</td>
<td>7.55e−03</td>
</tr>
<tr>
<td>SLDG $P^3$</td>
<td>7200 sec</td>
<td>3.32 sec</td>
<td>3.70e−03</td>
<td>4.33e−03</td>
</tr>
<tr>
<td>SLDG $P^3$</td>
<td>3600 sec</td>
<td>6.31 sec</td>
<td>2.80e−03</td>
<td>3.12e−03</td>
</tr>
</tbody>
</table>

Table: 2. Comparison between RKDG and SLDG (30 × 30 × 6 and $\alpha = \pi/4$)

<table>
<thead>
<tr>
<th>Scheme</th>
<th>time step</th>
<th>CPU time</th>
<th>$L_2$ error</th>
<th>$L_\infty$ error</th>
</tr>
</thead>
<tbody>
<tr>
<td>RKDG $P^3$</td>
<td>1440 sec</td>
<td>9.76 sec</td>
<td>1.21e−02</td>
<td>7.97e−03</td>
</tr>
<tr>
<td>SLDG $P^3$</td>
<td>7200 sec</td>
<td>3.32 sec</td>
<td>1.72e−02</td>
<td>3.56e−02</td>
</tr>
<tr>
<td>SLDG $P^3$</td>
<td>3600 sec</td>
<td>6.31 sec</td>
<td>5.15e−03</td>
<td>9.25e−03</td>
</tr>
</tbody>
</table>

In general, SLDG is more computationally efficient than RKDG for a given order of accuracy.
SLDG: Moving Vortex Test *(Nair & Jablonowski (2008))*

- Exact solution is known, and 12 days to complete a full rotation \((\alpha = \pi/4)\)
- SLDG $P^3$ is used for simulation on a $30 \times 30 \times 6$ grid, $\Delta t = 1$ hour
- Smooth evolution of vortex fields across the vertices of the cubed-sphere
Moving Vortex Test: Time traces of $\ell_1$, $\ell_2$, $\ell_\infty$ Error.

- Time traces of normalized errors are comparable with that of CSLAM ([Lauritzen et al.](https://advances.cis.ncar.gov/jcp-2010))

- The error is comparable to RKDG, but the time step is chosen as 6 times larger than that of RKDG.
For this test, the final solution is known and is exactly equal to the initial condition.

SLDG $P^3$ is used for simulation on a $30 \times 30 \times 6$ grid.

Time step is chosen as 0.0625 such that it takes 800 steps to complete a full evolution.

Results are comparable to those with RKDG, but the time step is 6 times larger.
Summary & Conclusions

Conclusions:

- The semi-Lagrangian discontinuous Galerkin (SLDG) scheme has been extended to the cubed-sphere geometry.
- SLDG employs a sequence of 1D operations to solve 2D transport problem (Strang split) on the cubed-sphere.
- The scheme is accurate, positivity preserving, and can take significantly longer time step as compared to the RKDG.
- Computational overhead associated with the trajectory algorithm is negligible in the context of multi-tracer transport for global models (such as CAM/HOMME).
- SLDG with CFL ≈ 1 is a viable option for multi-tracer transport on SE cubed-sphere grid.
- No need for the dual finite-volume grid, tracer transport and dynamics can be made consistent.

Future Work:

- More robust treatment on the cubed-sphere edges to minimize $\ell_\infty$ error.
- Apply SLDG scheme to nonlinear equations or systems (Shallow water equation, Euler system, etc.)