Conservative Semi-LAgrangian Multi-tracer scheme (CSLAM):
Lagrangian, Rigorous Flux-Form & Simplified Flux-Form version

Peter Hjort Lauritzen

Atmospheric Modeling & Predictability Section
Climate & Global Dynamics Division
National Center for Atmospheric Research, Boulder, Colorado
Finite-volume Lagrangian form of continuity equation for $\psi = \rho$, $\rho \phi$:

$$
\int_{A_k} \psi_{k+1}^n \, dx \, dy = \int_{a_k} \psi^n_k \, dx \, dy = \sum_{\ell=1}^{L_k} \int \int_{a_{k\ell}} f_\ell(x, y) \, dx \, dy,
$$

where the $a_{k\ell}$'s are non-empty overlap regions:

$$
a_{k\ell} = a_k \cap A_\ell, \quad a_{k\ell} \neq \emptyset; \quad \ell = 1, \ldots, L_k. \quad (1)
$$

and

$$
\rho = \text{air density},
$$

$$
\phi = \text{mixing ratio}.
$$
Finite-volume Lagrangian form of continuity equation for $\psi = \rho, \rho \phi$:

$$\int_{A_k} \psi_{k+1}^{n+1} \, dx \, dy = \int_{a_k} \psi_k^n \, dx \, dy = \sum_{\ell=1}^{L_k} \oint_{\partial a_{k\ell}} \left[ P \, dx + Q \, dy \right],$$

where $\partial a_{k\ell}$ is the boundary of $a_{k\ell}$ and

$$-\frac{\partial P}{\partial y} + \frac{\partial Q}{\partial x} = f_\ell(x, y) = \sum_{i+j \leq 2} c_\ell^{(i,j)} x^i y^j.$$

Use 2D polynomials of degree 2.
Finite-volume Lagrangian form of continuity equation for $\psi = \rho, \rho \phi$:

$$\int_{A_k} \psi_{k}^{n+1} \, dx \, dy = \int_{a_k} \psi_{k}^{n} \, dx \, dy = \sum_{\ell=1}^{L_k} \left[ \sum_{i+j \leq 2} c_{\ell}^{(i,j)} w_{k\ell}^{(i,j)} \right],$$

where weights $w_{k\ell}^{(i,j)}$ are functions of the coordinates of the vertices of $a_{k\ell}$.
Finite-volume Lagrangian form of continuity equation for $\psi = \rho, \rho \phi$:

$$\int_{A_k} \psi_{k}^{n+1} \, dx \, dy = \int_{a_k} \psi_{k}^{n} \, dx \, dy = \sum_{\ell=1}^{L_k} \left[ \sum_{i+j \leq 2} c_{\ell}^{(i,j)} \, w_{k\ell}^{(i,j)} \right] ,$$

- $w_{k\ell}^{(i,j)}$ can be re-used for each additional tracer (Dukowicz and Baumgardner, 2000)
- CSLAM is stable for long time-steps (CFL $> 1$), however,
  - remap breaks down if upstream cell is not simply-connected (trajectory algorithm will most likely ‘breakdown’ before that happens!)
- fully 2D method: geometrically flexible (can with relative ease be extended to other grids)
Extension to the cubed-sphere

In this Section we first discuss how the panel boundaries are treated in CSLAM. The mechanism for mass flux exchange between panels is then presented and finally we derive the spherical line integral formulae.

3.2.1. Departure cells

All computations are performed on the gnomonic projection in \((x, y, m)\)-coordinates so that the algorithm for Cartesian geometry described earlier can be employed. As in the Cartesian case we connect the departure points with straight line segments. As mentioned previously, by doing so in the gnomonic projection the sides of the departure cells are great-circle arcs on the sphere. For cells that stay completely on a panel when being transported by the flow (for one time-step) the overlap areas \(a_k\) are defined exactly as in the Cartesian case. The question then becomes how to deal with the cells that traverse the edges of the cube. Since the CSLAM scheme is fully two-dimensional it is possible to treat cells that cross panel edges in a rigorous two-dimensional manner that adds a minor complexity to the algorithm as compared to the Cartesian case.

For a particular panel \(m\) we introduce a halo zone around the panel and treat the halo cells on the same projection as panel \(m\) (Fig. 9). An algorithm for identifying indices of neighboring cells across panel sides is, for example, given in [27]. As an illustrative example consider a 1-cell halo zone and a resolution of \(N_c = 9\). Fig. 9(a) shows the Eulerian cells on the gnomonic projection for panel \(m\) (solid lines) as well as the halo cells (dashed lines). Since the sides of any grid cell on the cubed-sphere are great-circle arcs also the halo cell sides are straight lines on panel \(m\’s\) gnomonic projection. The halo cell sides are, however, not necessarily aligned with panel \(m\’s\) grid lines.

We compute the departure points for the grid cell vertices on panel \(m\) as well as for the grid cell vertices of the halo zone cells. The departure points connected by straight lines are shown on Fig. 9(b).

Next we restrict the overlap areas \(a_k\) to panel \(m\):

\[
a_m(k) = a_k \cap \bigcup_{(m)} \quad (1)
\]

so that the panel \(m\) restricted departure area is given by

\[
R(\lambda, \theta) = \int_0^{\lambda} \int_{-\beta}^{\beta} \int_{-\alpha}^{\alpha} f(\rho) \, d\rho \, d\beta \, d\alpha
\]

where \(\alpha, \beta \in [-\pi/4, \pi/4]\).

Note: straight line on the gnomonic projection = great-circle arc on the sphere!
In this Section we first discuss how the panel boundaries are treated in CSLAM. The mechanism for mass flux exchange...
Finite-volume flux-form of continuity equation for $\psi = \rho, \rho \phi$:

$$
\int_{A_k} \psi_{n+1}^k \, dx \, dy = \int_{A_k} \psi_n^k \, dx \, dy - \sum_{\epsilon=1}^{4} \left[ \sum_{\ell=1}^{L_{k\epsilon}} L_{k\epsilon}^\epsilon \int \int_{a_k^\epsilon \ell} f_{\ell}(x, y) \, dx \, dy \right], \quad (1)
$$

where

- $a_k^\epsilon = \text{‘flux-area’ (yellow area) = area swept through face } \epsilon$
- $L_{k\epsilon}^\epsilon = \text{number of overlap areas for } a_k^\epsilon; a_k^\epsilon \ell = a_k^\epsilon \cap A_k$
- $s_{k\epsilon}^\epsilon = 1 \text{ for outflow and } -1 \text{ for inflow}$.

All technology developed for CSLAM can be re-used.
Finite-volume flux-form of continuity equation for $\psi = \rho, \rho \phi$:

$$
\int_{A_k} \psi_k^{n+1} \, dx \, dy = \int_{A_k} \psi_k^n \, dx \, dy - \sum_{\epsilon=1}^{4} \left[ \sum_{\ell=1}^{L_k} s_{k\ell}^\epsilon \int \int_{a_{k\ell}} f_{\ell}(x, y) \, dx \, dy \right], \quad (1)
$$

unlimited FF-CSLAM = unlimited CSLAM: Why FF-CSLAM?

- You get mass-conservation no matter how you approximate fluxes!
- In CSLAM shape-preservation is enforced by filtering the sub-grid-cell reconstructions (also applicable for FF-CSLAM)
- In flux-form one may also apply flux-limiters such as FCT (Flux-Corrected Transport, Zalesak 1979).
- Flux-form easily allows for sub-cycling: $\Delta t_{\text{tracer}} = k \Delta t_{\text{air}}$
- Drawback: FF-CSLAM is more expensive than CSLAM (especially for CFL > 1)
For CFL \leq \frac{1}{2} the results may improve. **Very counterintuitive** (less rigorous and cheaper scheme can be better)!

- Von Neumann stability analysis and linear error analysis confirms this!
Limiter/filter to enforce shape-preservation

- In the literature: many 1D filters but few fully 2D filters
- Use simple 2D reconstruction function filter (Barth and Jespersen, 1989):
  - scale $f_\ell(x, y)$ so that extreme values lie within the adjacent cell-average values

\[
\text{no filter} \quad \text{monotone filter}
\]
\[ \Delta t = T/1200 \text{ (max CFL approximately 0.5)} \]

Results for 1.5° resolution: (R=rigorous FF-CSLAM, S=simplified FF-CSLAM, F=filter)

<table>
<thead>
<tr>
<th></th>
<th>( l_1 )</th>
<th>( l_2 )</th>
<th>( l_\infty )</th>
<th>( l_r )</th>
<th>( l_u )</th>
<th>( l_o )</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>( 5.65 \times 10^{-2} )</td>
<td>( 1.12 \times 10^{-1} )</td>
<td>( 1.55 \times 10^{-1} )</td>
<td>( 1.67 \times 10^{-3} )</td>
<td>( 1.84 \times 10^{-4} )</td>
<td>( 1.67 \times 10^{-3} )</td>
</tr>
<tr>
<td>S</td>
<td>( 5.51 \times 10^{-2} )</td>
<td>( 1.11 \times 10^{-1} )</td>
<td>( 1.50 \times 10^{-1} )</td>
<td>( 2.03 \times 10^{-3} )</td>
<td>( 1.94 \times 10^{-4} )</td>
<td>( 1.68 \times 10^{-3} )</td>
</tr>
<tr>
<td>R F</td>
<td>( 6.97 \times 10^{-2} )</td>
<td>( 1.82 \times 10^{-1} )</td>
<td>( 3.18 \times 10^{-1} )</td>
<td>( 2.51 \times 10^{-3} )</td>
<td>( 2.37 \times 10^{-5} )</td>
<td>( 0.0 )</td>
</tr>
<tr>
<td>S F</td>
<td>( 6.95 \times 10^{-2} )</td>
<td>( 1.83 \times 10^{-1} )</td>
<td>( 3.23 \times 10^{-1} )</td>
<td>( 2.81 \times 10^{-3} )</td>
<td>( 2.52 \times 10^{-5} )</td>
<td>( 0.0 )</td>
</tr>
</tbody>
</table>

- In terms of \( l_1, l_2, l_\infty \): S is more accurate than R, however, not so for \( l_r, l_u, l_o \)
- With filter S is less accurate than R.
- Simplified scheme (S) is, however, significantly simpler to code and requires no searching for overlap areas!!!
Δt = T/2400 (max CFL approximately 0.5)

Results for 0.75° resolution: (R=rigorous FF-CSLAM, S=simplified FF-CSLAM,F=filter)

<table>
<thead>
<tr>
<th></th>
<th>$l_1$</th>
<th>$l_2$</th>
<th>$l_\infty$</th>
<th>$l_r$</th>
<th>$l_u$</th>
<th>$l_o$</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>$9.19 \times 10^{-3}$</td>
<td>$2.03 \times 10^{-2}$</td>
<td>$3.83 \times 10^{-2}$</td>
<td>$4.72 \times 10^{-4}$</td>
<td>$1.46 \times 10^{-4}$</td>
<td>$2.36 \times 10^{-4}$</td>
</tr>
<tr>
<td>S</td>
<td>$8.72 \times 10^{-3}$</td>
<td>$1.96 \times 10^{-2}$</td>
<td>$3.72 \times 10^{-2}$</td>
<td>$4.62 \times 10^{-4}$</td>
<td>$1.46 \times 10^{-4}$</td>
<td>$2.28 \times 10^{-3}$</td>
</tr>
<tr>
<td>R F</td>
<td>$1.01 \times 10^{-2}$</td>
<td>$3.36 \times 10^{-2}$</td>
<td>$1.09 \times 10^{-1}$</td>
<td>$3.80 \times 10^{-4}$</td>
<td>$1.10 \times 10^{-4}$</td>
<td>$0.0$</td>
</tr>
<tr>
<td>S F</td>
<td>$1.05 \times 10^{-2}$</td>
<td>$3.45 \times 10^{-2}$</td>
<td>$1.11 \times 10^{-1}$</td>
<td>$3.73 \times 10^{-4}$</td>
<td>$1.18 \times 10^{-4}$</td>
<td>$0.0$</td>
</tr>
</tbody>
</table>

- In terms of $l_1$, $l_2$, $l_\infty$ as well as $l_r$, $l_u$, $l_o$: S is more accurate than R
- With filter S is less accurate than R (except for $l_r$).
- Simplified scheme (S) is, however, significantly simpler to code and requires no searching for overlap areas!!!
Questions


