DART Perspectives on Infrastructure Challenges of High Resolution and Coupled Data Assimilation

Kevin Raeder (NCAR: CAM+DART development)

Contributors:

Alicia Karspeck (NCAR: POP+DART)

NCAR: DART software:
Jonathan Hendricks, Helen Kershaw
Tim Hoar, Nancy Collins,
Jeffrey Anderson (NCAR: DART PI)
DART overview

- Data Assimilation Research Testbed
  - Ensemble Kalman Filter
  - Research
    - Dozens of model and observation interfaces
    - Good parallel performance on many machines (not tuned like operational)
    - Education via tutorials
  - Software Engineering Focus
    - Standardized, small interface to any sized model
    - Flexibility; choice of model, any relevant observations, filter
    - Tools which make EnKF work well with large models, ‘small’ ensembles.
    - Make DA accessible to non-DA experts

- CESM+DART
  - Community Earth System Model; multi-component climate model
  - Ensemble capability enables DART within CESM framework
Status of Cross-Component Assimilation with CESM

- "Coupled": model is ocean+atmosphere+...
- "Multi-component DA": DART can assimilate ‘native’ observations into each component.
- "Cross-component” = any observation can influence any model variable
- Bigger state vector requires Remote Memory Access version of DART.
- RMA only available for POP2 interface so far.
- Research into localization between components needs to be done.
- Resolve ambiguity of an observation.
- Learn importance of cross-component.

Significant software engineering is needed to adapt larger, more complex state vectors in DART to existing computing environments.
Our Challenges

1) Data motion: I/O and internal.
2) Memory: each member must be distributed over nodes.
3) Complexity: cross-component DA -> more communication.
4) Robustness: more processors leads to more job failures.
5) External storage and access: huge output data sets.
6) Software infrastructure support: harder to fund than hardware and science.
State vector (T,U,...) size
- > largest 32 bit integer: convert integers to 64 bit
- = 17 Gb (2/3 of a Yellowstone node)

Static data is stored on each processor for speed+convenience
- 553 Mb/proc x 16 procs/node = 9 Gb/node (1/3 of a Y node)

- Each state vector *is* distributed across multiple nodes:
  - internode communication for many observation forward operators,
  - book-keeping of which node has which parts of the state vector.

- The static memory *will be* distributed across nodes, with the same consequences.
0.1 degree POP2 on NCAR’s Yellowstone: Distributed memory Forward operators

Potentially any part of the state could be needed to calculate an observation forward operator. *(This is why in regular DART we transpose so we have a whole state vector on one process)*

The state is distributed across all processors.

Using MPI one-sided communication a task can get the state elements it needs from any other process. This happens asynchronously. *(Asynchronously is the key point – you don’t have to calculate which processors need what from where and when they need it)*
Examples from POP2 0.1 Degree: 
I/O intensive

30 members -> 516 Gb of state vectors read and written each assimilation. 
4 Gb/s for read + movement to where it’s needed. 
I/O is a significant fraction of the assimilation time, and it doesn’t scale well.
Examples from POP2 0.1 Degree:
Potential I/O Accelerations

Possibly:

- Pass state between model and DA via memory, rather than files.
- Read in only state values which will be affected by assimilation.
- POP2: Run ensembles with a 1° model to constrain the planetary scales and use a stationary ensemble OI to constrain the mesoscale variability.
Example from 1 Degree, Multi-Component Assimilation: Reliability

- Trying to run CESM on a petascale machine (other than Yellowstone) has been frustrating;
  - Job (~1 degree, coupled model, 30 members, DA) not big enough to get high priority => days in the queue. Queueing policy may not be suited to short jobs on lots of processors.
  - >1/3 of all CESM jobs run by 3 NCAR people fail due to machine problems, must be restarted. This will be worse for larger models. This points to the need for more robust machines and/or software to recover from a variety of failures.
Other Potential Solutions

- Multiple couplers in CESM (1 for each member).
- Compute forward operators within the model, rather than afterwards in the DA.
- Create a true ensemble coupled model?
  - Separate model members are replaced by a new dimension (“member”) in each state variable. A given computation would be performed for all members at the same time. That would move larger chunks of data fewer times.
Summary and Conclusions

- For ‘high’ resolution, significant software infrastructure development in CESM and DART is needed to even use currently available computing resources effectively.
- The trends of increasing processor count, decreasing shared memory/node, and increasing data set sizes will require ongoing software infrastructure development.
- Large model development in CESM+DART benefits greatly from true software engineers, rather than (talented) DA scientists.
Timing Assimilation for High Resolution POP

- 30 member ensemble
- 16515 observations
- cutoff 0.2 ~ 1300 km
- ptile = 16

<table>
<thead>
<tr>
<th># PEs</th>
<th>Read Time</th>
<th>Assim. Time</th>
<th>Write Time</th>
<th>Total Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>2048</td>
<td>301</td>
<td>1940</td>
<td>444</td>
<td>2697</td>
</tr>
<tr>
<td>4096</td>
<td>276</td>
<td>977</td>
<td>449</td>
<td>1711</td>
</tr>
<tr>
<td>8192</td>
<td>243</td>
<td>494</td>
<td>281</td>
<td>1029</td>
</tr>
</tbody>
</table>

* Time measured in seconds
- Designed for ~state-of-the-art, full data assimilation. Can be down-graded for computational savings.
- Ensemble information is valuable, especially for longer range forecasts.