Applying AD to the Community Land Model

Laboratory for Advanced Numerical Simulations
Mathematics and Computer Science
Argonne National Laboratory

Azamat Mametjanov, Boyana Norris,
Xiaoyan Zeng, Beth Drewniak,
Jean Utke, Mihai Anitescu, and Paul Hovland

Supported by DOE’s BER under Climate Science for Sustainable Energy Future (CSSEF)
Outline

CESM and CLM

AD and OpenAD

AD Process

Results
CESM and CLM
CESM: Community Earth System Model

- Is a global model for simulations of the climate system.
- Is composed of five fully-coupled components of
  1. atmosphere
  2. land
  3. ocean
  4. land-ice
  5. sea-ice
- Each component is configurable for one of the modes
  - active (fully prognostic)
  - data (intercomponent data cycling)
  - inactive (interface)
- Provides state-of-the-art simulations of past, present and future climate states (1850–2100)
CLM: Community Land Model

Is a CESM component for simulations of

1. energy fluxes in land bio-geophysics
2. chemical compound fluxes in land bio-geochemistry
3. water fluxes in land hydrology

Divides the modeled surface grid into grid-cells

- land units
  - columns
    - plant functional types (PFT)
CLM: Structure

Land unit models
  ▶ glacier
  ▶ lake
  ▶ wetland
  ▶ urban area
  ▶ vegetated area

Column models vertical layers of
  ▶ soil – up to 15 layers
  ▶ snow – up to 5 layers

Plant functional type models
  ▶ trees and shrubs – 11 types
  ▶ grasses – 3 types
  ▶ crops – 1 type
CLM extension: CLM-Crop

Extends CLM with irrigated/managed crop species

- corn
- wheat
- soybean

Models pools and fluxes of Carbon and Nitrogen (CN) in

- leaves
- stems
- roots
- harvested organs
CLM-Crop

Vertical Mixing & Transport:

$$\frac{\partial C}{\partial t} = \frac{\partial}{\partial t} \left( D \frac{\partial C}{\partial z} \right) - \frac{\partial}{\partial t} \left( uC \right) + R$$
CLM-Crop Parameter Calibration

- Crop growth is substantially affected by CN fluxes
- CN fluxes are determined by CN ratios of a given crop species
- There is a large uncertainty in regards to CN ratios
- We study sensitivity of CN ratios to calibrate the parameters for further improvement of model accuracy
AD and OpenAD
Computation encoded by a program can be broken into a sequence of elemental operations.

Independent (and varied) variable values are transformed into (useful and) dependent variable values.

AD computes derivatives of elemental operations acting on active variables.

Based on the chain rule, derivative values are accumulated to obtain a derivative of the entire program.
OpenAD: modular Fortran/C/C++ AD tool
AD Process
## CLM source code

<table>
<thead>
<tr>
<th>Modules</th>
<th>Files</th>
<th>Lines</th>
</tr>
</thead>
<tbody>
<tr>
<td>Biogeochemistry</td>
<td>39</td>
<td>18568</td>
</tr>
<tr>
<td>Biogeophysics</td>
<td>24</td>
<td>13931</td>
</tr>
<tr>
<td>Riverroute</td>
<td>3</td>
<td>1435</td>
</tr>
<tr>
<td>Couplers/drivers</td>
<td>53</td>
<td>34416</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>119</strong></td>
<td><strong>68350</strong></td>
</tr>
</tbody>
</table>
CLM module dependencies
Call graphs

- Help to understand control flow dependencies
- CESM is supported by PGI compiler suite
- Ported to GNU suite to profile with gprof
AD pattern: start small, iterative increments

- AD is excellent when applied to numeric programming of smooth functions
- Real codes represent complex compositions of (non-smooth) computations
- Pattern
  - start with a small numerical core $C$
  - differentiate: $C'$
  - validate $C'$
  - add more functions/subroutines
  - repeat until the desired scope is achieved
Local AD: CNAllocation module
OpenAD work-flow

- Identify module dependencies
- Collect source files
- Preprocess with proper build sequence parameters
- Concatenate the sources into a single file
- Insert annotations marking independent and dependent variables
- Invoke transformations:
  1. preprocess
  2. fortran → whirl
  3. whirl → xaif
  4. xaif → xaif’
  5. xaif’ → whirl’
  6. whirl’ → fortran’
  7. postprocess
Results
AD of CNAlocation calculation

12 inputs
- final leaf/stem/root CN ratio
- leaf/organ/live-wood/dead-wood CN ratio
- new fine-root/stem C per new leaf C
- new coarse-root C per new stem C
- fraction of new wood that is live
- fraction of allocation that goes to current growth

6 outputs
- leaf/stem/organ carbon/nitrogen
## Select derivative results

<table>
<thead>
<tr>
<th>ORGAN</th>
<th>C</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>fleafcn</td>
<td>1277.0876953</td>
<td>25.5417539</td>
</tr>
<tr>
<td>frootcn</td>
<td>809.0263835</td>
<td>16.1805277</td>
</tr>
<tr>
<td>fstemcn</td>
<td>350.6178413</td>
<td>7.0123568</td>
</tr>
<tr>
<td>deadwdcn</td>
<td>0.0000000</td>
<td>0.0000000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>STEM</th>
<th>C</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>fleafcn</td>
<td>14.4554170</td>
<td>0.2891083</td>
</tr>
<tr>
<td>frootcn</td>
<td>9.4894822</td>
<td>-12.9865931</td>
</tr>
<tr>
<td>fstemcn</td>
<td>3.9686602</td>
<td>-25.9019159</td>
</tr>
<tr>
<td>deadwdcn</td>
<td>0.0000000</td>
<td>0.0000000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LEAF</th>
<th>C</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>fleafcn</td>
<td>7.0353917</td>
<td>-93.0305059</td>
</tr>
<tr>
<td>frootcn</td>
<td>4.6136744</td>
<td>-46.2578484</td>
</tr>
<tr>
<td>fstemcn</td>
<td>1.9315305</td>
<td>0.1931531</td>
</tr>
<tr>
<td>deadwdcn</td>
<td>0.0000000</td>
<td>0.0000000</td>
</tr>
</tbody>
</table>

### CORN

<table>
<thead>
<tr>
<th>ORGAN</th>
<th>C</th>
<th>N</th>
</tr>
</thead>
<tbody>
<tr>
<td>fleafcn</td>
<td>1277.0876953</td>
<td>25.5417539</td>
</tr>
<tr>
<td>frootcn</td>
<td>809.0263835</td>
<td>16.1805277</td>
</tr>
<tr>
<td>fstemcn</td>
<td>350.6178413</td>
<td>7.0123568</td>
</tr>
<tr>
<td>deadwdcn</td>
<td>0.0000000</td>
<td>0.0000000</td>
</tr>
</tbody>
</table>
Lessons Learned

CESM application
- Highly structured, hierarchical Fortran 90 code
- State preservation via global variables
- Highly configurable (ifdef) code

Need greater tool support
- Control flow graph: dependency analysis
- Data flow graph: activity analysis
- Automated active variable type-changing
Thank you