Tools for Advanced Model Development

Andrew Lee[a], Jaffer Ghouse[a], Carl Laird[b], John Siirola[b]
[a] National Energy Technology Laboratory, [b] Sandia National Laboratories

Unit Models

- Commercial process simulation tools have libraries of models for common unit operations. However, libraries offer limited flexibility and lack models for novel processes.
- AMLs offer the ability for users to develop custom models for process systems, but require significant effort to develop , debug and solve.
- IDAES aims to bridge the gap between process simulators and AMLs by offering a flexible, modular framework for constructing models of unit operations.

Property Blocks

- Properties implemented as modular blocks
- Interactions defined at a high level to allow flexible formulation of equations
- Easily switched in and out to allow unit models to be applied to different processes
- Framework automates many interactions

Ports

- Describe points where models can be connected
- Port members are defined by the user via the property package.
- User can pass any state information required
- Can connect to any other Port with the same members

Contact

Andrew Lee, andrew.lee@netl.doe.gov, 412-386-7103

Disclaimer/Acknowledgement : This presentation was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof. KeyLogic Systems, Inc.'s contributions to this work were funded by the National Energy Technology Laboratory under the Mission Execution and Strategic Analysis contract (DE-FE0025912) for support services. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia LLC, a wholly owned subsidiary of Honeywell International Inc. for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Example

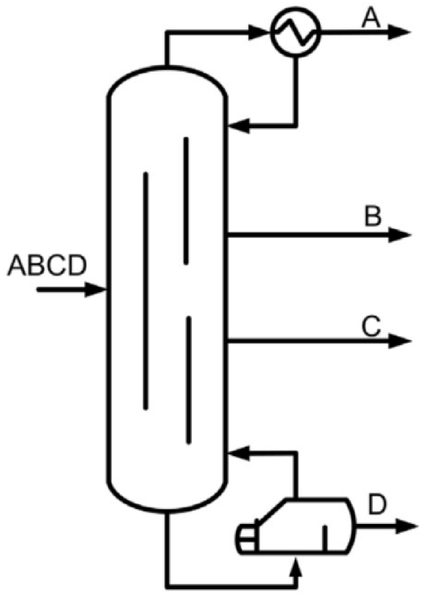
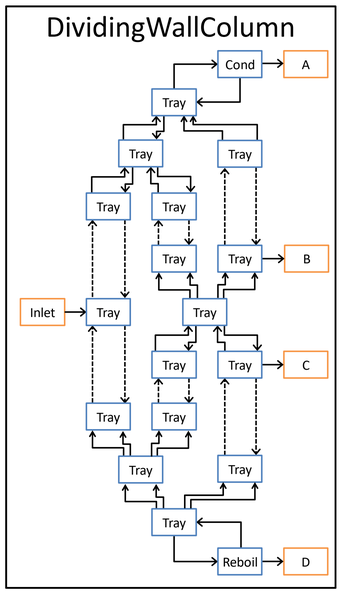
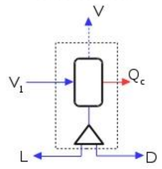


Image from Okoli and Adams, Chem. Eng. and Proc., 2015



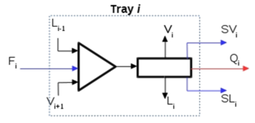
Source: NETL

Condenser Block

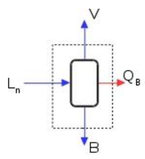


Tray Block

Tray *i*



Reboiler Block



Control Volumes

- Three types of Control Volumes available for different applications
  - 0-D (Inlet-Outlet type)
  - 1-D (Pipes, PFRs)
  - Static (Dead zones)
- Control Volumes can be connected together to form complex units with different levels of detail
- Represent defined volume over which material, energy and momentum balances will be written
- Prebuilt methods for common forms of balance equations, e.g.

$$Acc = In - Out + Gen + Trans$$

IDAES Framework

- Library of models for common unit operations
- All models are fully open and modifiable
- Models are constructed from modular elements for common model features
- Elements provide block-modular structure of process simulators, with the flexibility of AMLs
- Framework provides tools to automate common task, such as writing material, energy and momentum balances to reduce effort required to develop models and reduce errors
- Open source allows for community to contribute new features
- Implementation in Python gives access to wide range of external libraries (e.g. NumPy, SciPy)
- Users are free to use as many or as few of the framework tools as they desire

Initialization

- Each unit model can define a custom initialization routine
- Users can develop custom initialization routines making use of a wide range of tools and solvers
- Routines consider entire unit operation, not just sub-components
- IDAES is developing for solving complex models
  - Decomposition solvers
  - Reliability and convergence tools

Sample output from model reliability and convergence tester

```
  0.0% [----------------] Root Process: Sample-1
 …
 96.0% [***************-] Root Process: Sample-25

==== Scenario Statistics ====
          Parameter        Min        Mean      StdDev        Max
-------------------------------------------------------------------
     Inlet_Flowrate          1      477127      169716      877237
  Inlet_Temperature    202.486     604.057     192.441        1000
-------------------------------------------------------------------

==== Summary ====
Number of Successful Cases (solved=True): 97/100
... Iterations      (min, -1std, mean, +1std, max):     4, 6.136298, 6.835052, 7.533805,     9
... Solver Time (s) (min, -1std, mean, +1std, max): 0.002000, 0.003171, 0.003990, 0.004808, 0.008000

==== Notable Cases ====
Sample-57 : failed solve
            Inlet_Flowrate:     512046 ( 0.06 standard deviations above/below the mean)
         Inlet_Temperature:       1000 ( 2.00 standard deviations above/below the mean)
```