Making Optimization-Based Process Modeling Easier

Andrew Lee, John Siirola

SAND2024-12393PE

The Problem

The biggest bottleneck for Equation Oriented modeling is diagnosing solver failures

- It is easy to write models that are hard to solve

The IDAES Team is working to try to make this easier

- We welcome input from users on pain points

```
This is Ipopt version 3.14.11, running with linear solver ma27.

Number of nonzeros in equality constraint Jacobian...:     6052
Number of nonzeros in inequality constraint Jacobian.:        0
Number of nonzeros in Lagrangian Hessian.............:     2666

Total number of variables............................:     1760
                     variables with only lower bounds:      102
                variables with lower and upper bounds:      736
                     variables with only upper bounds:        0
Total number of equality constraints.................:     1551
Total number of inequality constraints...............:        0

iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alpha_pr  ls
   0  0.0000000e+00 2.52e+02 1.00e+00  -1.0 0.00e+00    -  0.00e+00 0.00e+00   0
   1  8.5659089e+03 9.29e+01 6.67e+02  -1.0 1.71e+03    -  7.27e-02 9.92e-01h  1
   2  1.0459436e+04 5.27e-01 3.80e+11  -1.0 6.24e+03    -  4.36e-02 1.00e+00h  1
   3  1.0433252e+04 5.05e-01 3.64e+11  -1.0 4.41e+02  -4.0 6.36e-02 4.11e-02f  1
   4  1.0371288e+04 4.30e-01 3.10e+11  -1.0 9.66e+02  -3.6 5.61e-02 1.48e-01f  1
   5  1.0371288e+04 9.21e+01 3.35e+11  -1.0 1.80e-05  16.3 9.90e-01 9.90e-01s 22
MA27BD returned iflag=-4 and requires more memory.
 Increase liw from 100065 to 200130 and la from 108695 to 223566 and factorize again.
MA27BD returned iflag=-4 and requires more memory.
 Increase liw from 200130 to 400260 and la from 223566 to 450480 and factorize again.
   6r 1.0371288e+04 9.21e+01 9.92e+02  -1.0 0.00e+00  15.8 0.00e+00 0.00e+00R  1
[...]
 319r 2.0874480e-01 4.12e-05 9.09e-13  -7.1 3.17e+02    -  1.00e+00 1.00e+00h  1
Restoration phase converged to a point with small primal infeasibility.

Number of Iterations....: 319

[...]

Total seconds in IPOPT                               = 2.186

EXIT: Restoration Failed!
```

The Problem

- Process models suffer from a number of challenges
  - Process models are large, but often involve many similar constraints
  - Thermophysical properties are highly non-linear
  - Recycle streams add degeneracies
  - Capturing / exploiting process structure complicates nonlinear system
  - Process engineers are generally not mathematicians

- Modular, hierarchical model libraries add a layer of abstraction
  - Library models are not scaled for user's case studies
  - Modular construction introduces extra variables and constraints

New Features in 2024

1. Improved Solver Interfaces

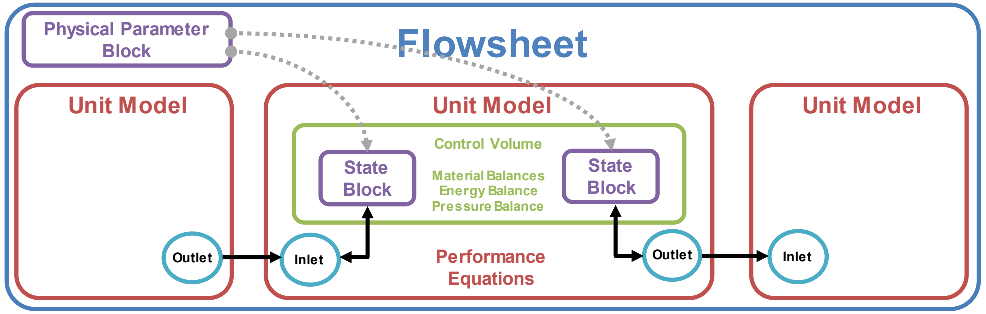
    1. Linear Presolve for Non-Linear Models
    2. Scaling within the Solver Writer

2. New Scaling Toolbox and Workflow

3. Expanded Diagnostics Toolset

Improved Solver Interfaces

- IDAES models go through several steps on their way to the solver
  - Constructing the initial (block hierarchical) process model
  - Transforming the model to a (MI)(N)LP
    - Reformulating disjuncts, discretizing DAE systems, expanding arcs
  - Compiling the model to an intermediate representation
    - Identifying constraints, variables
    - Separating expressions into linear and nonlinear components
    - Sorting variables and constraints (by domain, use in nonlinear expressions)
  - "Writing out" the compiled representation
    - Generating the "NL file"
  - Invoking the solver



We have spent the last 18 months focused on redesigning the "Solver Writers"

- Improve overall performance
- Generate more efficient representations
  - Exploit "defined variables"
- Implement basic presolve
- Implement scaling on the compiled representation

Linear Presolve

```
This is Ipopt version 3.14.11, running with linear solver ma27.

Number of nonzeros in equality constraint Jacobian...:     6052
Number of nonzeros in inequality constraint Jacobian.:        0
Number of nonzeros in Lagrangian Hessian.............:     2666

Total number of variables............................:     1760
                     variables with only lower bounds:      102
                variables with lower and upper bounds:      736
                     variables with only upper bounds:        0
Total number of equality constraints.................:     1551
Total number of inequality constraints...............:        0

iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alpha_pr  ls
   0  0.0000000e+00 2.52e+02 1.00e+00  -1.0 0.00e+00    -  0.00e+00 0.00e+00   0
   1  8.5659089e+03 9.29e+01 6.67e+02  -1.0 1.71e+03    -  7.27e-02 9.92e-01h  1
   2  1.0459436e+04 5.27e-01 3.80e+11  -1.0 6.24e+03    -  4.36e-02 1.00e+00h  1
   3  1.0433252e+04 5.05e-01 3.64e+11  -1.0 4.41e+02  -4.0 6.36e-02 4.11e-02f  1
   4  1.0371288e+04 4.30e-01 3.10e+11  -1.0 9.66e+02  -3.6 5.61e-02 1.48e-01f  1
   5  1.0371288e+04 9.21e+01 3.35e+11  -1.0 1.80e-05  16.3 9.90e-01 9.90e-01s 22
MA27BD returned iflag=-4 and requires more memory.
 Increase liw from 100065 to 200130 and la from 108695 to 223566 and factorize again.
MA27BD returned iflag=-4 and requires more memory.
 Increase liw from 200130 to 400260 and la from 223566 to 450480 and factorize again.
   6r 1.0371288e+04 9.21e+01 9.92e+02  -1.0 0.00e+00  15.8 0.00e+00 0.00e+00R  1
[...]
 319r 2.0874480e-01 4.12e-05 9.09e-13  -7.1 3.17e+02    -  1.00e+00 1.00e+00h  1
Restoration phase converged to a point with small primal infeasibility.

Number of Iterations....: 319

[...]

Total seconds in IPOPT                               = 2.186

EXIT: Restoration Failed!
```

- Linear models get significant benefits from extensive presolve in the solver
- (Most) nonlinear solvers do not have similar steps
  - Rely on the modeling environment to perform any presolve
  - Part of the redesign of the “NL Writer” enables operations on the compiled model (e.g., presolve)
- First step: implementing “variable aggregation with zero fill-in”
  - Identify (and remove) implicitly fixed variables
  - Identify (and remove) any bivariate linear constraints (e.g. $ax + by = c$)
    - (recursively)

Presolve (significantly) helps (some) models

Without Presolve

```
This is Ipopt version 3.14.11, running with linear solver ma27.

Number of nonzeros in equality constraint Jacobian...:     6052
Number of nonzeros in inequality constraint Jacobian.:        0
Number of nonzeros in Lagrangian Hessian.............:     2666

Total number of variables............................:     1760
                     variables with only lower bounds:      102
                variables with lower and upper bounds:      736
                     variables with only upper bounds:        0
Total number of equality constraints.................:     1551
Total number of inequality constraints...............:        0

iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alpha_pr  ls
   0  0.0000000e+00 2.52e+02 1.00e+00  -1.0 0.00e+00    -  0.00e+00 0.00e+00   0
   1  8.5659089e+03 9.29e+01 6.67e+02  -1.0 1.71e+03    -  7.27e-02 9.92e-01h  1
   2  1.0459436e+04 5.27e-01 3.80e+11  -1.0 6.24e+03    -  4.36e-02 1.00e+00h  1
   3  1.0433252e+04 5.05e-01 3.64e+11  -1.0 4.41e+02  -4.0 6.36e-02 4.11e-02f  1
   4  1.0371288e+04 4.30e-01 3.10e+11  -1.0 9.66e+02  -3.6 5.61e-02 1.48e-01f  1
   5  1.0371288e+04 9.21e+01 3.35e+11  -1.0 1.80e-05  16.3 9.90e-01 9.90e-01s 22
MA27BD returned iflag=-4 and requires more memory.
 Increase liw from 100065 to 200130 and la from 108695 to 223566 and factorize again.
MA27BD returned iflag=-4 and requires more memory.
 Increase liw from 200130 to 400260 and la from 223566 to 450480 and factorize again.
   6r 1.0371288e+04 9.21e+01 9.92e+02  -1.0 0.00e+00  15.8 0.00e+00 0.00e+00R  1
[...]
 319r 2.0874480e-01 4.12e-05 9.09e-13  -7.1 3.17e+02    -  1.00e+00 1.00e+00h  1
Restoration phase converged to a point with small primal infeasibility.

Number of Iterations....: 319

[...]

Total seconds in IPOPT                               = 2.186

EXIT: Restoration Failed!
```

Removed 227 variables and constraints

With Presolve

```
This is Ipopt version 3.14.11, running with linear solver ma27.

Number of nonzeros in equality constraint Jacobian...:     5499
Number of nonzeros in inequality constraint Jacobian.:        0
Number of nonzeros in Lagrangian Hessian.............:     2660

      of variables............................:     1533
                     variables with only lower bounds:        0
                variables with lower and upper bounds:      721
                     variables with only upper bounds:        0
      of equality constraints.................:     1324
Total number of inequality constraints...............:        0

iter    objective    inf_pr   inf_du lg(mu)  ||d||  lg(rg) alpha_du alpha_pr  ls
   0  0.0000000e+00 2.52e+02 0.00e+00  -1.0 0.00e+00    -  0.00e+00 0.00e+00   0
   1  2.9971033e+00 2.47e+02 6.22e+01  -1.0 5.73e+04    -  1.70e-02 1.93e-02h  1
   2  3.8475690e+00 2.46e+02 7.82e+01  -1.0 1.67e+04    -  5.33e-02 2.33e-03h  1
   3  1.7777952e+01 2.38e+02 8.32e+02  -1.0 1.13e+05    -  2.96e-04 3.34e-02h  1
   4  4.5705165e+02 1.94e+02 8.98e+02  -1.0 2.48e+04    -  4.36e-02 1.83e-01h  1
   5  4.7435196e+02 1.91e+02 1.23e+03  -1.0 1.92e+02   0.0 5.21e-04 1.56e-02h  1
   6  5.3332345e+02 1.81e+02 4.33e+05  -1.0 1.80e+02   1.3 3.50e-05 5.15e-02h  1
   7  7.0259915e+02 1.56e+02 4.98e+05  -1.0 1.75e+02   0.9 5.60e-03 1.38e-01h  1
   8  7.0523497e+02 1.56e+02 4.97e+05  -1.0 1.13e+02   1.3 8.64e-03 2.70e-03h  1
   9r 7.0523497e+02 1.56e+02 9.99e+02   2.2 0.00e+00   0.8 0.00e+00 2.98e-07R  6
  10r 4.0390538e+03 6.27e+01 9.97e+02   2.2 5.42e+06    -  2.72e-02 8.08e-04f  1
  11  3.8750164e+03 6.13e+01 3.18e+01  -1.0 3.13e+04    -  1.88e-02 2.27e-02f  1
[...]
  90  8.5411094e-02 1.19e-07 5.93e-16  -9.0 5.43e-01    -  1.00e+00 1.00e+00h  1

Number of Iterations....: 90

[...]

Total seconds in IPOPT                               = 0.163

EXIT: Optimal Solution Found.
```

Scaling in the Writer

- IDAES models rely heavily on scaling to improve solver performance
  - Currently rely on features specific to Ipopt (suffixes)
    - Or a “scaling transformation” that rewrites the entire model
  - New development: scaling implemented within the NL writer
    - Relies on the same machinery as the presolver
    - Enables a unified *solver agnostic* approach to implementing scaling

  - Current work: expanding this to other solver interfaces (Baron, GAMS, LP)

Model Scaling Tools

- Model scaling is an ongoing pain point
  - Many current projects involve very large or small values
  - E.g. trace concentrations, electricity grid power flows
- Existing scaling tools have some issues
  - Tailored specifically for IPOPT
  - Lack of transparency
  - Lack of flexibility
  - Tend to emit lots of warnings
  - This meant they were not being used
- Introducing a new Scaling Toolbox to make scaling easier

What is Good Scaling?

- “Good” scaling is hard to define

- Scaling actually consists of three parts
  - Variable scaling
  - Constraint residual scaling
  - Jacobian scaling

- Different aspects of scaling are not always complementary
  - Need to balance each aspect

What is Good Scaling?

- Poor scaling is easier to define
  - Variables with very large or small magnitudes
  - Constraints with terms of mismatched magnitude or cancellation
  - Entries in the Jacobian with extreme values

- Scaling is more about avoiding bad scaling than finding "good" scaling
  - "Near-enough" is better than "perfect"

Signs of Poor Scaling: Numerical Diagnostics

```
====================================================================================
Model Statistics

    Jacobian Condition Number: 1.540E+12

------------------------------------------------------------------------------------
1 WARNINGS

    WARNING: 1 Constraint with large residuals (>1.0E-05)

------------------------------------------------------------------------------------
6 Cautions

    Caution: 1 Variable with value close to their bounds (abs=1.0E-04, rel=1.0E-04)
    Caution: 4 Variables with value close to zero (tol=1.0E-08)
    Caution: 4 Variables with extreme value (<1.0E-04 or >1.0E+04)
    Caution: 3 Variables with extreme Jacobian values (<1.0E-04 or >1.0E+04)
    Caution: 2 Constraints with extreme Jacobian values (<1.0E-04 or >1.0E+04)
    Caution: 6 extreme Jacobian Entries (<1.0E-04 or >1.0E+04)

------------------------------------------------------------------------------------
Suggested next steps:

    display_constraints_with_large_residuals()
    compute_infeasibility_explanation()

====================================================================================
```

Approaches to Scaling

- Full User Scaling
  - User scales all variables and constraints themselves
  - Assisted by methods for common techniques
  - Labor intensive

- Automatic Scaling based on model state
  - Fully automated so requires minimal user input
  - Only considers current state, so often over-tunes scaling
  - Best used as an initial guess

- Model Specific Routines
  - Heuristics defined by model developer
  - Can leverage understanding of model structure and behavior
  - Depend heavily on skill of developer

New Tools In Development

- Utility tools
  - Tools for manipulating scaling factors (set, get, delete, etc.)
- AutoScaler class
  - Scale variable by current magnitude
  - Scale constraints by Jacobian norm
- CustomScalerBase class
  - Methods for common ways to scale variable and constraints

Variables by…	Constraints by…
Nominal magnitude	Harmonic mean of terms
Bounds	Inverse mean of terms
Units of measurement	Inverse root-mean-squares of terms
Pre-defined default	Inverse of maximum term
Other component	Inverse of minimum term
	Inverse of estimated norm

IDAES Institute for the Design of Advanced Energy Systems

Example of New Tools

```python
from idaes.core.scaling import AutoScaler, set_scaling_factor
from idaes.core.solvers import get_solver

# Build model
model = build_model()

# Add user scaling factors
set_scaling_factor(model.v, 1e-3)

# Apply scaling using new AutoScaler toolbox
# AutoScaler requires an initialized model
# See CustomScalerBase for tools to use on un-initialized models
scaler = AutoScaler()
scaler.variables_by_magnitude(model)
scaler.constraints_by_jacobian_norm(model)

# Solve scaled model directly with new Pyomo solver interface
solver = get_solver(solver="ipopt_v2", writer_config={"scale_model": True})
results = solver.solve(model)
```

IDAES® Institute for the Design of Advanced Energy Systems

Profiling Scaling Techniques

- Automated method for comparing scaling techniques
  - Condition Number is an indicator of the "worst case" gain for error
  - Test Case: Gibbs Reactor demo from IDAES

Constraint Scaling Routine	Best-Guess Variable Scaling		Perfect Variable Scaling	
	Condition No.	Perturbation	Condition No.	Perturbation
Unscaled	5.703e17	Failed, 57 iter		
Variable Only	9.245e16	Failed, 82 iter	6.577e14	Solved, 9 iter
Harmonic Mean	1.302e17	Failed, 99 iter	2.837e12	Solved, 17 iter
Inverse Sum	9.545e18	Solved, 30 iter	8.769e10	Solved, 3 iter
Inv. Root Sum Sqs.	2.863e19	Solved, 30 iter	1.301e11	Solved, 3 iter
Inv. Max. Mag.	8.590e19	Solved, 30 iter	1.774e11	Solved, 3 iter
Inv. Min. Mag.	1.268e17	Failed, 87 iter	5.599e12	Solved, 16 iter
Nominal L1 Norm	1.189e16	Failed, 61 iter	2.060e6	Solved, 4 iter
Nominal L2 Norm	1.188e16	Failed, 53 iter	3.074e6	Solved, 4 iter
AutoScaler L1 Norm	1.461e9	Failed, 29 iter	2.978e3	Solved, 6 iter
AutoScaler L2 Norm	6.613e8	Failed, 29 iter	2.511e3	Solved, 6 iter

New Diagnostics Capabilities

- More checks for more issues
  - Potential evaluation errors (singularities)
  - Infeasibility explainer (infeasibility)
  - Near parallel variables and constraints (degeneracies, ill-conditioning)
  - Poorly posed constraints (scaling)

- All core models are now tested for diagnostics issues

New Diagnostics Capabilities

- Potential evaluation errors (singularities)
  - $\log x$ or $\sqrt{x}$ where bounds allow $x \leq 0$
  - $\frac{1}{x}$ where bounds allow for $x = 0$
  - x^y where bounds allow $x \leq 0$ and $y \notin \mathbb{Z}$
- Part of numerical checks

New Diagnostics Capabilities

- Near parallel variables and constraints (degeneracies, ill-conditioning)
  - Identify potential duplicate variables and constraints

- Part of numerical checks

Duplicated Constraint	Numerically Parallel Constraints	Parallel Variables
$A + B = C$	$A + B = C$	$A + B = C$
$D + B = E$	$A + B = C + D$	$D = 5(A + B)$
$A + B = C$	$D = \varepsilon$	$\log(A + B) = C$

New Diagnostics Capabilities

- Poorly posed constraints (scaling)
  - Constraints with poorly matched terms

$$0 = A + B + \varepsilon$$
$$0 = e^{(A+B+\varepsilon)}$$

  - Constraints with potential catastrophic cancelation

$$A + B = C$$
$$e^{A+B} = C$$
$$A + B + C = D$$

where $A + B \cong 0$

- Part of numerical checks

IDAES®
Institute for the Design of Advanced Energy Systems

New Diagnostics Capabilities

- Infeasibility explainer
  - What relaxations would allow the model to be feasible?

- Find the minimum set(s) of relaxations required
  - Variable bounds
  - Constraint slacks

- Numerical report suggests this if constraint violations are detected

Summary

- IDAES Team is working to address common pain points

- This year:
  - Improvements solver writers in Pyomo
  - New IDAES Scaling Toolbox
  - More diagnostics checks

- Welcome suggestions on next pain points to address

QUESTIONS AND COMMENTS?

Acknowledgement and Disclaimer

Acknowledgement: This work was conducted as part of the U.S. Department of Energy's Institute for the Design of Advanced Energy Systems (IDAES) supported by the Office of Fossil Energy and Carbon Management's Simulation-based Engineering/Crosscutting Research Program.