Simulation & Optimization of Pressure Swing Adsorption

Yajun Wang & Alexander W. Dowling
Lorenz T. Biegler
Department of Chemical Engineering
Carnegie Mellon University
March 2014
Motivation

• Pressure Swing Adsorption (PSA)
  -- Gas separation (e.g. H₂ – CO₂)
  -- In IGCC power plant

• Objective
  -- Modify PSA simulation
  -- Diagnose convergence problems in PSA optimization

State variables: $C_{CO_2}$, $C_{H_2}$, $q_{CO_2}$, $q_{H_2}$, $T$

5 operating steps (time slots):

- $\alpha$
- $\beta$
- $\phi$
- $P_{ads}$
- $P_{des}$

Additional control variables: $t_s$, $L_{bed}$

Cyclic steady state

$$z_{C_{OB} \left( x, 0 \right)} = z_{C_{nB} \left( x, t_f \right)}$$

$$z_{C_{nB} \left( x, 0 \right)} = z_{C_{OB} \left( x, t_f \right)}$$

PSA modeling\textsuperscript{[4]}

- **Mass balance**
  \[ \epsilon_b \frac{\partial c_i}{\partial t} + (1 - \epsilon_b) \rho_s \frac{\partial q_i}{\partial t} + \frac{\partial (v c_i)}{\partial x} = 0 \]

- **Mass transfer**
  \[ \frac{\partial q_i}{\partial t} = k_i (q_i^* - q_i) \]

- **Energy balance**
  \[ \left( \epsilon_t \sum_i c_i \left( C_p^i - R \right) + \rho_s C_{ps} \right) \frac{\partial T}{\partial t} - \rho_s \sum_i \Delta H_i^{ads} \frac{\partial q_i}{\partial t} + \frac{\partial (\nu h)}{\partial x} + U_A (T - T_w) = 0 \]

- **Momentum balance**

- **Adsorption Isotherm**

- **Ideal gas equation**

- **Check valve equation, Bed connection equations, Auxiliary equations**

\textsuperscript{[4]} Ling Jiang, Lorenz T Biegler, and V Grant Fox. Simulation and optimization of pressure-swing adsorption systems for air separation.
PSA Simulation

- System discretization
  - Convert PDEs to ODEs
  - Discretize space domain\[^1\]
  - Finite volume method

\[ \nabla \cdot f(z) = \frac{1}{\Delta x_i} (f(z_{j+\frac{1}{2}}) - f(z_{j-\frac{1}{2}})) \]

- Flux limiter
  - Keep sharpness of steep fronts
  - Prevent oscillations

\[ z \in \{C_i, q_i, T\} \]
Two types of flux limiters

Van Leer\textsuperscript{[4]}

\[ r = \frac{\bar{z}_j - \bar{z}_{j-1}}{\bar{z}_{j+1} - \bar{z}_j} \quad \text{if} \quad v_{j+\frac{1}{2}} \geq 0 \]

\[ r = \frac{\bar{z}_{j+1} - \bar{z}_{j+2}}{\bar{z}_j - \bar{z}_{j+1}} \quad \text{if} \quad v_{j+\frac{1}{2}} < 0 \]

\[ r' = 0.5(r^2 + \epsilon^2)^{\frac{1}{2}} + 0.5r \approx \max(0, r) \]

\[ \Phi(r') = \frac{2r'}{1 + r'} \]

\[ \bar{z}_{j+\frac{1}{2}} = \bar{z}_j + \frac{\bar{z}_{j+1} - \bar{z}_j}{2} \Phi(r') \quad \text{if} \quad v_j \geq 0 \]

\[ \bar{z}_{j+\frac{1}{2}} = \bar{z}_{j+1} + \frac{\bar{z}_j - \bar{z}_{j+1}}{2} \Phi(r') \quad \text{if} \quad v_j < 0 \]

Non-differentiable
Two types of flux limiters

WENO\textsuperscript{[3]}

\[
\begin{align*}
    z_{j+\frac{1}{2}} &= \frac{\alpha_{0,j}}{\alpha_{0,j} + \alpha_{1,j}} \left[ \frac{1}{2} (z_j + z_{j+1}) \right] + \frac{\alpha_{1,j}}{\alpha_{0,j} + \alpha_{1,j}} \left[ \frac{3}{2} z_j - \frac{1}{2} z_{j-1} \right] \\
    \alpha_{0,j} &= \frac{2/3}{(z_{j+1} - z_j + \delta)^4} \\
    \alpha_{1,j} &= \frac{1/3}{(z_j - z_{j-1} + \delta)^4}
\end{align*}
\]

Differentiable

WENO performs similarly as van Leer to maintain accurate fronts.

Without flux limiter, the simulation is poor near fronts.
WENO is computationally more efficient

Time to reach cyclic steady state

<table>
<thead>
<tr>
<th>Flux Limiter Type</th>
<th>No flux limiter</th>
<th>van Leer</th>
<th>WENO</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Time(s) Iter</td>
<td>Time(s) Iter</td>
<td>Time(s) Iter</td>
</tr>
<tr>
<td>2 comps 11 grids</td>
<td>205.9 236</td>
<td>264.2 232</td>
<td>227.0 234</td>
</tr>
<tr>
<td>2 comps 21 grids</td>
<td>1064.4 202</td>
<td>1772.7 232</td>
<td>1537.4 234</td>
</tr>
<tr>
<td>2 comps 51 grids</td>
<td>10754.3 246</td>
<td>15915.3 236</td>
<td>14022.2 236</td>
</tr>
<tr>
<td>5 comps 11 grids</td>
<td>1125.1 234</td>
<td>1749.0 234</td>
<td>1344.9 234</td>
</tr>
<tr>
<td>5 comps 21 grids</td>
<td>5116.9 212</td>
<td>10124.5 246</td>
<td>7764.7 248</td>
</tr>
<tr>
<td>5 comps 51 grids</td>
<td>64571.8 260</td>
<td>115865.0 244</td>
<td>95196.2 246</td>
</tr>
</tbody>
</table>

WENO Reduce 17.84% computational time
Optimization formulation

\[ \min \Phi(z(x, t_f), p) \]

s.t. \[ F\left(\frac{\partial(z)}{\partial(t)}, \frac{\partial(z)}{\partial(x)}, z(x, t), z(x, 0), p\right) = 0 \]

\[ g(z(x, t), p) \geq 0 \]

\[ 0 \leq (\alpha(t), \beta(t), \phi(t)) \leq 1 \]

\[ b_L \leq (P_{ads}(t), P_{des}(t), t_s, L_{bed}) \leq b_U \]

\[ p = [\alpha(t), \beta(t), P_{ads}(t), P_{des}(t), t_s, L_{bed}] \]
Periodic Boundary Condition Formulation\[1\]

- Add cyclic steady state equations

\[ z_{CoB}(x, 0) = z_{CnB}(x, t_f) \]
\[ z_{CnB}(x, 0) = z_{CoB}(x, t_f) \]

Step 1

Switch Beds and Repeat

- Optimal variables: initial states and control variables

\[ C_{CO_2} \quad C_{H_2} \quad q_{CO_2} \quad q_{H_2} \quad T \]

\[ \alpha \quad \beta \quad \phi \quad P_{ads} \quad P_{des} \quad t_s \quad L_{bed} \]
Convergence troubles[1]

• Sensitivity calculation
  -- Expensive (more than 95% computational time)
  -- Approximate within integrator tolerances, $O(10^{-6} \text{ to } 10^{-10})$

• Hard to converge
  -- Large number of iterations, computational time
  -- Infeasibility

• Sensitive to initial guess

MATLAB fmincon
  \[\begin{align*}
  &\text{SQP} \\
  &\text{Active-set} \\
  &\text{Interior-point}
  \end{align*}\]

*First derivatives from direct sensitivity
Optimization Diagnostics

Optimization ends up with no feasible solution

<table>
<thead>
<tr>
<th>Final iteration</th>
<th>92</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objective value</td>
<td>85.08</td>
</tr>
<tr>
<td>Constraints violation</td>
<td>0.989</td>
</tr>
<tr>
<td>KKT violation</td>
<td>29.69</td>
</tr>
</tbody>
</table>

Large KKT violation is caused by $\phi$

Check approximate Hessian matrix at final point, large elements also correspond to $\phi$
Importance of scaling

Scale $\phi$

Optimization data at iteration 80

<table>
<thead>
<tr>
<th></th>
<th>Not scaled</th>
<th>Scaled by 100</th>
<th>Scaled by 1000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Objective value</td>
<td>86.24</td>
<td>86.26</td>
<td>85.8</td>
</tr>
<tr>
<td>Constraints violation</td>
<td>0.289</td>
<td>$9.78 \times 10^{-4}$</td>
<td>$2.32 \times 10^{-3}$</td>
</tr>
<tr>
<td>KKT violation</td>
<td>29.42</td>
<td>3.338</td>
<td>8.598</td>
</tr>
</tbody>
</table>

With scaled $\phi$, both primal and dual feasibility can be improved largely.
Reduce problem dimension

- Fix control variables
  optimization problem \(\rightarrow\) nonlinear equations solving problem
  \# bed state variables = \# bed equations

- Optimization with only one type of control variables
  \(\alpha\ \ \beta\ \ \phi\ \ P_{\text{ads}}\ \ P_{\text{des}}\ \ t_s\ \ L_{\text{bed}}\)

- Redo optimization with all control variables except \(\phi\)

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Final iteration</td>
<td>232</td>
</tr>
<tr>
<td>Objective value</td>
<td>84.71</td>
</tr>
<tr>
<td>Constraints violation</td>
<td>7.68\times10^{-7}</td>
</tr>
<tr>
<td>KKT violation</td>
<td>4.39\times10^{-3}</td>
</tr>
</tbody>
</table>
Conclusion & Future work

✓ WENO flux limiter is more suitable for PSA optimization – high efficiency and better differentiability
✓ Scaling improves optimizer performance
✓ Simplified problems converge to optimum
  o Continue optimization diagnostics
  o Consider optimization with reduced order models
Acknowledgement

• Professor Larry Biegler
• Alexander W. Dowling
• The Department of Chemical Engineering at Carnegie Mellon University
• National Energy Technology Laboratory (NETL) (initial funding)