Simulation and Optimal Design of Multiple-Bed Pressure Swing Adsorption Systems

Ling Jiang, V. Grant Fox, and Lorenz T. Biegler
Dept. of Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA 15123

DOI 10.1002/aic.10223
Published online in Wiley InterScience (www.interscience.wiley.com).

Pressure swing adsorption (PSA) is a very versatile technology for gas separation and purification. The widespread industrial application of PSA has called for an efficient set of simulation, design, and optimization methodologies. In previous work by Jiang and co-workers, we used a Newton-based approach to quickly converge the cyclic steady state and design constraints, and a simultaneous tailored approach with the state-of-art nonlinear optimization strategy to design optimal PSA processes. In this work we extend the simulation and optimization strategies to multiple bed systems. Both unibed and multibed frameworks are adopted to describe bed behaviors. The unibed framework models only one bed over a cycle and uses storage buffers to mimic the bed interactions. The multibed framework simultaneously solves all beds but only for a portion of the cycle. Challenges and implementation details of both frameworks are discussed. A five-bed, 11-step hydrocarbon separation process, which separates H_2 from a mixture of H_2, N_2, CO_2, CO, and CH_4, is used for illustration. By manipulating valve constants, step times, flow rates, and bed geometry, the optimizer successfully maximizes H_2 recovery, while meeting product purity and pressure specifications. © 2004 American Institute of Chemical Engineers *AIChE J,* 50: 2904–2917, 2004

Keywords: adsorption/gas, optimization, computer simulations (MC and MD), simulation, multiscale, design (process simulation)

Introduction

Over the past few decades, there has been widespread development of pressure swing adsorption (PSA) systems (for a general overview, see Ruthven et al., 1994). Their applications have expanded from bulk gas separations and drying to trace contaminant removal. With extensive industrial applications, there is significant interest for an efficient modeling, simulation, and optimization strategy. However, despite such growth in the practical application of PSA, the design and optimization of a PSA system still largely remains an experimental effort for several reasons. First, most practical PSA processes are complex systems described by coupled hyperbolic nonlinear partial differential equations (PDEs) in time and space domains with different initial and boundary conditions defining the steps of the process. The computational effort required to solve such systems is usually quite expensive and time-consuming. Second, high nonlinearities and ill-conditioning that often arise from multiple adsorbent layers and nonisothermal effects, as well as stringent product specifications (such as high purity) required by many industrial processes, often lead to the failure of numerical solvers. Therefore, a robust and efficient design framework using rigorous process models is the main objective of our work.

In Jiang et al. (2003), we developed a series of rigorous numerical procedures to design optimal PSA processes with single-bed models. A concise overview of this work will be given in the next section and an extensive review on the simulation and optimization of PSA processes is given in Biegler et al. (2004b). In this work we extend our simulation and optimization strategies from single-bed systems to multiple-bed systems. Multibed PSA systems are more involved than one or two bed operations because of the complex

Correspondence concerning this article should be addressed to L. T. Biegler at 1b01@andrew.cmu.edu.

interconnectivity of the beds. The third section introduces a five-bed hydrogen purification process used as a case study in this report. The implementations of unibed and multibed frameworks are discussed in the fourth section, followed by the results in the fifth section and conclusions in the sixth section.

Solution Procedures Overview

This section gives a quick overview of the numerical tools we developed and applied throughout this work. Refer to Jiang et al. (2003) for details.

PDE discretization

The mathematical models for PSA systems are the coupled mass, momentum, and heat conservation laws as well as adsorption equilibrium and thermodynamics. The mass and energy balances are represented by hyperbolic PDEs, which cause sharp adsorption fronts. To ensure the accurate approximation of the adsorption models, we apply the conservative finite-volume method to discretize PDEs in space and reduce PDEs to differential algebraic equations (DAEs). The modified second-order Van Leer flux limiter is used to mitigate numerical error and avoid physically unrealistic smearing and oscillation near the adsorption fronts.

DAE solver and sensitivity evaluation

DASPK 3.0 (Li and Petzold, 1999) is used to integrate the DAEs over time. DASPK solves initial value problems of DAEs using a combination of backward diffentiation formulae (BDFs) and a choice of linear system solution methods. For a general DAE system

$$F(t, y, y', q) = 0 \tag{1}$$

DASPK solves this equation by a modified version of Newton's method. DASPK performs more efficiently if the DAE Jacobian from Eq. 1 is banded. In this case, the storage needed is greatly reduced, numerical differencing is much cheaper, and a number of important algorithms execute much faster. The differential equation is said to have half-bandwidths ML (lower) and MU (upper) if row i in the Jacobian involves variables only in column j with $i\text{-}ML \le j \le i + MU$ for $i = 1, 2, \ldots, NEQ$. With the finite-volume discretization and Van Leer flux limiter, described in Jiang et al. (2003), the DAE system has a bandwidth ML = MU = 3*nodes, which means each spatial node is associated with itself and only two upstream and two downstream nodes. In this study, to reduce the bandwidth further for each node, we formally eliminate the algebraic equations and variables and rewrite the DAE system as the corresponding ordinary differential equation (ODE) system.

DASPK also simultaneously performs sensitivity analysis on the parameters contained within the DAE system. Sensitivity analysis entails finding the derivative of each variable with respect to each parameter. DASPK evaluates sensitivities by differentiating the original DAE system, yielding additional $N_q \times N_y$ sensitivity equations, as follows

$$F(t, y, y', q) = 0$$

$$\frac{\partial F}{\partial y} s_i + \frac{\partial F}{\partial y'} s_i' + \frac{\partial F}{\partial q_i} = 0 \qquad i = 1, \ldots, N_q$$

where

$$s_i = \frac{\partial y}{\partial_{qi}}$$

Accurate sensitivities $s_i(t)$ are essential for design and optimization. This *direct sensitivity approach* solves the sensitivity equations together with the original DAEs; it is straightforward and easy to implement. However, the computational cost increases linearly with the number of parameters N_q and can be quite time-consuming when N_q is large. With parallel computing and a *message passing interface* (MPI), the sensitivity parameters are distributed to different processors and sensitivities are calculated locally. Thanks to the low communication and synchronization costs among processors, a nearly linear speed-up rate can be achieved (Jiang et al., 2004).

The DAE Jacobian from Eq. 1 can be evaluated either by a finite-difference approximation [we use (the default) relative perturbation size of 10^{-3}] or through automatic differentiation. Although a finite-difference approximation is generally acceptable for well-scaled problems, the automatic differentiation tool provides a better solution for highly stiff and nonlinear systems. Here, automatic differentiation techniques repeatedly apply the chain rule to the composition of elementary operations and compute derivatives that are correct up to machine precision. In this study, we use ADIFOR (Bischof et al., 1994), an automatic differentiation tool for FORTRAN programs developed at Argonne National Laboratory and Rice University. More detailed information on the interface between ADIFOR and DASPK can be found in Li and Petzold (1999).

Cyclic steady-state (CSS) convergence acceleration

PSA systems are operated in a cyclic manner with each bed repeatedly undergoing a sequence of steps such as pressurization, adsorption, blowdown, and desorption. After a startup time, the system approaches CSS, where the bed conditions at the start and end of each cycle in each bed are identical; this leads to normal production. The traditional way to determine a CSS is to repeatedly simulate a series of complete cycles until the bed conditions do not change from cycle to cycle. This successive substitution truly mimics the operation of a real plant but is usually very time consuming. As an alternative, we implement a Newton-based approach to directly determine CSS. CSS is written as a two-point boundary condition: $y_0 - y(t_{cycle}) = 0$ to solve for y_0. The sensitivity matrix (that is, the Jacobian of the CSS equations) in the Newton iteration comes from the DASPK sensitivity evaluation. This method attains a quadratic convergence rate near the solution and is thus faster than successive substitution. Ill-conditioning and high nonlinearity, which often arise from multiple adsorbent layers, nonisothermal effects, steep adsorption fronts, and trace components, are handled by the trust-region approach and a hybrid step, which is a combination of Newton and steepest-descent steps. Moreover, when design specifications such as purity and operating pressure are imposed, the CSS equations are augmented to include the design constraints to determine simultaneously the operating parameters with CSS. Details of this approach are described in Jiang et al. (2003).

PSA optimization

To optimize PSA processes, we typically seek to enhance product recovery or reduce operating cost (ϕ) subject to design specifications (W) at cyclic steady state (C). The decision variables (q) are geometric parameters, such as bed length and diameter, and process parameters such as flow rates, step times, and valve sizes. $F(y, y', q, t)$ is the bed model represented by DAEs. The optimization problem is expressed as

$$\begin{aligned} &\min \phi(y, y_0, q) \\ &\text{s.t.} \quad F(y, y', q, t) = 0 \\ &\qquad W[y(t, y_0, q)] \le 0 \\ &\qquad C(y_0) = y_0 - y(t_{cycle}) = 0 \\ &\qquad \text{LB} \le (y_0, q) \le \text{UB} \end{aligned} \tag{2}$$

Many previous efforts in optimizing PSA plants involve tradeoffs among model complexity, computation efficiency, and solver robustness. Some efficient solvers work well only with relatively simple models. Parameter studies or black box optimization approaches evaluate the effects of the design variables on the performance of PSA processes and obtain preliminary optimal designs, but they can be very costly and less efficient. To overcome these deficiencies, we develop a simultaneous tailored approach with the reduced-space successive quadratic programming (rSQP) algorithm (Ternet and Biegler, 1998).

In the tailored approach, the convergence of CSS is incorporated as a constraint to the optimizer, whereas the detailed bed models are solved in an inner loop, to obtain values of the constraints and objective function. Moreover, accurate gradients with respect to parameters are obtained through sensitivity evaluations. The CSS is not converged until the optimal solution is reached, and thus the time-consuming CSS convergence loop is eliminated. rSQP is well suited to optimize large nonlinear programming systems with relatively few decision variables by exploiting the problem structure. For a general nonlinear problem

$$\begin{aligned} &\min_x \quad f(x) \\ &\text{s.t.} \quad c(x) = 0 \\ &\qquad x^L \le x \le x^U \end{aligned} \tag{3}$$

where $f(x)$ is the objective function and $c(x)$ is a set of constraints; the variables are partitioned into independent and dependent variables. rSQP solves problem 3 by applying a Newton-like nonlinear programming method with a search direction defined as $d_k = Y_k p_Y + Z_k p_Z$. Here p_Y updates the dependent variables, which serve to improve the solution of the equality constraints, whereas p_Z updates independent variables, acts in the null space of the constraint gradients, and serves to optimize the objective function. More details on the rSQP algorithm can be found in Ternet and Biegler (1998). In addition, extensions of the rSQP algorithm to deal with adsorption processes are described in Jiang et al. (2003).

Process Description

To demonstrate our multibed optimization strategy we consider a typical five-bed PSA process to obtain high-purity hydrogen from a gas mixture of H_2 (73%), N_2 (0.5%), CO_2 (16%), CO (5%), and CH_4 (5.5%). This is similar to processes described in Malek and Farooq (1997) and Waldron and Sircar (2000). The adsorbent bed is packed with APHP and UOP 5A zeolite occupying one third and two thirds of the bed, respectively. As shown in Yang and Lee (1998), simultaneous usage of multiple kinds of adsorbent can effectively yield high-purity product because each adsorbent has a specific selectivity for a certain gas. The APHP layer does a bulk separation and the zeolite purifies the raffinate stream. All of the model parameters (Weist, 2001) and problem data for this process are listed in the Appendix.

The PSA process undergoes 11 steps with the following base case step times:

(1) feed product and repress gas (140 s)
(2) first cocurrent depress (EQ1) (30 s)
(3) second cocurrent depress (EQ2) (40 s)
(4) provide purge (70 s)
(5) third cocurrent depress (EQ3) and blowdown (BD1) (30 s)
(6) countercurrent blowdown (40 s)
(7) receive purge (70 s)
(8) third equalization repress (EQ3) (30 s)
(9) second equalization repress (EQ2) (40 s)
(10) first equalization repress (EQ1) (30 s)
(11) product hydrogen repress (110 s)

Pressure equalization steps (EQ) allow a significant saving in overall process energy consumption because less mechanical energy is required to repressurize the purged, low-pressure bed. The product recovery also increases because less feed/product gas is required to repressurize the bed. However, these benefits are achieved at the cost of a significant reduction in the H_2 productivity (Waldron and Sircar, 2000). An idle step follows step 9 and synchronizes the five-bed operation. The effluent in step 1 is the high-purity H_2. The flowchart and cycle sequence are shown in Figure 1 and Table 1, respectively.

Mathematical modeling

The model used in this study is taken from standard industrial simulation models for PSA beds (Kumar et al., 1994; Waldron and Sircar, 2000). Using a plug flow assumption, the N-component material balance for the bulk phase in the adsorption bed is given by

$$\varepsilon_B \frac{\partial \rho_i}{\partial t} + \rho_s \frac{\partial n_i}{\partial t} + \frac{\partial (\rho_i v)}{\partial z} = 0 \qquad i = 1, 2, \ldots, N_c \tag{4}$$

All variables are defined in the Notation section and all model parameters are supplied in the Appendix. Assuming thermal equilibrium between fluid and particles and considering the heat loss through the wall, the energy balance for the gas and solid phases is given by

$$\left[\varepsilon_B \sum_{i=1}^{N} \rho_i (c_p^i - R) + \rho_s \left(c_s + \sum_{i=1}^{N} n_i c_p^i \right) + \frac{4}{D} c_{pw} \rho_w h_w \right] \frac{\partial T}{\partial t} - \rho_s \sum_{i=1}^{N} q_i \frac{\partial n_i}{\partial t} + \frac{4}{D} h_{tc} (T - T_{amb}) + \frac{\partial (vh)}{\partial z} = 0 \tag{5}$$

where the enthalpy is calculated by

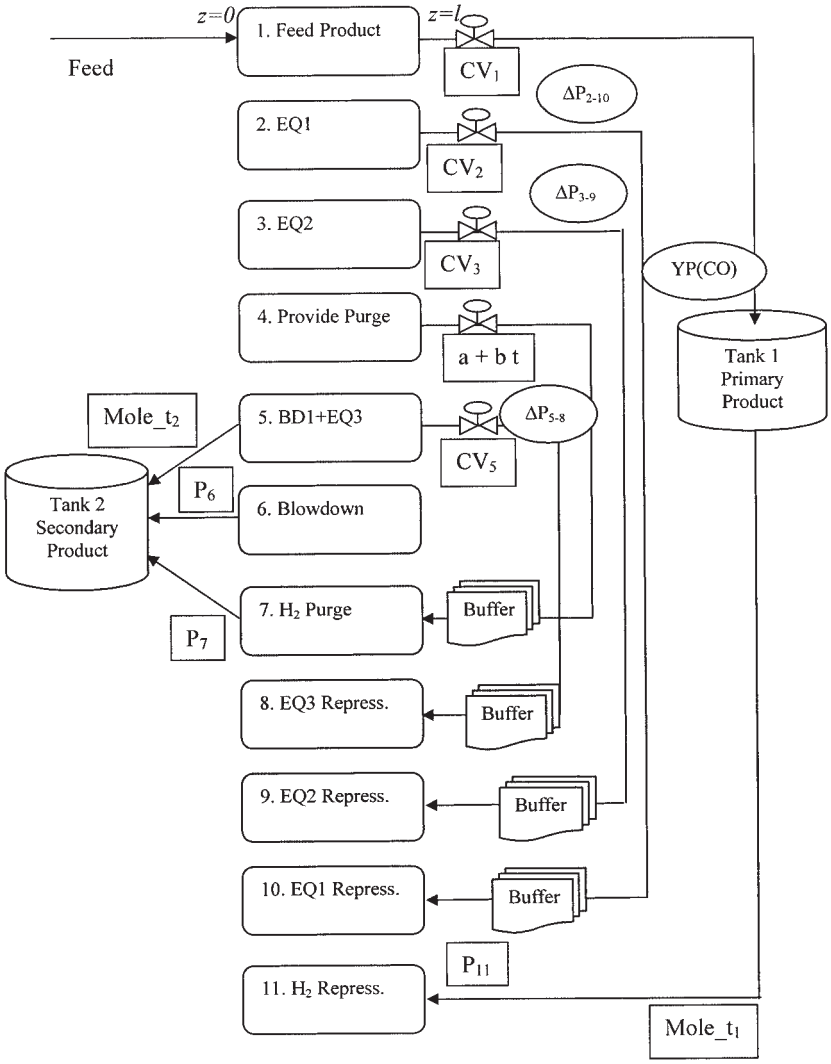

Figure 1. Flow sheet of 5-bed PSA system.

$$h = \sum_i \rho_i [c^i_{p,A} T + (c^i_{p,B}/2) T^2 + (c^i_{p,C}/3) T^3 + (c^i_{p,D}/4) T^4]$$

$$-\frac{\partial P}{\partial z} = 150 \frac{\mu \nu (1-\varepsilon_B)^2}{d_p^2 \varepsilon_B^3} + 1.75 \frac{\rho M \nu^2 (1-\varepsilon_B)}{d_p \varepsilon_B^3} \quad (6)$$

Here, we assume that Ergun's equation can be used to describe the relation between superficial velocity and pressure drop along the bed

where ρ and M are the bulk gas molar density and molecular weight, respectively. For equilibrium-based separations, the adsorption rate is described by the linear driving force (LDF)

Table 1. Time Chart of Five-Bed System

	Stage 1			Stage 2			Stage 3			Stage 4			Stage 5		
Bed A	Feed			EQ1	EQ2	Pr purge	EQ3	Blow-down	Purge	EQ3	EQ2	Idle	EQ1	Repress	
Bed B	EQ1	Repress		Feed			EQ1	EQ2	Pr purge	EQ3	Blow-down	Purge	EQ3	EQ2	Idle
Bed C	EQ3	EQ2	Idle	EQ1	Repress		Feed			EQ1	EQ2	Pr purge	EQ3	Blow-down	Purge
Bed D	EQ3	Blow-down	Purge	EQ3	EQ2	Idle	EQ1	Repress		Feed			EQ1	EQ2	Pr purge
Bed E	EQ1	EQ2	Pr purge	EQ3	Blow-down	Purge	EQ3	EQ2	Idle	EQ1	Repress		Feed		

Table 2. Boundary Conditions

Step k	$z = 0$	$z = l$
Step 1	Constant flow of feed	Valve controlled (CV_1), feed to tank 1
Step 2	Zero velocity	Valve controlled (CV_2), feed to 10
Step 3	Zero velocity	Valve controlled (CV_3), feed to 9
Step 4	Zero velocity	Time-dependent flow $= a + bt$, feed to 7
Step 5	Constant flow (mole_t_2) with check valve, feed to tank 2	Valve controlled (CV_5), feed to 8
Step 6	Constant flow (mole_t_2) with check valve, feed to tank 2	Zero velocity
Step 7	Constant flow (mole_t_2) with check valve, feed to tank 2	Receive flow from 4
Step 8	Zero velocity	Receive flow from 5
Step 9	Zero velocity	Receive flow from 3
Step 10	Zero velocity	Receive flow from 2
Step 11	Zero velocity	Constant flow from tank 1 (mole_t_1)

model, with a single lumped mass-transfer parameter k_i, as follows

$$\frac{\partial n_i}{\partial t} = \frac{k_i}{RT}(P_i - P_i^*) \tag{7}$$

Here we choose to use partial pressure as the driving force rather than solid phase loading because partial pressure is more directly related to the chemical potential, which thermodynamics suggests is appropriate. Moreover, this LDF model and its parameters have been verified in numerous industrial applications (Weist, 2001).

The equilibrium isotherm uses the dual-site Langmuir form, which has been widely adopted for these industrial applications (Weist, 2001), as follows

$$n_i^* = m_1 \frac{b_i P_i^*}{1 + \sum_{j=1}^{N} b_j P_j^*} + m_2 \frac{d_i P_i^*}{1 + \sum_{j=1}^{N} d_j P_j^*}$$

$$b = b_0 \exp\left(\frac{q_{ib}}{RT}\right) \qquad d = d_0 \exp\left(\frac{q_{id}}{RT}\right) \tag{8}$$

At the entrance ($z = 0$) and exit ($z = l$) ends of the bed are empty volumes free of adsorbents, which are modeled as fixed-volume CSTRs. They are nonisothermal and nonisobaric and are described by simplified material and energy balances without adsorption. The two product tanks connecting to the beds are isothermal and isobaric, and are described by material balances only. Initially, the beds are filled with pure hydrogen. The boundary conditions for each step are given in Table 2.

The velocity through the valves in steps 2, 3, and 5 are calculated as follows. Let P_H denote the inlet pressure and P_L denote the outlet pressure from the valve. If $P_L > P_H$ then $v = 0$; otherwise, calculate the critical pressure ratio $P_{crit} = (2/(1 + \gamma))^{\gamma/(1-\gamma)}$ and define

$$v = \begin{cases} CV \quad P_H \sqrt{\left| \dfrac{1 - (P_L/P_H)^2}{MT} \right|} & \text{if } P_H/P_L < P_{crit} \\ CV \quad P_H \sqrt{\left| \dfrac{1 - P_{crit}^{-2}}{MT} \right|} & \text{otherwise} \end{cases} \tag{9}$$

where $\gamma = C_p/C_v$.

In addition, a check valve is implemented to prevent the bed pressure from dropping too low during depressurization steps 5, 6, and 7 and disrupting calculations. A damping factor (based on $P(0)$, the bed pressure at $z = 0$) multiplies the nominal flow (mole_t_2) to give the actual flow to tank 2. This factor is calculated by

$$\frac{\text{Actual flow}}{\text{Nominal flow}} = factor = \begin{cases} 1, & P(0) \geq 1.0 \\ \dfrac{1}{2}\left(1 + \sin \pi \dfrac{2P(0) - 1.1}{1.8}\right), & 1.0 > P(0) \geq 0.1 \\ 0, & P(0) \leq 0.1 \end{cases} \tag{10}$$

Unibed Framework

Because all beds are assumed identical and undergo identical process steps in a sequential manner, we require only one bed to simulate a multibed cycle. To describe bed connectivities, temporal effluent arrays (flow, pressure, enthalpy) from a process step are retained and used later when the bed is undergoing the appropriate corresponding step (Kumar et al., 1994). For instance, when calculating the effluent flow through the valve during step 2, we use the pressure from step 10 as back pressure P_L. Meanwhile, the flow is temporarily stored as a function of time in a storage buffer. During step 10, the moles and enthalpy stored at step 2 are used as influents, and the process pressures at step 10 are stored for use by step 2 in the next cycle. Steps 3 and 9, steps 4 and 7, and steps 5 and 8 are simulated in a similar manner. The effluent streams and back pressures are stored at discrete time intervals in the storage buffers shown in Figure 1. Linear interpolation is used to obtain information between two time points. Because only a limited set of information is stored, some precision is lost. An overall balance for the step is used to ensure that the procedure does not bias the calculations and to indicate whether more temporal points should be used. The bed starts the next cycle using the final conditions attained at the previous cycle. The process is assumed to have reached cyclic steady state when the bed conditions at the beginning of step 1 match the conditions at the end of step 11 of the same cycle. The computational procedure is shown in Figure 2.

The unibed approach leads to a low problem size and computation time, given that we solve for only one bed. However, it may not accurately predict the transient behavior, given that "unibed" operation does not exist in reality. On the other hand, if one is most interested in the cyclic steady-state solution, this

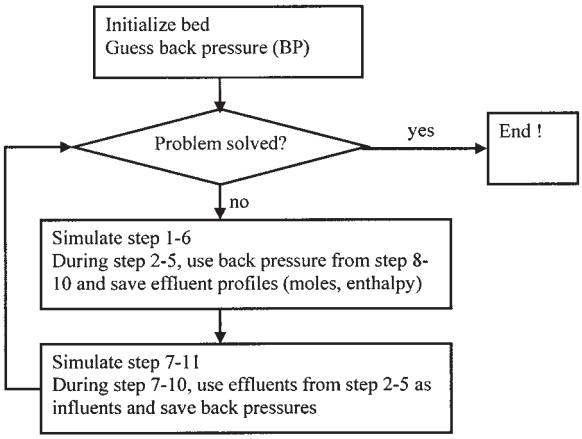

Figure 2. Computational procedure for unibed framework.

is not a significant drawback because unibed and multibed approaches yield the same CSS.

Because of the data buffers in Figure 1, the sensitivity calculation is difficult under the unibed framework because sensitivity parameters include not only the bed conditions at the current step but the bed conditions of the providing step, which appear at boundary conditions. Assuming that step l feeds step k, the sensitivity equations for step k are

$$F_k[t, y^k, y^l, (y^k)', q] = 0$$

$$\frac{\partial F_k}{\partial y^k} s_i^k + \frac{\partial F_k}{\partial y^l} s_i^l + \frac{\partial F_k}{\partial (y^k)'} (s_i^k)' + \frac{\partial F_k}{\partial q_i} = 0 \qquad i = 1, \ldots, N_q$$

where

$$s_i^k = \frac{\partial y^k}{\partial q_i} \qquad s_i^l = \frac{\partial y^l}{\partial q_i} \tag{11}$$

and y^k and y^l are the state variables for steps k and l, respectively. Here, we need to store intermediate sensitivities s_i^l during step l, multiply them by $\partial F_k/\partial y^l$ and modify the sensitivity equations. This work is nontrivial and takes significant effort and workspace. Alternatively, we may perturb the entire cycle and evaluate

$$\frac{\partial y}{\partial q_i} = \frac{y(q_i + h, t_{cycle}) - y(q_i, t_{cycle})}{h}$$

This is very convenient because the perturbed parameters enter step k from step l through boundary conditions and automatically propagate their influence. Now there is no need to store the sensitivity profiles. However, with perturbation, the gradient accuracy diminishes because it is controlled by the perturbation size and DAE integration error tolerance. For simplicity, we adopt the perturbation method to generate sensitivities under the unibed framework.

Both black box and simultaneous tailored approaches are used to solve the optimization problem. With the black box approach, the optimizer determines values for the decision variables q, only. Based on these variables the bed models are executed until CSS is reached. At CSS, the objective function, constraints, and their derivatives with respect to the decision variables are evaluated and returned to the optimizer. Because many CSS are converged at every iteration, the black box approach is very time consuming. To determine optimal values for the decision variables, we apply rSQP as the optimization algorithm. To generate sensitivities, we perturb one parameter at a time, execute the bed model through CSS, and evaluate the deviations in the objective and constraint functions at CSS. Here the accuracy of the gradients diminishes further because it is controlled by the DAE integration error, perturbation size, and error tolerance for CSS. With the simultaneous tailored approach and rSQP, CSS constraints are not converged at each iteration of the optimization algorithm. On the other hand, the optimization problem now includes the initial conditions y_0 as well as the design variables q in the optimization problem.

Table 3. Multibed Chart (Stage 1)

Stage 1			
	Period 1	Period 2	Period 3
Bed A	Feed (1)		
Bed B	EQ1 (7)	Repressurization (8)	
Bed C	EQ3 (5)	EQ2 (6)	Idle
Bed D	EQ3 (2)	Blowdown (3)	Purge (4)
Bed E	EQ1 (2)	EQ2 (3)	Provide purge (4)

Tank 1

Tank 2

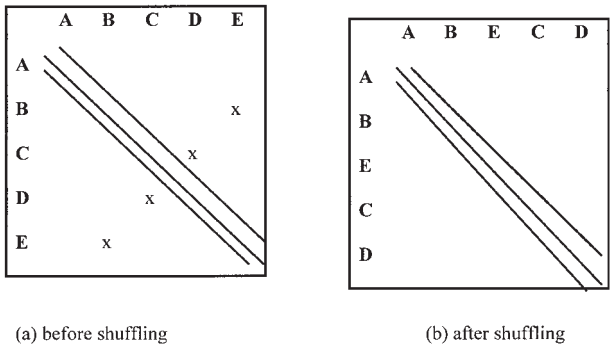

Figure 3. Banded structure of the DAE Jacobian and bed interconnectivity at period 1.

Large-scale optimization methods like rSQP are required to handle these problems efficiently.

Multibed Framework

A notable feature of the five-bed PSA process sequence is the repetitive stages within each cycle. For the convenience of modeling, a cycle can be viewed as consisting of repetitive blocks of five stages as in Table 1. Each stage can be further divided into three periods and defining the duration of three periods is sufficient to define the cycle duration as $t_c = 5(t_{p1} + t_{p2} + t_{p3})$. The process symmetry allows for an efficient way of simulation: we solve all beds simultaneously but only for one stage of the cycle; this results in significant savings of computation time. Table 3 depicts the multibed time chart.

At the end of Stage 1, the beds reinitialize themselves by using the final conditions from each other. As seen from Table 1, the switching order is as follows: bed A provides its final conditions to initialize bed E; bed B to bed A; bed C to bed B; bed D to bed C; and bed E to bed D. After the beds are reinitialized, the simulation starts again. After a number of cycles are simulated by successive substitution, we match final and initial conditions among these beds and enforce them as CSS equality constraints in the optimization problem.

Unlike unibed, the multibed approach depicts the actual dynamic cycle and accurately predicts both the transient and

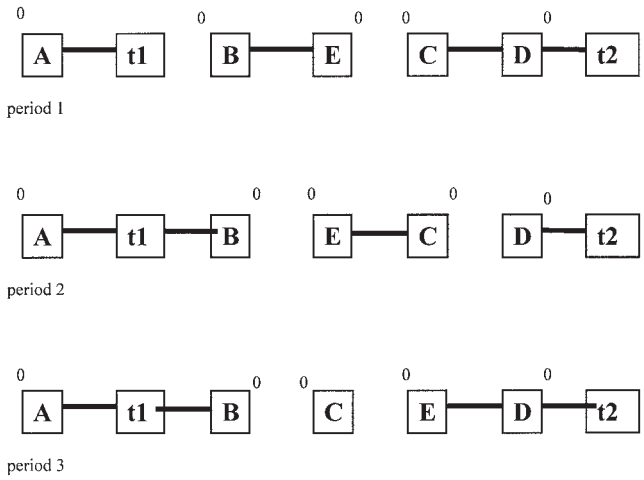

Figure 4. Array of differential variables at three periods.

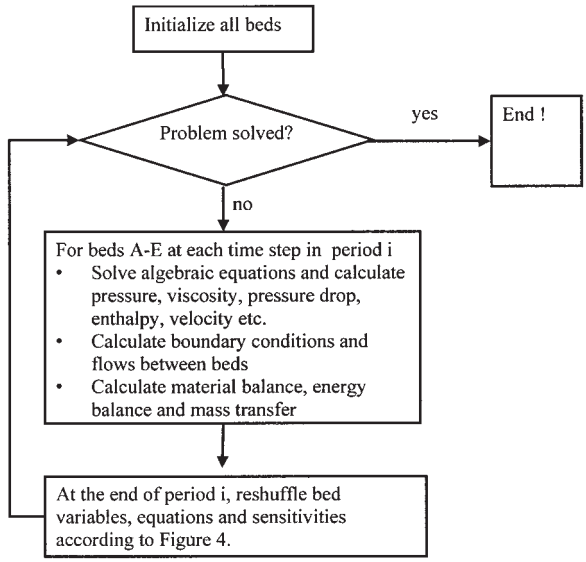

Figure 5. Computation flow sheet under multibed framework.

steady-state behaviors. However, this is accomplished at the cost of solving a larger and stiffer DAE system. The simultaneous integration of five beds is computationally expensive and requires a much larger workspace than unibed.

When integrating the DAEs inside DASPK, we prefer a banded structure for the DAE Jacobian. Unfortunately, the banded structure is destroyed by the bed interconnectivity. For instance, bed B and bed E are connected at period 1. If the differential variables are aligned in the order of A, B, C, D, and E, we have to use a large bandwidth to include the interconnectivity between B and E (Figure 3a). A large bandwidth results in longer computation time for DASPK, which is undesirable. Furthermore, because only a few nodes from each of the two beds are involved in bed interconnectivity, it is usually not worthwhile to increase the bandwidth to include all of the nodes in the bed.

To allow our sensitivity calculations to exploit an efficient banded linear solver, we develop a bed-shuffling scheme to avoid the use of large bandwidth (Figure 3b). According to the bed connectivity at different periods, the bed variables are shuffled and repeatedly reoriented. Figure 4 shows the bed alignment at different periods; "0" stands for the entrance ($z = 0$) of the bed. Because exit ends ($z = l$) are connected during equalization steps, they are placed next to

dy/dp	A	B	C	D	E
A	xx	0	0	0	0
B	x	xx	0	0	xx
C	0	x	xx	xx	xx
D	0	x	xx	xx	xx
E	0	xx	xx	xx	xx

Figure 6. Bed sensitivity matrix.

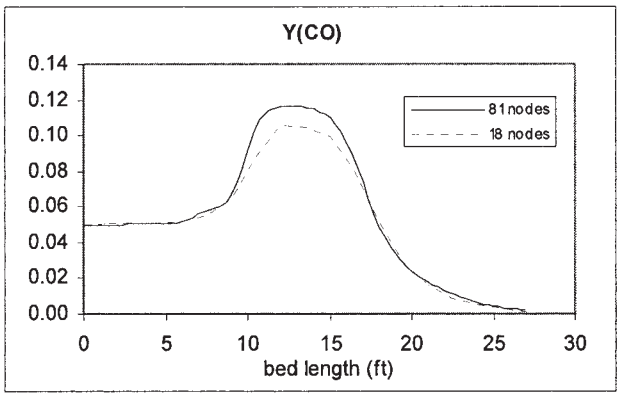

Figure 7(a) Mole fraction of carbon monoxide

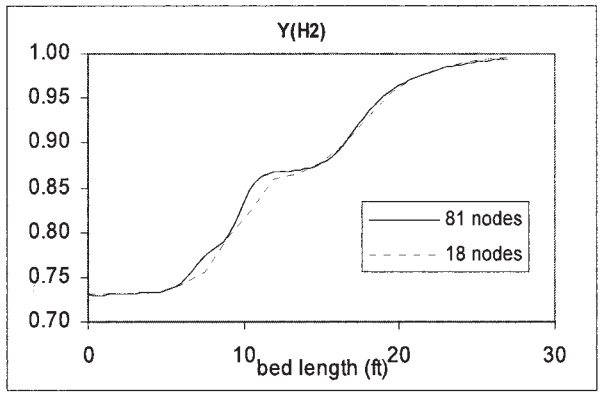

Figure 7(b) Mole fraction of hydrogen

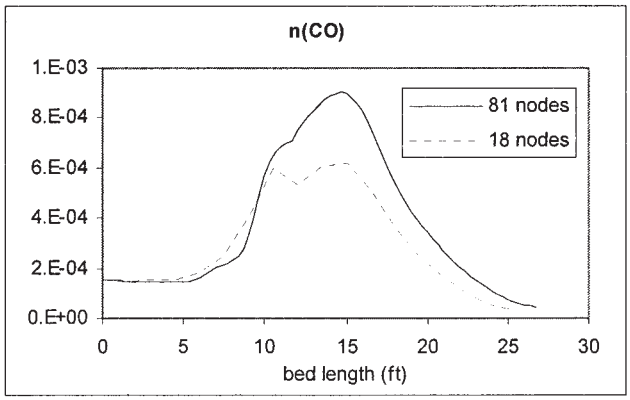

Figure 7(c) Solid loading of carbon monoxide

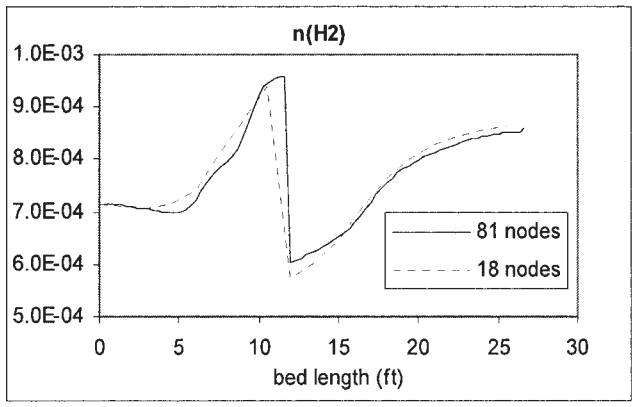

Figure 7(d) Solid loading of hydrogen

Figure 7. (a)–(d). Comparison of bed profiles with 18 and 81 nodes under base conditions at CSS, end of feed step.

each other in the differential variable array to minimize the bandwidth. The only exception is bed D, where the gas is withdrawn from the bed entrance so the entrance sits next to tank 2. At the end of each period, the array of differential variables is reshuffled before starting the next period (Figure 4). The computational flow sheet under multibed framework is illustrated in Figure 5. Note that the multibed framework eliminates the need for intermediate profile storage. Each bed has up-to-date states that are ready to use by other beds for connecting flows. The sensitivity calculation is also straightforward, except that the sensitivities have to be shuffled along with the state variables at the end of each period.

As we inspect the sensitivities of one bed with respect to another, the matrix is fairly sparse (Figure 6). Some blocks are zeros because of no bed connectivity, whereas blocks with "xx" indicate strong connections between two beds, given that they are directly linked during at least one period (such as B–E, C–D, C–E, D–E). Other blocks ("x") indicate smaller sensitivities because of weaker connections. In a given period, these beds are influenced either through a product tank (such as A–B) or through a third bed (such as B–E–C, B–E–D).

Trace components

Trace components are defined for those with quantities that are negligible in the system. As the raffinate becomes increasingly richer in hydrogen along the adsorption bed, the other four components gradually decrease in the gas phase and eventually become trace quantities. Trace components make the DAE integration error harder to control. When the perturbation method is used to generate derivatives, the tiny perturbation sizes of the trace components cause the derivatives to be overwhelmed by numerical noise. It is also inappropriate to assume that trace components exercise little influence over other variables and approximate those sensitivities by zeros. In fact, some derivatives with respect to trace components are not trivial. The magnitudes of the sensitivities depend on the variables' location in the bed (such as entrance, exit) and species (such as gas density, solid loading). Because these components cannot be ignored, we treat all variables on an equal basis and evaluate all sensitivities inside DASPK, even though DASPK often has to take extra time steps to satisfy the error tolerance for the trace components. With the use of ADIFOR instead of finite differencing for internal Jacobian and sensitivity equations, DASPK can produce accurate sensitivities even for trace components.

With the trace components at the ppm purity level, the rSQP optimizer may complain about inconsistent linearization of the constraint functions. Thus, we assign an artificial variable to each constraint and add a penalty to the original objective function. The penalty function, which penalizes the constraint

violation, is the sum of all artificial variables augmented by a large positive number (the "big M" method). Thus

$$\begin{aligned} &\min \quad f(x) \\ &\text{s.t.} \quad g(x) \le 0 \quad i = 1, 2, \ldots, n \end{aligned} \tag{12}$$

is transformed to

$$\begin{aligned} &\min \quad f(x) + M \sum_{i=1}^{n} a_i \\ &\text{s.t.} \quad g_i(x) + \sigma_i = a_i \quad \sigma_i, a_i \ge 0 \end{aligned} \tag{13}$$

where M is a large positive constant, a_i is an artificial variable and σ_i is a slack variable. By adding artificial variables, we loosen the constraints and remove any infeasibility. By minimizing the modified objective function, the optimizer eventually drives the artificial variables to zero at the optimum so the optimality condition is unaffected.

Computational Results

The five-bed, 11-step hydrocarbon PSA cycle (Figure 1) is used to demonstrate the simulation and optimization methodologies. The results of unibed and multibed are shown in this section.

Unibed results

The adsorption bed is discretized into 81 nodes along the axial direction. Each node has 11 differential variables, which are gas densities and solid loadings for five species as well as one temperature variable. The additional differential variables are the moles in two product tanks, the gas densities and temperature at two voids at bed ends, and the flow of mass and energy at an end of the bed. The total number of differential variables for the DAE system is $81*(2*N_{comp} + 1) + 2*(N_{comp} + 1) + 2*N_{comp} + (N_{comp} + 1) = 919$. The equilibrium partial pressure at each node is determined from a dual-site Langmuir isotherm using a Newton solver in an inner loop. The DAE integration was performed using DASPK 3.0, with the banded (ML = MU = 3*nodes) option and a finite-differencing-generated internal Jacobian. The integration time is made dimensionless. During each connecting step, intermediate effluents are stored at 20 time points in the storage buffer.

After we simulate the bed models for 250 cycles, the bed reaches cyclic steady state. These profiles have also been validated using the SIMPAC simulator at APCI (Kumar et al., 1994). More information on these solutions can be found in Jiang (2004).

With 81 spatial nodes, each cycle evaluation takes about 120 s on a 1-GHz machine. To reduce the computational cost, we attempt to use fewer bed nodes. We compare the bed profiles under 81 and 18 nodes (Figures 7a–d) and find that 18 nodes can grasp the basic shape of the bed profiles fairly well, although some fronts are less sharp. The number of equations with 18 nodes is 226 and the DAE integration takes less than a minute per cycle. As a result, 18 nodes are used for all subsequent experiments.

For optimization, we seek to maximize H_2 recovery while maintaining the CO impurity below the 10 ppm level and the bed pressures at specific targets at cyclic steady state. H_2 recovery (RC) is the moles of H_2 produced per mole of H_2 feed per cycle. The CO impurity $YP(CO)$ is measured as the mole fraction of CO in the product tank 1 at the end of feed step. The pressure constraints are imposed for the following reasons. To maintain steady flows between providing and receiving steps, a pressure difference (10.13 kPa) is required. The constraint on bed pressure at the end of steps 6 and 7 is set by the requirements of a downstream fuel gas header. The only exceptions are P_6 and P_7, which are measured at the bed entrance. The optimization problem formulation is as follows

$$\begin{aligned} &\max \quad H_2 \text{ recovery} \\ &\text{s.t.} \quad y(t_{cycle}) = y_0 \text{ (CSS conditions)} \\ &\qquad YP(CO) \le 10 \text{ ppm} \\ &\qquad P_1 \le 2502.7 \text{ kPa} \\ &\qquad P_2 - P_{10} \ge 10.13 \text{ kPa} \\ &\qquad P_3 - P_9 \ge 10.13 \text{ kPa} \\ &\qquad P_5 - P_8 \ge 10.13 \text{ kPa} \\ &\qquad P_6 \ge 170.73 \text{ kPa} \\ &\qquad P_7 \ge 170.73 \text{ kPa} \\ &\qquad P_{11} \le 2482.5 \text{ kPa} \end{aligned} \tag{14}$$

To synchronize five-bed operations, the step times are constrained by the following system of equations

$$\begin{gathered} t_{cycle} = t_1 + t_2 + \cdots + t_{11} + t_{idle} \\ t_1 = t_{cycle}/5 \\ t_{idle} = t_{cycle}/10 \\ t_2 = t_{10} \\ t_3 = t_9 \\ t_5 = t_8 \\ t_4 = t_7 \\ t_2 + t_3 = t_5 + t_6 \\ t_5 + t_6 + t_7 = t_1 \\ t_{10} + t_{11} = t_1 \\ t_2 + t_3 = t_1/2 \end{gathered} \tag{15}$$

There are 13 time variables, 11 time constraints, and 2 degrees of freedom. We choose total cycle time t_{cycle} and step time t_5 to be the free variables and calculate all other step times based on these two.

The decision variable set includes two step times (t_5, t_{cycle}), four valve constants (CV_1, CV_2, CV_3, CV_5), four molar flow rates (mole_t_1, mole_t_2, a, b), and bed diameter. The upper and lower bounds for the decision variables are listed in Table 4.

Table 4. Upper and Lower Bounds for the Decision Variables

Variable Name	Symbol	UB	LB
Step time	t_5, t_{cycle}	150% of start pt.	50% of start pt.
Bed diameter	D	3.66 m	1.52 m
Valves	CV_1, CV_2, CV_3, CV_5	150% of start pt.	50% of start pt.
Flow	mole_t_1, mole_t_2	150% of start pt.	50% of start pt.
Flow	a, b	$+\infty$	0

Table 5. Optimization Results under Unibed Framework

	CV_1	CV_2	CV_3	CV_5	Mole_t_2	a	b	Mole_t_1	t_{cycle}	t_5	D
Base	0.3	0.054	0.075	0.036	1180	250	6.25	636.4	700	30	2.13
BB	0.3	0.055	0.075	0.037	1180.1	250	6.24	636.4	700.1	30.34	2.13
ST	0.3	0.056	0.09	0.036	1182	250	5.3	634.6	700	30.23	2.13

	$YP(CO)$	P_1	P_2–P_{10}	P_3–P_9	P_5–P_8	P_6	P_7	P_{11}	RC (H_2)
Base	1.0×10^{-3}	2500.7	50.66	0	43.57	173.3	198.6	2371	85.79%
BB	4.6×10^{-4}	2495.6	31.41	10.13	36.48	169.2	194.5	2381.1	85.94%
ST	2.9×10^{-4}	2494.6	26.34	10.13	56.74	175.3	172.3	2381.1	85.94%

	$\left\|\frac{\nabla_x L(x_k, \lambda_k)}{\|f(x_k)\| + 1}\right\|$	$\|c(x_k)\|_\infty$	No. of Iterations	CPU Hours
BB	3.9×10^{-4}	1.2×10^{-2}	5	20.5 h, 7142 cycle evaluation
ST	1.67×10^{-3}	5.3×10^{-3}	21	18.5 h, 6442 cycle evaluation

We first apply the black box approach using the perturbation method for generating sensitivities. The optimization problem consists of 11 variables and eight constraints. We experiment with several perturbation sizes (0.1%, 0.5%, and 1%), CSS tolerances (0.005, 0.01, 0.05), and DAE integration error tolerances (10^{-6}, 10^{-7}) and experience trade-offs in terms of accuracy and time efficiency. Note that a large perturbation step is a poor approximation of the derivative but a small perturbation step is overwhelmed by numerical noise. Here the CSS tolerance must be chosen to reflect the integration error. The more stringent the integration error is, the more accurate the CSS is, and the longer time and more cycles it takes to reach CSS and evaluate functions and derivatives. After several trials, 0.5%, 0.01, 10^{-6} appear to be a good combination for perturbation size, CSS tolerance, and DAE integration tolerance, respectively.

We also apply the simultaneous tailored approach with the perturbation method. Given that CSS is not required to converge at each iteration, the error caused by approximating CSS is eliminated. In addition to the 11 parameters, back pressures and bed variables are also included in the decision variables set and are updated by the optimizer at each iteration. The constraint set also includes CSS conditions and the deviations between the calculated process pressures and the back pressures assigned by the optimizer. The numbers of variables and constraints grow significantly by (210 + 60) with these additions.

The optimization results of black box (BB) and simultaneous tailored (ST) approaches are shown in Table 5. The computer programs were run on an i686 1-GHz Linux machine. The black box approach takes many more cycle evaluations at each iteration than the tailored approach because CSS has to converge for every function evaluation and derivative calculation.

Unfortunately, the optimizer moves for both approaches are quite sluggish and the decision variables seldom change. The purity constraints are not active. This is surprising, given that active purity constraints can lead to higher recoveries. Although all constraints are satisfied, the convergence is not robust. For instance, raising the pressure constraints by 10.13 kPa caused the optimizer to fail to converge the constraints. Therefore, with poor sensitivities generated by perturbation, the optimizer is unable to converge to a good optimum.

Multibed results

To obtain good sensitivities, we apply the multibed framework and implement the direct sensitivity approach. The total number of ODEs is 5 beds$*$[18$*$(2$*N_{\text{comp}}$+1) + 2$*$(N_{comp} + 1)] + 2 tanks $*N_{\text{comp}} = 1060$. The decision variables consist of bed conditions and 11 parameters (back pressures are unnecessary for multibed). The constraints include CSS conditions as well as pressure and purity. We use the three periods t_{p1}, t_{p2}, and t_{p3} to substitute for t_5 and t_{cycle}. The time periods are further determined by $t_{p1} + t_{p2} = t_{p3}$, which still leaves us with two

Table 6. Optimal Conditions at 10, 100, and 1000 ppm under Multibed Framework

	CV_1	CV_2	CV_3	CV_5	mole_t_2	a	b	mole_t_1	t_{p1}	t_{p2}	t_{p3}	D
LB	0.15	0.027	0.0375	0.018	590	0	0	318.2	22.5	30	52.5	1.52
UB	0.45	0.081	0.1125	0.054	1770	$+\infty$	$+\infty$	954.5	37.5	50	87.5	3.66
1000 ppm	0.262	0.07	0.07	0.054	1114.9	0	7.92	629.6	37.5	50	87.5	2.33
100 ppm	0.259	0.074	0.073	0.054	1152.5	0	9.09	636.1	37.5	50	87.5	2.41
10 ppm	0.255	0.078	0.076	0.054	1187	0	10.22	618.4	37.5	50	87.5	2.49

	P_1	P_2–P_{10}	P_3–P_9	P_5–P_8	P_6	P_7	P_{11}	RC (H_2)
1000 ppm	2502.73	10.13	10.13	14.19	170.73	170.7	2467.3	88.64%
100 ppm	2502.73	10.13	10.13	10.13	170.73	170.7	2420.7	87.25%
10 ppm	2502.73	10.13	10.13	10.13	170.73	170.7	2354.8	85.88%

	$\left\|\frac{\nabla_x L(x_k, \lambda_k)}{\|f(x_k)\| + 1}\right\|$	$\|c(x_k)\|_\infty$	$\|Yp_Y\|$	$\|Zp_Z\|$
1000 ppm (AD)	8.2×10^{-7}	1.2×10^{-4}	7.6×10^{-4}	1.8×10^{-3}
100 ppm (AD)	1.1×10^{-5}	1.3×10^{-4}	4.6×10^{-4}	6.4×10^{-4}
10 ppm (AD)	6.4×10^{-6}	6.8×10^{-4}	1.3×10^{-3}	1.4×10^{-3}

Table 7. Different Optimal Conditions Yielded by Finite Differencing (FD) and Automatic Differentiation (AD) at 1000 ppm Case

1000 ppm	CV_1	CV_2	CV_3	CV_5	mole_t_2	a	b	mole_t_1	t_{p1}	t_{p2}	t_{p3}	D
FD	0.298	0.074	0.074	0.046	1137	251.6	3.06	637.8	37.5	50	87.5	2.38
AD	0.262	0.07	0.07	0.054	1114.9	0	7.92	629.6	37.5	50	87.5	2.33

	P_1	P_2–P_{10}	P_3–P_9	P_5–P_8	P_6	P_7	P_{11}	RC (H_2)
FD	2495.6	10.13	10.13	35.46	170.7	170.7	2436.9	87.70%
AD	2502	10.13	10.13	14.19	170.7	170.7	2467.3	88.64%

	$\left\|\frac{\nabla_x L(x_k, \lambda_k)}{\|f(x_k)\| + 1}\right\|$	$\|c(x_k)\|_\infty$	$\|Yp_Y\|$	$\|Zp_Z\|$
FD	6.3×10^{-4}	9.5×10^{-3}	0.11	0.11
AD	8.2×10^{-7}	1.2×10^{-4}	7.6×10^{-4}	1.8×10^{-3}

degrees of freedom. This formulation is consistent with Eq. 15. The new optimization problem has (12 + 1050) variables and (9 + 1050) constraints. We considered three different CO impurity levels at 10, 100, and 1000 ppm; the optimal conditions are shown in Table 6. DASPK package is used to calculate sensitivities with the staggered corrector method. Two different options, ADIFOR seed matrix (AD) and default finite differencing (FD), are used to evaluate the DAE Jacobian and sensitivity equations. Their impacts on the optimization results are compared in Table 7. Because ADIFOR generates more accurate sensitivities than the FD option, higher recoveries are achieved. Also, because finite differencing generates less accurate derivatives, it encounters optimizer failure at an earlier stage and is unable to converge to tight tolerances. The computer programs run for 50–200 h on a 2.4-GHz Linux machine.

We observe a trade-off between H_2 purity and H_2 recovery. H_2 recovery increases as the H_2 purity is reduced. However, the penalty for producing higher purity H_2 is not large. A substantial upgrade in the CO impurity from 1000 to 10 ppm results in a decrease of only 2.8% H_2 recovery, which is consistent with results reported by Waldron and Sircar (2000). The 100 and 10 ppm cases were run with the addition of artificial variables. Comparing Tables 6 and 8, we see that longer production times (t_{p1}, t_{p2}, t_{p3}) lead to higher H_2 recovery but this is achieved at the cost of a lower production rate.

The optimizer guarantees only a local optimum, so different starting conditions may result in different optimal solutions. In this case, it appears that the problem has a shallow response surface in the neighborhood of the optimum; differences in the two solutions can also be dictated by convergence error. We observe this by comparing two solutions for the 1000 ppm case with ADIFOR (Table 8), with solutions obtained from different starting points.

Although the optimization results are achieved with 18-bed nodes, the simulation results for 18 and 81 nodes are fairly close and it is safe to use the optimal conditions from 18-node cases on the 81 nodes. Table 9 compares the pressure, purity, and recovery between 18 and 81 nodes under optimal conditions for 10 ppm purity. More end-of-step (EOS) bed profiles are shown in Figure 8. Note that optimal values of the objective function change only slightly, indicating that the response surface for this problem is fairly flat.

Conclusions

The parameter study by Waldron and Sircar (2000) finds that various combinations of process steps, operating conditions, and feed gas conditions can be used to produce the same product purity with different degrees of H_2 recovery. Our experiences have confirmed this. H_2 recovery and purity are fairly robust objectives and change only slightly with operating conditions. This creates some flexibility in the design and operation of H_2 systems.

Although 81 nodes provide a more precise description of the bed profiles, the long computation time and huge memory usage are prohibitive. With 81 nodes under a multibed framework, the DAE system is 81/18 = 4.5 times as large as the 18-node case and the memory usage by ADIFOR and sensitivity package is $(4.5)^2$ or about 20 times as large. Currently, the memory usage by the 18-node optimization program is 300 Mbyte. We estimate that the memory usage by 81 nodes is beyond the current capacity of our computer cluster.

We have adopted both black box and simultaneous tailored frameworks to solve optimization problems. With the black box approach, the optimizer solves a small system with few decision variables and constraints. The sensitivities, with respect to decision variables, are evaluated at CSS

Table 8. Higher Recovery with Longer Production Time for 1000 ppm Case

CV_1	CV_2	CV_3	CV_5	mole_t_2	a	b	mole_t_1	t_{p1}	t_{p2}	t_{p3}	D
0.26	0.069	0.069	0.048	1103	0	6.03	630.1	46.6	62.5	109.1	2.58

P_1	P_2–P_{10}	P_3–P_9	P_5–P_8	P_6	P_7	P_{11}	RC (H_2)
2502	10.13	10.13	29.38	170.7	170.7	2478.4	89.14%

$\left\|\frac{\nabla_x L(x_k, \lambda_k)}{\|f(x_k)\| + 1}\right\|$	$\|c(x_k)\|_\infty$	$\|Yp_Y\|$	$\|Zp_Z\|$
1.9×10^{-6}	8.7×10^{-5}	3.8×10^{-4}	6.6×10^{-4}

Table 9. Comparison between 18 and 81 Nodes at 10 ppm Optimum

	$YP(CO)$	P_1	P_2–P_{10}	P_3–P_9	P_5–P_8	P_6	P_7	P_{11}	RC (H_2)
18 nodes	1.00×10^{-5}	2502.73	10.13	10.13	10.13	170.73	170.7	2354.8	85.88%
81 nodes	0.376×10^{-5}	2502.73	10.13	10.13	10.13	175.2	177.7	2354.8	85.89%

using the perturbation method. Because convergence of the cyclic steady state is required at each iteration, many cycle evaluations are necessary. With the simultaneous tailored approach, the optimizer updates the bed states as well as the design variables, and solves a much larger system. The sensitivities can be evaluated by either the perturbation method or the direct sensitivity approach by DASPK. The latter, with the help of ADIFOR, produces very accurate sensitivities that lead to the success of optimization, as proven by the case study in this report.

Unibed and multibed formulations are both used to describe the five-bed operation. Unibed solves only one bed and uses storage buffers to mimic the bed interactions. Implementation of the direct sensitivity approach is difficult because of the data buffers. The multibed framework simultaneously solves all beds but only for a portion of the cycle. It does not require storage buffers and the sensitivity implementation is fairly straightforward. To take advantage of efficient linear banded solvers, variable shuffling is required to minimize the computational cost. Multibed formulation truly mimics the dynamic cycle and accurately predicts both transient and cyclic steady-state behaviors, whereas the unibed framework predicts only cyclic steady-state behavior well and approximates the transient state only approximately.

Derivative calculation from the DAE model still remains the bottleneck in the optimization process. Under the multibed formulation of the hydrocarbon process, the evaluation of one sensitivity matrix usually takes 2 h on a 2.4-GHz processor and 5 h on a 1-GHz processor. We have implemented parallel computing with message passing interface to accelerate the direct sensitivity evaluation (Jiang et al., 2004) and have achieved satisfactory speed-up rates. In the future, we seek to implement the adjoint sensitivity approach. By solving the state DAEs forward and adjoint equations backward in time, this approach is more efficient than the direct approach, especially when there are many optimization variables.

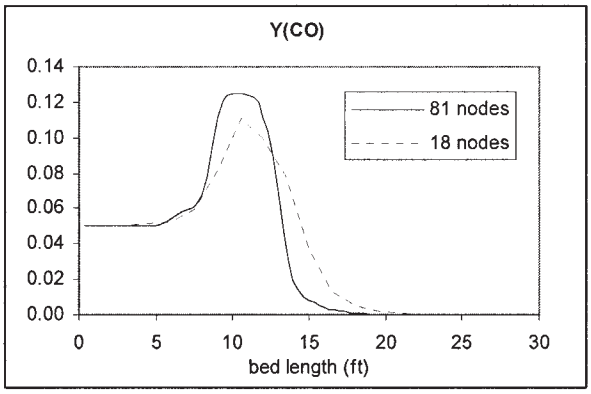

Figure 8(a) Mole fraction of carbon monoxide

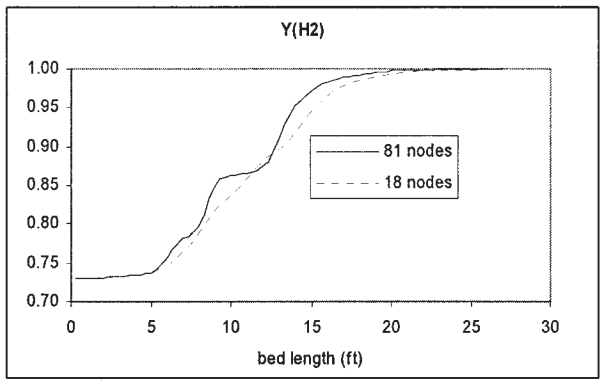

Figure 8(b) Mole fraction of hydrogen

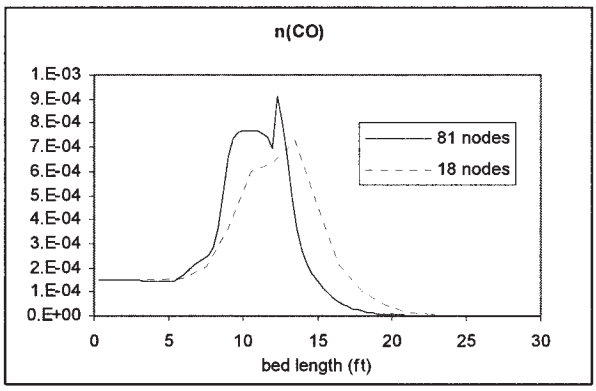

Figure 8 (c) solid loading of carbon monoxide

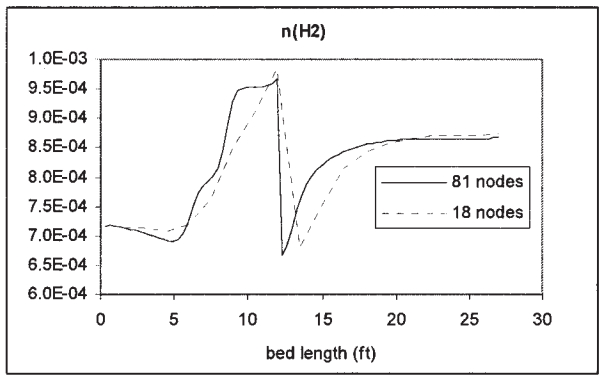

Figure 8 (d) Solid loading of hydrogen

Figure 8. (a)–(d) Comparison of bed profiles under 18 and 81 nodes at 10 ppm optimal conditions, end of feed step.

Finally, future work will include an analysis of sensitivity of the model parameters for optimization studies. Such an analysis is fairly straightforward with the rSQP algorithm and follows from the studies of Ganesh and Biegler (1987) and Wolbert et al. (1994). In particular, we note that the optimization algorithms play a very useful role in the face of uncertainty. First, parameter estimation problems that provide the best fit to experimental data, and resolve the uncertainty in model parameters, are optimization problems that require efficient algorithms like the one in this study. Second, solutions to optimization problems also illustrate the weakness of models obtained from experimental data. Instead of a static fit, they can indicate the fidelity of the model (or lack thereof) *in the desired direction of the process.*

Acknowledgments

The authors gratefully acknowledge the financial support of the NSF/GOALI program and Air Products and Chemicals Inc. The authors also thank Dr. Shengtai Li for help with DASPK package.

Notation

ρ_i = component molar density, mol/m^3
n_i = component loading on solid phase, mol/mol adsorbent
n_i^* = equilibrium loading on solid phase, mol/mol adsorbent
T = temperature, K
v = superficial velocity, m/s
P, P_i = total and component partial pressure, kPa
P_i^* = component equilibrium partial pressure, kPa
μ = gas viscosity, mol s^{-1} m^{-1}
h = mixture enthalpy, J/m^3
R = gas constant, J mol^{-1} K^{-1}
b, d = affinity parameter, 1/kPa
t = time, s
z = axial position in the adsorption bed, m

Literature Cited

Biegler, L. T., L. Jiang, and V. G. Fox, "Advances in Simulation and Optimization of Pressure Swing Adsorption Systems," *Sep and Purification Rev.*, **33**, 1 (2004).

Bischof, C., and A. Carle, *ADIFOR 2.0 User's Guide* (Revision D), Argonne National Laboratory, Information and Publishing Division, U.S. Dept. of Energy, Argonne, IL (1998).

Bischof, C., A. Carle, P. Khademi, and A. Mauer, "The ADIFOR 2.0 System for the Automatic Differentiation of Fortran 77 Programs," CRPC-TR94491 (1992).

Ganesh, N., and L. T. Biegler, "A Reduced Hessian Strategy for Sensitivity Analysis of Optimal Flowsheets," *AIChE J.*, **33**(2), 282 (1987).

Jiang, L., "Optimization of Partial Differential Equation Systems: Application to Pressure Swing Adsorption Processes," PhD Thesis, Carnegie Mellon University, Pittsburgh, PA (2004).

Jiang, L., L. T. Biegler, and V. G. Fox, "Simulation and Optimization of Pressure Swing Adsorption Systems for Air Separation," *AIChE J.*, **49**, 1140 (2003).

Jiang, L., L. T. Biegler, and V. G. Fox, "Sensitivity-Based Optimization of Pressure Swing Adsorption Systems with Parallel Implementation," Proc. of 8th Int. Symp. on Process System Engineering, Kunming, China (2004).

Kumar, R., V. G. Fox, V. G. Hartzog, R. E. Larson, Y. C. Chen, P. A. Houghton, and T. Naheiri, "A Versatile Process Simulator for Adsorptive Separations," *Chem. Eng. Sci.*, **49**, 3115 (1994).

Li, S., Personal Communication, Los Alamos National Laboratory (2003).

Li, S., and L. Petzold, "Design of New DASPK for Sensitivity Analysis," Technical Report, University of California, Santa Barbara (1999).

Li, S., L. Petzold, and W. Zhu, "Sensitivity Analysis of Differential–Algebraic Equations: A Comparison of Methods on a Special Problem," *Appl. Numer. Math.*, **32**, 161 (2000).

Malek, A., and S. Farooq, "Study of a Six-Bed Pressure Swing Adsorption Process," *AIChE J.*, **43**, 2509 (1997).

Ruthven, D., S. Farooq, and K. Knaebel, *Pressure Swing Adsorption*, VCH Publishers, New York (1994).

Sircar, S., "Pressure Swing Adsorption. Commentaries," *Ind. Eng. Chem. Res.*, **41**, 1389 (2002).

Ternet, D. J., and L. T. Biegler, "Recent Improvements to a Multiplier-Free Reduced Hessian Successive Quadratic Programming Algorithm," *Comput. Chem. Eng.*, **22**, 963 (1998).

Waldron, W. E., and S. Sircar, "Parametric Study of a Pressure Swing Adsorption Process," *Adsorption*, **6**, 179 (2000).

Weist, E. L., Air Products and Chemicals, Inc., Allentown, PA, Personal Communication (2001).

Wolbert, D., X. Joulia, B. Koehret, and L. T. Biegler, "Flowsheet Optimization and Optimal Sensitivity Analysis Using Exact Derivatives," *Comput. Chem. Eng.*, **18**(11/12), 1083 (1994).

Yang, J., and C. H. Lee, "Adsorption Dynamics of a Layered Bed PSA for H_2 Recovery from Coke Oven Gas," *AIChE J.*, **44**, 1325 (1998).

Appendix. Model Parameters (Weist, 2001)

Physical Properties for Bed Model (in the Order of N_2, CH_4, CO_2, CO, H_2)

Name	Symbol	Unit	APHP (Layer 1)	UOP5A (Layer 2)
Affinity parameter	b_0	1/kPa	3.27×10^{-6}, 7.45×10^{-7}, 6.96×10^{-7}, 1.44×10^{-6}, 1.06×10^{-6}	4.14×10^{-7}, 9.83×10^{-7}, 1.46×10^{-8}, 6.39×10^{-8}, 1.27×10^{-6}
	d_0	1/kPa	2.02×10^{-7}, 1.68×10^{-6}, 1.88×10^{-7}, 6.67×10^{-7}, 1.06×10^{-6}	4.17×10^{-7}, 9.97×10^{-7}, 7.90×10^{-8}, 3.11×10^{-7}, 1.25×10^{-6}
Heat of adsorption	q_i	J/mol	Site 1: 16107, 24124, 27101, 19381, 8420 Site 2: 16107, 13720, 22500, 13716, 8420	Site 1: 22165, 18887, 45875, 31376, 9420 Site 2: 17773, 18853, 38690, 21445, 9420
Mass transport coefficient	k_i	mol s^{-1} kPa^{-1} mol $solid^{-1}$	0.053, 0.061, 0.045, 0.053, 0.145	0.038, 0.041, 0.032, 0.038, 0.095
Saturation capacity of adsorbent	m	mol/kmol adsorbent	Site 1: 1.16×10^{-3} Site 2: 8.33×10^{-3}	Site 1: 8.18×10^{-4} Site 2: 2.23×10^{-3}
Adsorbent bed diameter	d	m	2.13	
Adsorbent bed length	l	m	3.66	4.57
Internal void fraction	ε_B	m^3 void/m^3 vessel	0.37	0.34
External void fraction	ε_T	m^3 void/m^3 vessel	0.71	0.74
Bulk density of solid	ρ_s	kg/m^3	544.64	672.79
Heat capacity of solid	C_s	J/(kg − K)	711.75	837.35
Adsorbent particle diameter	d_p	m	0.00149	0.00162
Wall heat capacity	C_{pw}	J/(kg − K)	477.98	
Wall density	ρ_w	kg/m^3	7753.05	
Wall thickness	h_w	m	0.0365	
Void volume	V_vf, V_ve	m^3	0.9622, 1.5565	
Ambient temperature	T_{amb}	K	299.82	
Heat transfer coefficient	h_{tc}	J s^{-1} m^{-2} K^{-1}	0.2839	

Gas Properties (in the Order of N_2, CH_4, CO_2, CO, H_2)

Name	Symbol	Unit	
Molecular weight	$M_{0,i}$	g/mol	28.016, 16.042, 44.010, 28.010, 2.016
Coefficients for heat capacity	$c^i_{p,A}$	J mol^{-1} K^{-1}	13109, 9021, 10096, 12773, 13202
	$c^i_{p,B}/2$	J mol^{-1} K^{-2}	−0.713, 22.8, 27.1, 0.76, −0.869
	$c^i_{p,C}/3$	J mol^{-1} K^{-3}	3.67×10^{-3}, 5.75×10^{-3}, -1.59×10^{-2}, 2.44×10^{-3}, 1.82×10^{-3}
	$c^i_{p,D}/4$	J mol^{-1} K^{-3}	-1.31×10^{-6}, -5.00×10^{-6}, 3.39×10^{-6}, -1.01×10^{-6}, -3.94×10^{-7}
Coefficients for viscosity	$\mu_{0,i}$	mol s^{-1} m^{-1}	9.86×10^{-6}, 3.86×10^{-6}, 3.05×10^{-6}, 9.38×10^{-6}, 6.42×10^{-6}
	$\mu_{1,i}$	mol s^{-1} m^{-1} K^{-1}	9.80×10^{-8}, 6.92×10^{-8}, 9.98×10^{-8}, 9.92×10^{-8}, 4.42×10^{-8}

Bed Initial Conditions

Name	Symbol	Unit	
Temperature	T	K	310.93
Pressure	P	kPa	2520.73
Mole fraction	y_i		0, 0, 0, 0, 1

Boundary Conditions

Name	Symbol	Unit	
Flow rate	Feed	mol/h	3185
	Mole_t_1	mol/h	636.4
	Mole_t_2	mol/h	1180
	a	mol/h	250
	b	mol h^{-1} s^{-1}	6.25
Valve constant	CV_1		0.3
	CV_2		0.054
	CV_3		0.075
	CV_5		0.036
Feed temperature	T_feed	K	310.93
Feed pressure	P_feed	kPa	2502.73
Feed mole fraction	Y_feed		0.005, 0.055, 0.160, 0.050, 0.730

Tank Initial Conditions (Isobaric, Isothermal)

		Tank 1	Tank 2
Mole fraction		0, 0, 0, 0, 1	0.0040, 0.1940, 0.4390, 0.089, 0.274
Pressure	kPa	2472.33	111.46
Temperature	K	310.93	310.93

Manuscript received Dec. 16, 2003, and revision received Mar. 14, 2004.