Simulation and Optimization of Pressure-Swing Adsorption Systems for Air Separation

Ling Jiang and Lorenz T. Biegler
Dept. of Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213

V. Grant Fox
Air Products and Chemicals Inc., Allentown, PA 18195

Over the past 20 years, PSA processes have gained increasing commercial acceptance as an energy-efficient separation technique. After a startup time, the system reaches cyclic steady state (CSS), at which the conditions in each bed at the start and end of each cycle are identical. Contrary to the traditional successive substitution method, we apply a direct determination approach using a Newton-based method with accurate sensitivities to achieve fast and robust convergence to CSS. Trust region methods and scaling are used to handle ill-conditioning and model nonlinearities. When design specifications are imposed, this approach is easily extended to include design constraints. This eliminates trial-and-error experiments and determines all the operating parameters simultaneously. In addition, we modify a standard flux limiter in order to deal with nondifferentiable terms and ensure computational accuracy and efficiency. Optimal PSA processes are designed by means of state-of-the-art rSQP-based optimization algorithms. The simultaneous tailored approach incorporates detailed adsorption models and specialized solution methods. Here, CSS convergence is achieved only at the optimal solution and the time-consuming CSS convergence loop is eliminated. Applications of several nonisothermal VSA O_2 bulk gas separation processes are presented to illustrate all of these approaches.

Introduction

Over the past two decades, there has been widespread development of pressure swing adsorption (PSA) systems. Their applications have expanded from drying and trace-component removal to bulk gas separations. With extensive industrial applications, there is significant interest for an efficient modeling, simulation, and optimization strategy. PSA systems are distributed in nature, with spatial and temporal variations that are mathematically represented by partial differential equations (PDEs). The conservation equations and models for the equation of state, equilibrium, and thermodynamic and transport properties are shown in Table 1.

Fixed-bed adsorption systems are typically operated in a cyclic manner with each bed repeatedly undergoing a sequence of steps such as pressurization, adsorption, pressure equalization, blowdown, desorption, and pressure equalization. After a startup time, the system approaches cyclic steady state (CSS), where the conditions at the start and end of each cycle in each bed are identical, and lead to normal production.

The CSS condition is of central importance in the design and optimization PSA processes. However, due to the intrinsic dynamics of PSA systems the convergence rate to CSS can be very slow. In real plants, it can take up to two or three days for PSA systems to reach periodic conditions. The traditional way to determine a CSS is to simulate a series of com-

Correspondence concerning this article should be addressed to L. T. Biegler.

plete cycles until the bed conditions do not change from cycle to cycle. This *successive substitution* method has the advantage of simulating the true operation of a real plant. However, from a mathematical point of view, this approach is linearly convergent and, in many cases, thousands of cycles must be run before a satisfactory cyclic condition can be reached. Thus, this approach is very time-consuming (Ritter and Yang, 1991). Webley (1998) uses a *node-refinement* procedure to accelerate the convergence for successive substitution. The procedure starts with a small number of discretization nodes and runs until CSS is reached. Then finer discretization nodes are used and an interpolation scheme is used to add these nodal values as new state variables. A new run is performed until CSS is reached. This process is repeated until the bed profiles change very little from stage to stage. Although this scheme is proven to be useful, it is still cumbersome and expensive in the case of design and optimization.

When design specifications must be met, the problem based on CSS can be expressed as

$$\begin{aligned} F(y, y', q, t) &= 0 \\ W(y(t, y_0, q)) &= 0 \\ C(y_0) = y_0 - y(t_{\text{cycle}}) &= 0 \\ LB \le (y_0, q) &\le UB \end{aligned} \tag{1}$$

Here F is the discretized-bed model from Table 1; y are the state variables; y_0 are initial conditions for the state variables; q are design variables, such as flow rates or valve constants, which are subject to the lower bounds (LB) and upper bounds (UB); W are design constraints that can include purity or pressure requirements; and C are the CSS conditions. For this design problem, it is often necessary to simulate the cycles and reach CSS several times for different operating conditions before finding the desired one (Croft and LeVan, 1994). Alternatively, PID control loops can be implemented and design variables can be adjusted frequently to converge the cycle (Webley, 1998). The disadvantage here is that performance of the controller relies heavily on the PID tuning. Problems also arise when design variables strongly interfere with each other.

To overcome the inherent slow convergence of successive substitution method, two types of direct-determination methods have been proposed. One is to discretize the PDEs in both temporal and spatial domains simultaneously using a finite difference method or a finite-element method. To obtain an accurate result, large discretization dimensions in both temporal and spatial domains are needed. This leads to a very large set of algebraic equations. These equations, together with the periodic CSS condition and design constraints, are solved by a standard nonlinear equation solver (Nilchan, 1997; Nilchan and Pantelides, 1998). The second method is a shooting method based on the method of lines. Method of lines applies spatial discretization only and converts the PDEs to a mixed set of differential algebraic equations (DAEs). The DAEs are integrated over time along with sensitivity information for these solutions. The sensitivities are then used to determine a new estimate of initial conditions using Newton-based algorithms. The new estimate is later used as the starting point for integration of a new cycle. This process repeats until the CSS condition is satisfied (Smith and Westerberg, 1992; Croft and LeVan, 1994; Ding and LeVan, 2001). Newton algorithms can achieve quadratic or superlinear convergence rates and require far fewer iterations than the successive substitution method. However, extra computation time is needed to calculate the Jacobian. To simplify the Jacobian calculation, Smith and Westerberg (1992) use Broyden's method and Kvamsdal and Hertzberg (1997) apply Muller's update formula as well as Broyden's method. However, as Ding and LeVan (2001) point out, for strongly nonlinear systems, these algorithms often fail to converge, because the Jacobian becomes poorly approximated. Ding and LeVan (2001) further propose a sensitivity interpolation scheme with dynamic grid allocation. They first calculate sensitivities only with respect to a few grids and use an interpolation scheme to generate the whole sensitivity matrix. More grid points are added and a new sensitivity matrix is calculated if the convergence rate is poor. This scheme seems to speed up the convergence rate quite dramatically. However, strong nonlinearities and ill-conditioning, which often arise from complex models, are not addressed in their article, and convergence experiments are based on heuristics. Future general application of their method is not clear.

Table 1. PSA Adsorption Model

Mass balance

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

Linear driving-force equation

$$\frac{\partial n_i}{\partial t} = \frac{k_i}{R_{PV} T}(P_i - P_i^*) \qquad i = 1, 2, \ldots, N_c$$

Energy balance

$$\left(\epsilon_B \sum_{i=1}^{N} \rho_i (c_p^i - R_E) + \rho_s c_s + \rho_s \sum_{i=1}^{N} n_i c_p^i \right) \frac{\partial T}{\partial t} - \rho_s \sum_{i=1}^{N} q_i \frac{\partial n_i}{\partial t} + \frac{\partial (v_h)}{\partial z} = 0$$

Steady-state momentum balance (ergun equation)

$$-\frac{\partial P}{\partial z} = 150 \frac{\mu v}{d_p^2} \frac{(1-\epsilon_B)^2}{\epsilon_B^3} + 1.75 \frac{\rho M v^2}{d_p} \frac{(1-\epsilon_B)}{\epsilon_B^3}$$

Single-site Langmuir adsorption isotherm

$$n_i^* = m_1 \frac{b_i P_i}{1 + \sum_{j=1}^{N} b_j P_j}, \qquad \text{where} \quad b = b_0 \exp\left(\frac{q_i}{R_E T} \right)$$

Gas in the macro pores of the adsorbent

$\text{poregas} = (\epsilon_T - \epsilon_B)\rho_i / \rho_s$ and $n_i = n_i^* + \text{poregas}$

Ideal gas law

$\rho = P/(R_{PV} T)$

Enthalpy

$$h = \sum_i \rho_i (aT + bT^2 + c_{p,c}^i T^3 + c_{p,D}^i T^4)$$

Molecular weight

$$M = \sum_i y_i M_{0,i}, \qquad \text{where mole fraction} \quad y_i = \rho_i / \rho$$

Fluid viscosity

$$\mu = \sum_i \mu_i y_i \qquad \text{where} \quad \mu_i = \mu_{0,i} + \mu_{1,i} T$$

Extending from Eq. 1, the optimization problem based on CSS of PSA systems can be expressed as

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

The terminology is the same as in Eq. 1, except that now q are decision variables for optimization. Candidates for q can be geometric parameters such as bed length, diameter, and adsorption loading or process parameters such as flow rates, step times, and operating pressures; ϕ is the objective function, and can be

- Maximize overall recovery at desired purity
- Maximize profit at desired purity
- Minimize power requirement at desired purity

Several studies have attempted to optimize the PSA systems. Most of them can be classified either as *black-box* or *equation-oriented approaches*. In the black-box approach, the optimizer selects a set of decision variables for every experiment. Based on these variables, the bed models are executed until CSS is reached. At the end of the run, the values of objective function and constraints are returned to the optimizer. For instance, Webley (1998) uses response surface methodology for this task. Based on a quadratic fit to the surface for each response variable, interactions of significant decision variables can be found. The black-box approach is robust and capable of dealing with very detailed and complicated models. It is, therefore, widely used in both industry and academia. However, since it requires CSS convergence for every single set of decision variables and hundreds of experiments are usually needed, it is extremely time-consuming.

On the other hand, Nilchan and Pantelides (1998) implement an equation-based approach using gPROMS. The optimizer, CSS condition, and bed equations are all put in the same framework and the convergence task is borne by the nonlinear equation solver within gPROMS. This approach is efficient for simple models. However, nondifferentiable terms and steep fronts from more complex bed equations can often lead to failure of the Newton solver. Besides, for complicated models, the solver may fail due to the error accumulation caused by complete discretization (Ko et al., 2003). Kvamsdal and Hertzberg (1997) use SQP and sensitivity update schemes to optimize a simple case. They report that quasi-Newton updating approaches should not be used to update sensitivities for optimization, as convergence is unstable. Instead, they suggest a thorough parameter study be performed in order to reduce the number of free variables and to find initial values close to optimum before a realistic optimization problem is solved.

In summary, most previous work on PSA optimization is based either on a robust but expensive black-box approach, or on an equation-based approach designed for simpler adsorption models. In this article we develop a new approach that overcomes some of the shortcomings of these approaches. Instead, we develop an efficient, flexible, and reliable optimization strategy that incorporates realistic, detailed process models and rigorous solution procedures. This article is organized as follows. The next section explains the overall solution strategy, including PDE discretization, CSS convergence acceleration, sensitivity calculation, and optimization. The third section presents three single-bed PSA processes as case studies. The fourth section states the conclusions and future work.

Solution Strategy

This section is divided into four parts in order to focus on the key elements of the optimization approach. We first discuss the discretization of the PSA bed equations. Next we deal with convergence acceleration of CSS problems. This is followed by a discussion of sensitivity calculations for DAEs, and finally, the implementation of a gradient-based optimization algorithm.

PDE discretization

In our work, we use the *methods of lines* to spatially discretize partial differential equations into sets of differential algebraic equations. In order to ensure the accurate approximation of the adsorption models, the selection of an appropriate discretization method is essential. Because first- and second-order finite difference or finite-element methods often introduce physically unrealistic numerical smearing or oscillation near steep adsorption fronts, it is important that we use high-resolution methods to mitigate the numerical noise (LeVeque, 1992). In addition, an accurate solution needs to be addressed with *conservative methods*, which preserve the mass and energy balance in the spatial direction. Therefore, we apply the *finite volume method*, since it is particularly suited for modeling hyperbolic conservation laws, given its inherent conservative properties (Hirsch, 1988). Webley and He (2000) use this method for their PSA simulation. With a finite volume method, the spatial domain is divided into a discrete number of volume elements and PDEs are integrated over each element, j

$$\int_{z_{j-1/2}}^{z_{j+1/2}} f(z)\, dz = \Delta_j \bar{f}_j \tag{3}$$

Here Δ_j is the spatial length of volume element j and $\bar{f}_j$ is the volume average of f in volume j. We assume $\bar{f}_j = f_j$, where f_j is the value of f at the center of volume, j; and $j \pm 1/2$ are the walls of volume j. In one dimension, the scheme is shown in Figure 1.

Thus, PDEs are converted to a set of coupled ODEs for the unknown nodal values. For example, for the mass-bal-

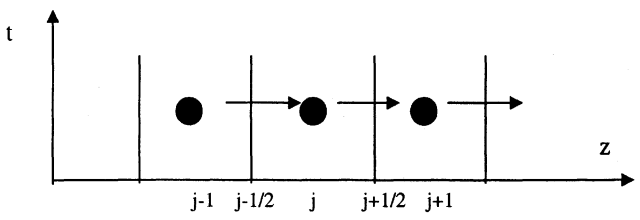


Figure 1. Finite volume method in 1-dimension.

ance equation in Table 1, the derived ODE is (for the jth node and ith component)

$$\epsilon_B \frac{d\rho_{i,j}}{dt} + \rho_s \frac{dn_{i,j}}{dt} + \frac{1}{\Delta_j}\left(\rho_{i,j+1/2}\nu_{j+1/2} - \rho_{i,j-1/2}\nu_{j-1/2}\right) = 0 \quad (4)$$

Webley and He (2000) suggest using the quadratic upstream interpolation scheme (QUICK) to interpolate cell values to wall values, where wall values are obtained from quadratic upstream interpolation of one downstream node and two upstream node values. To avoid physically unrealistic oscillations and ensure positivity of solutions, they also use an adaptive flux limiter based on the ratio of successive differences.

We verified that the QUICK flux limiter scheme is able to track the sharp front fairly well. However, since the first derivative of the flux limiter equation is not continuous everywhere, the sensitivity calculation encounters difficulty in integration. To overcome this problem we apply the Van Leer flux limiter (Hirsch, 1988)

$$\Psi(r) = \frac{r + |r|}{1 + r} \quad (5)$$

Here r is the ratio calculated by

$$r_j = \frac{\rho_j - \rho_{j-1}}{\rho_{j-1} - \rho_{j-2}} \qquad \text{if} \quad v_j \geq 0$$

and

$$r_j = \frac{\rho_{j-1} - \rho_j}{\rho_j - \rho_{j+1}} \qquad \text{if} \quad v_j < 0 \quad (6)$$

This form is shown in Figure 2(I).

Unfortunately, since the derivative near the origin is still discontinuous, this form also suffers the same numerical difficulty as QUICK. Therefore, we adopt a smoothing function (Biegler et al., 1997) and modify the interpolation scheme for wall values as

$$r' = 0.5\left(r^2 + \epsilon^2\right)^{1/2} + 0.5r$$

$$\Psi(r') = \frac{2r'}{1 + r'} \quad (7)$$

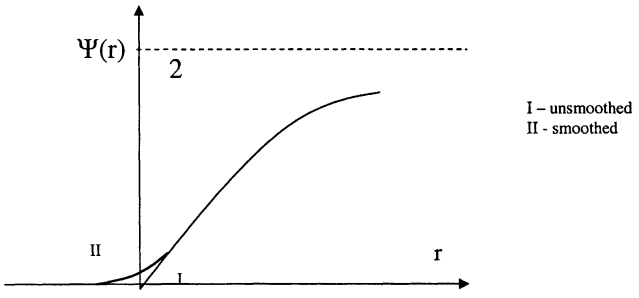


Figure 2. Van Leer's flux limiter.

This smoothed form is shown in Figure 2(II). Therefore, the overall interpolation scheme becomes

$$\rho_{j+1/2} = \rho_j + \frac{\rho_{j-1} - \rho_{j-2}}{2}\Psi(r') \qquad \text{if} \qquad v_j \geq 0$$

or

$$\rho_{j+1/2} = \rho_{j+1} + \frac{\rho_j - \rho_{j+1}}{2}\Psi(r') \qquad \text{if} \qquad v_j < 0 \quad (8)$$

Although $\Psi(r)$ is now a smooth function, we note that $\rho_{j+1/2}$ is not smooth when v_j changes the sign. This can affect the accuracy of the sensitivity calculations if we do not track the sign change exactly. However, in our work the sign change occurs at zero flow, and this zero contribution to the conservation equation does not lead to computational problems. Therefore, the results from Eq. 8 are directly used in Eq. 4. The same scheme can be used for other state variables, such as temperature. After being smoothed, the Van Leer flux limiter works well with integration of states and their sensitivities. To test the performance of different limiters on shock/contact discontinuities, we evaluated test problems from Leonard (1991). The results show that our smoothed Van Leer is more advantageous than the QUICK scheme and donor cell, with no oscillation and the least smearing at the fronts. We also compared our work with an analytical solution to an adsorption problem from Bird et al. (1960). By increasing the mass-transfer coefficient in this, we could make the adsorption front arbitrarily sharp. The smoothed Van Leer still performs the best among the three. Figure 3 shows the agreement between our work and the analytical solution for an extremely sharp front.

CSS convergence acceleration

After converting the partial differential equations into differential algebraic equations, we use a DAE solver to integrate the bed operation over a cycle. Instead of using a time-consuming successive substitution method, we apply the direct determination approach proposed by Smith and Westerberg (1991) and Croft and LeVan (1994). In direct determination approach, the periodic condition at cyclic steady state

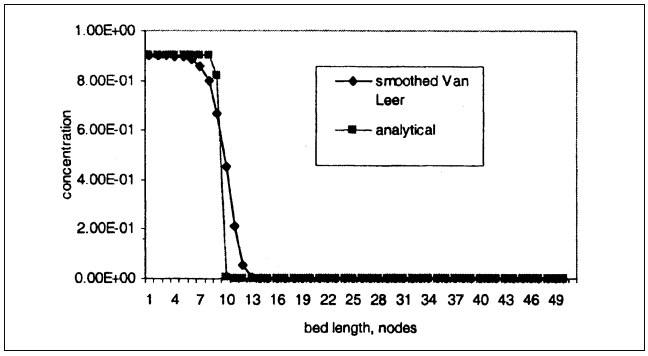


Figure 3. Test of smoothed Van Leer flux limiter using an adsorption model.

is defined mathematically as

$$e_k = C(y_{0,k}) = y(t_{\text{cycle}})_k - y_{0,k} \tag{9}$$

Here k is the iteration number, y_k and y_{0k} are final and initial conditions at the kth iteration, e_k is the system error, which must be equal to zero at CSS. With Newton's method, a direction is given by

$$p_k = -J_k^{-1} e_k \tag{10}$$

where the Jacobian is defined by

$$J_k = \frac{\partial e_k}{\partial (y_{0k})} = \left[\frac{\partial y_k}{\partial y_{0k}} - I \right] \tag{11}$$

where $\partial y_k / \partial y_{0k}$ is a matrix of sensitivities of the final condition on the initial condition at kth iteration. Therefore, the new estimate is given by

$$y_{0,k+1} = y_{0,k} + p_k \tag{12}$$

When there are design constraints, such as pressure or product purity requirements, the direct determination approach can be easily extended to find operating parameters for cycles that satisfy these constraints (Croft and LeVan, 1994). We can incorporate the design constraints into the error term and define the augmented error vector and augmented Jacobian as

$$e_k = \begin{bmatrix} C_k \\ W_k \end{bmatrix} = \begin{bmatrix} y_k - y_{0k} \\ W_k \end{bmatrix} = 0 \tag{13}$$

$$J_k = \frac{\partial e_k}{\partial (y_{0k}, q_k)} = \begin{bmatrix} \dfrac{\partial y_k}{\partial y_{0k}} - I & \dfrac{\partial y_k}{\partial q_k} \\ \dfrac{\partial W_k}{\partial y_{0k}} & \dfrac{\partial W_k}{\partial q_k} \end{bmatrix} \tag{14}$$

where W is a vector function of design constraints to be satisfied at CSS and q is the set of design variables that are chosen to satisfy the design constraints. After a new step has been determined by Eq. 10, the new states for (y_{0k}, q_k) are

$$\begin{bmatrix} y_{0,k+1} \\ q_{k+1} \end{bmatrix} = \begin{bmatrix} y_{0,k} \\ q_k \end{bmatrix} + p_k \tag{15}$$

Variations of Newton and quasi-Newton methods can also be applied here, as shown by Smith and Westerberg (1992), Croft and LeVan (1994), Ding and LeVan (2001), Kvamsdal and Hertzberg (1997), to solve relatively simple cycles and achieve a fast convergence rate. However, because adsorption models can be very sensitive to input parameters, the Jacobian matrix is likely to be poorly conditioned. In this case, the Newton step may be very poor and the method may fail. Thus, we consider a hybrid trust region method instead.

To develop this method, we note that for $\phi(x) = e^T e/2$, the steepest descent direction is defined by

$$p_k^{sd} = -\nabla \phi(x_k) = -J_k^T e_k \tag{16}$$

where J_k is defined in Eq. 11. Locally this step gives the greatest reduction in $\phi(x)$, and will give an improved point even if J_k is ill-conditioned. However, globally this step has only a linear rate of convergence, so the convergence can be very slow. Therefore, we consider an algorithm that uses a combination of Newton and steepest-descent steps and chooses a search direction between them automatically, such as Powell's dogleg method (Powell, 1970). The implementation of the dogleg method is as follows (Biegler et al., 1997). We define a Cauchy step from steepest-descent direction

$$p^c = \beta p^{sd} \qquad \text{where} \quad \beta = \| p^{sd} \|^2 / \| J p^{sd} \|^2 \tag{17}$$

where $\| \cdot \|$ is the Euclidean norm and β is determined by minimizing $\phi(x)$ along the steepest-descent direction. For a desired step length, γ, the search direction, p, can be determined by

- for $\gamma \le \| p^c \|$, $p = \gamma p^c / \| p^c \|$
- for $\gamma \ge \| p^N \|$, $p = p^N$
- for $\| p^N \| > \gamma > \| p^c \|$, $p = \eta p^N + (1 - \eta) p^c$

where

$$\eta = (\gamma - \| p^c \|) / (\| p^N \| - \| p^c \|) \tag{18}$$

For a small γ, a Cauchy step is taken, while for large γ, we take a Newton step. Otherwise, we take a linear combination of the steepest descent and Newton step. In this way, a poor Newton step due to an ill-conditioned Jacobian is avoided. The dogleg approach has been incorporated within the *Hybrd* open source code by More et al. (1980), which can be found on Netlib (www.netlib.org).

An accurate Jacobian is essential to the success of the Newton method. However, the long computation time for the Jacobian is also of great concern to the efficiency of the direct determination approach. In the *Hybrd* code, Broyden updates are used to avoid the full Jacobian evaluation, and this has proved to be cheap and efficient. The performance of the Broyden update is monitored using the system error norm. If the error continuously increases during selection of the trust region, the Broyden matrix is rejected and an exact Jacobian is calculated. To calculate the Jacobian, we implement two options: the finite difference method and the direct sensitivity method. In the finite difference method, each column of the Jacobian is generated by perturbing one element of the variable vector and integrating the DAE system over one cycle. Finite difference is easy to implement. However, in the case where the variables vary over different orders of magnitude, the accuracy of finite difference may be unsatisfactory. A more accurate direct sensitivity approach will be discussed in the following subsection.

In addition, due to nonisothermal and multilayer adsorbent effects and steep adsorption fronts, the bed model can be so nonlinear that even the trust region and dogleg step cannot guarantee robust convergence. Sometimes a Newton

search direction can point directly to the CSS solution, yet still fail to converge. We have observed this phenomenon in one PSA model. Also the Cauchy step is nearly orthogonal to the Newton step in some ill-conditioned cases and will converge extremely slowly to a cyclic steady state. Instead, by scaling the state variables properly, we obtain a better conditioned Jacobian. In our work, two methods were applied to scale the Jacobian. The first approach is by using the MC19 routine from the *Harwell Subroutine Library*. MC19 is used to calculate scaling factors automatically so that the scaled matrix has nonzero elements near unity. In the second method we examine the Jacobian elements and decide by inspection which variables need to be rescaled. For instance, if the sensitivities relating to solid loading are particularly large, we may consider a smaller scaling factor for solid loading. Both approaches achieve better-conditioned Jacobians, are proven to be effective, and have been included in our solution strategy to achieve robust convergence.

DAE solver and sensitivity evaluation

After converting the PDEs to DAEs, we apply the DAE solver DASPK 3.0 (Li and Petzold, 1999) to integrate the bed equations. DASPK solves initial-value problems for differential/algebraic equations of the form $F(t,y,y',p) = 0$ using a combination of backward differentiation formulas (BDF) and a choice of linear system solution methods. For a general DAE system

$$F_1(t, y, y', q) = 0 \tag{19}$$

DASPK solves this equation by a modified version of Newton's method

$$y^{k+1} = y^k - c\left(\alpha\frac{\partial F_1}{\partial y'} + \frac{\partial F_1}{\partial y}\right)^{-1} F_1(t, y^k, \alpha y^k + \beta, q) \tag{20}$$

Here, α, β, c are coefficients from the BDF discretization. DASPK can also perform a sensitivity analysis with respect to the parameters dy/dq_i and $dy/dy_{0,i}$ simultaneously with solution of the DAE system. Other DAE solvers, such as DDASAC (Stewart et al., 1995) and ODESSA (Leis and Kramer, 1988) can perform the same task, but DASPK 3.0 adopts a more advanced sensitivity calculation method. Sensitivity analysis entails finding the derivative of each variable y with respect to each parameter. This produces additional $N_q \times N_y$ sensitivity equations, which yield

$$F_1(t, y, y', q) = 0$$

$$\frac{\partial F_1}{\partial y}s_i + \frac{\partial F_1}{\partial y'}s_i' + \frac{\partial F_1}{\partial q_i} = 0, \qquad i = 1, \ldots, N_q$$

where

$$s_i = \partial y/\partial q_i \tag{21}$$

If we define

$$y = \begin{bmatrix} y \\ s_1 \\ \cdots \\ s_{Nq} \end{bmatrix}, \qquad F = \begin{bmatrix} F_1(t, y, y', q) \\ \dfrac{\partial F_1}{\partial y}s_1 + \dfrac{\partial F_1}{\partial y'}s_1' + \dfrac{\partial F_1}{\partial q_i} \\ \cdots \\ \dfrac{\partial F_1}{\partial y}s_{Nq} + \dfrac{\partial F_1}{\partial y'}s_{Nq}' + \dfrac{\partial F_1}{\partial q_i} \end{bmatrix} \tag{22}$$

the system can be rewritten as $F(t, Y, Y', q) = 0$ and can be solved by Newton's iteration $Y^{k+1} = Y^k - J_F^{-1}F(Y^k)$, where J_F represents the Jacobian of F. After solving for sensitivities, s_i, we load the sensitivities into a Jacobian, which we use later to determine new states.

Sensitivity methods can be classified into the *staggered direct* (Caracotsios and Stewart, 1985), *simultaneous corrector* (Maly and Petzold, 1996), and *staggered corrector* options (Feehery et al., 1997). At each time step, the simultaneous corrector method solves the entire dynamic system of state and sensitivity Eqs. 21. Staggered methods first solve the differential equations for the state variables at each time step. After the Newton method has converged for the state variables, sensitivity equations are solved at the current step. In solving the sensitivity equations, the staggered direct method updates the Jacobian factorization $\alpha(\partial F_1/\partial y')+(\partial F_1/\partial y)$ at every step, while the staggered corrector, the most recent method, refactorizes this Jacobian matrix only when necessary. Since this is often the most expensive step in the sensitivity algorithm, considerable savings are made. For more details on comparison of these three methods, see Li and Petzold (2000).

The $\alpha(\partial F/\partial y')+(\partial F/\partial y)$ matrix and sensitivity residuals are determined either by a finite difference approximation or through automatic differentiation. When automatic differentiation tools are used to evaluate $\alpha(\partial F/\partial y')+(\partial F/\partial y)$ and sensitivity residuals, the accuracy and computation efficiency can be further enhanced. Automatic differentiation techniques apply the chain rule repeatedly to the function subroutine to compute derivatives that are correct up to machine precision. For instance, the code ADIFOR 2.0 (Bischof et al., 1994) can be implemented with DASPK 3.0 to generate accurate sensitivities. In our studies we compared both the automatic differentiation and finite difference options. For a well-scaled problem, it appears that the finite difference option was reasonably accurate. However, automatic differentiation may be faster as a result of more accurate sensitivities and fewer Newton solver failures.

Low-Order Parameterization for CSS. Nevertheless, a disadvantage of the direct sensitivity approach is that the computational cost of sensitivities increases linearly with N_q and N_y; this will be expensive when N_q is large. Therefore, we also consider low-order profile parameterizations to reduce this cost. Many adsorption-bed profiles have some simple shapes that can be approximated by polynomials, splines, Fourier transformation, or eigenfunction expansions. As a first step, we develop a low-order profile representation where cyclic steady-state profiles are approximated by Lagrange interpolation polynomials. We choose interpolation points and Lagrange polynomials to approximate the state variables spa-

tially over the whole bed

$$y(z,t) = \sum_j y_j l_j(z)$$

where

$$l_j(z) = \frac{\prod_{i \neq j} (z - z_i)}{\prod_{i \neq j} (z_j - z_i)} \tag{23}$$

Here, y is any state variable, y_j are the chosen interpolation points, $l(z)$ is the Lagrange polynomial, and z_j is the location of the point j. Thus, instead of solving $y(t_{\text{cycle}}) = y_0$ for all state variables, the convergence criterion shrinks to

$$y_j(t_{\text{cycle}}) = y_{0,j} \tag{24}$$

Since y_j is only a small subset of the state variables y, the number of sensitivity parameter N_q is significantly reduced and sensitivity calculation is much cheaper. However, this simplification may introduce inaccuracy into design and optimization results (Biegler et al., 1985). As pointed out, the use of simplified models can lead to false optima. This will be seen further in the case study below.

PSA optimization

As mentioned earlier, the black-box approach is inefficient for optimization and the equation-based approach cannot handle detailed models well. To overcome these deficiencies, we develop a *simultaneous tailored approach*. Here convergence of cyclic steady state is incorporated as a constraint in the optimization problem. At each iteration, the detailed DAE bed model is solved in an inner loop, in order to obtain values of the constraints and objective function. Also, accurate gradients with respect to parameters are obtained through sensitivity calculations. Since cyclic steady state is converged only at the optimum, the time-consuming CSS convergence loop is eliminated and the tailored approach is much faster than the black-box approach. Compared with the equation-based approach, the tailored approach is robust and able to deal with complicated models. Nondifferentiability caused by flux limiters is handled by using smoothing functions and initialization is fairly easy. Figure 4 shows the structures of three optimization algorithms.

We adopt the reduced-space successive quadratic programming (rSQP) (Boggs and Tolle, 2000; Ternet and Biegler, 1998) as the optimization algorithm in the simultaneous tailored approach. Since the number of design variables is relatively small (< 20), rSQP is designed to optimize large nonlinear programming systems of the form Eq. 25 and is well suited because it exploits the problem structure.

$$\begin{aligned} &\min_x f(x) \\ &s.t. c(x) = 0 \\ &x^L \leq x \leq x^U \end{aligned} \tag{25}$$

Here, $f(x)$ is the objective function and $c(x)$ is a set of constraints. SQP is derived from a quadratic approximation of the Lagrange function and by linearizing the constraints. At an iterate, x_k, a search direction, d_k, is generated by solving

$$\begin{aligned} &\min_d g(x_k)^T d + \frac{1}{2} d^T B(x_k) d \\ &c(x_k) + A(x_k)^T d = 0 \\ &x^L \leq x_k + d \leq x^U \end{aligned} \tag{26}$$

where $g(x)$ denotes the gradient of $f(x)$; $B(x)$ denotes the Hessian of the Lagrange function $L(x, \lambda) = f(x) + \lambda^T c(x)$, and $A(x)$ denotes the $n \times m$ matrix of constraint gradients $A(x_k) = [\nabla c_1(x_k), \ldots, \nabla c_m(x_k)] = [J \; Q]$. The variables are partitioned into independent $(n - m)$ and dependent (m) variables, and the search direction is defined as $d_k = Y_k p_Y + Z_k p_Z$, $p_Y \in R^m$, $p_Z \in R^{n-m}$. Here p_Y updates the depen-

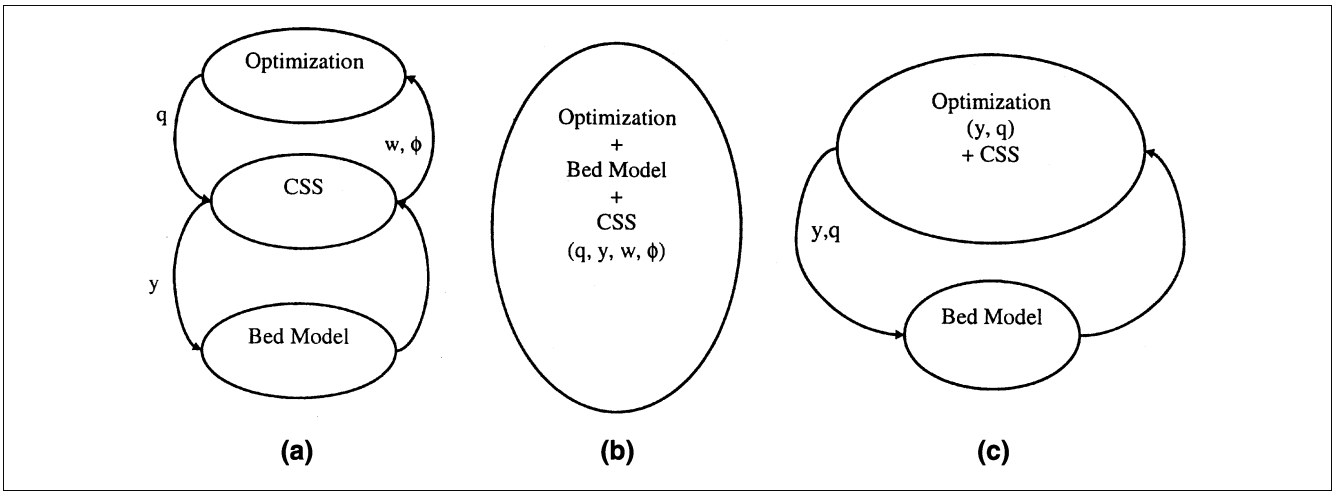


Figure 4. (a) Black-box approach; (b) equation-oriented approach; (c) simultaneous tailored approach.

dent variables, which serve to improve the solution of the equality constraints, while p_Z updates independent variables, which act in the null space of the constraints and serve to minimize the objective function; Y_k and Z_k are two basis matrices with $A_k^T Z_k = 0$ and $Y_k \in R^{n\times m}$, $Z_k \in R^{n\times(n-m)}$. Here $Y_k = \begin{bmatrix} 0 \\ I \end{bmatrix}$, $Z_k = \begin{bmatrix} I \\ -J^{-1}Q \end{bmatrix}$. The direction p_Y usually comes from a Newton step

$$(Yp_Y)_N = \begin{bmatrix} 0 \\ I \end{bmatrix} \qquad p_y = \begin{bmatrix} 0 \\ -J^{-1}c_k \end{bmatrix} \tag{27}$$

and the p_Z step is computed by solving a smaller QP subproblem in the space of independent variables (Ternet and Biegler, 1998)

$$\min_d g(x_k)^T Zp_Z + \frac{1}{2} p_Z^T B(x_k) p_Z$$
$$x^L \le x_k + Zp_Z + Yp_Y \le x^U \tag{28}$$

As mentioned earlier, the Jacobian $J(x)$ that includes the CSS equations may be ill-conditioned. To ensure robust convergence and prevent bad Newton steps, we incorporate a dogleg step for the calculation of p_Y. With this implementation the p_Y step is transformed to a hybrid step, between a Newton direction and a steepest-descent direction. Here the steepest Cauchy step is defined by

$$(Yp_Y)_c = -\beta(Ac_k)$$

where

$$\beta = \frac{\|(Ac_k)\|^2}{\|A^T(Ac_k)\|^2} \tag{29}$$

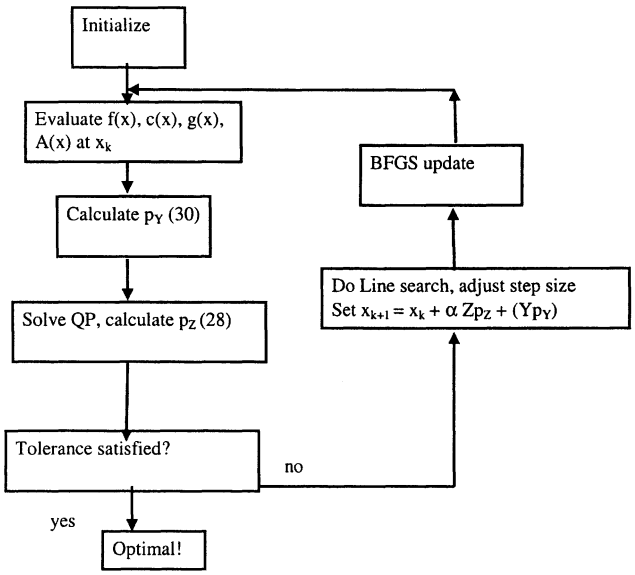


Figure 5. Flow sheet for rSQP Optimization Algorithm.

In addition, the rSQP algorithm incorporates an Armijo line-search strategy to promote global convergence behavior. The stepsize α_k is chosen so that $x_{k+1} = x_k + \alpha_k d_k$ is found that reduces an exact penalty function $\varphi = f(x) + \mu_k \| c(x) \|_2$, where $\| c \|_2$ is the Euclidean norm of constraints. The dogleg algorithm with line search can be derived from Eq. 18 by substituting $\alpha \| (Yp_Y)_N \|$ for γ, to yield

if

$$\alpha = 1,$$

then

$$(Yp_Y) = (Yp_Y)_N$$

if

$$\alpha \le \alpha_c,$$

then

$$(Yp_Y) = \frac{\alpha}{\alpha_c}(Yp_Y)_c$$

where

$$\alpha_c = \frac{\|(Yp_Y)_c\|}{\|(Yp_Y)_N\|}$$

else

$$(Yp_Y) = \eta(Yp_Y)_N + (1-\eta)(Yp_Y)_c$$

where

$$\eta = \frac{\alpha - \alpha_c}{1 - \alpha_c} \tag{30}$$

Here μ is a penalty parameter defined to include the hybrid features

$$\mu \ge \max\left\{ \frac{\upsilon + g^T(Yp_Y)_N|}{\| c \|_2}, \| c \|_2 \beta \frac{|\upsilon + g^T(Yp_Y)_c/\alpha_c|}{|(Yp_Y)_c^T(Yp_Y)_c/\alpha_c|} \right\} \tag{31}$$

where υ is the vector of bound multipliers. To validate the modification with the dogleg step, we ran several test problems and found that the dogleg modification performs at least

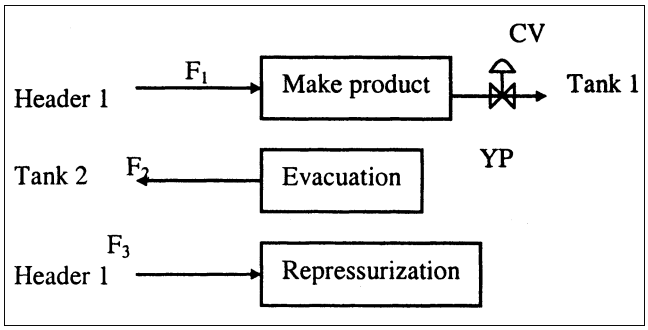


Figure 6. Single-bed, 3-step PSA cycle.

The boundary conditions are:

Step k	Beginning of bed	End of bed
Step 1	Constant flow	Determined by valve equation
Step 2	Constant flow (reverse direction)	Velocity is zero
Step 3	Constant flow	Velocity is zero

And the valve equation is given by:

$$\text{if } P_{\text{tank}} > P_{end\_of\_bed} \text{ then } v = 0$$

$$\text{else, calculate critical pressure ratio } Pcrit = \left(\frac{2}{1+\gamma}\right)^{\frac{\gamma}{1-\gamma}}$$

$$\text{if } P_{end\_of\_bed} / P_{\text{tank}} > Pcrit \text{ then flow is choked, } v = CV * P_{end\_of\_bed} * \sqrt{\left|\frac{1-\left(\frac{1}{Pcrit}\right)^2}{M \cdot T}\right|}$$

$$\text{else, } v = CV * \sqrt{\left|\frac{P_{\text{tank}}^2 - P_{end\_of\_bed}^2}{M \cdot T}\right|}$$

Figure 7. Boundary conditions for System 1.

as well as the Newton step. In cases where the constraints are particularly hard to converge, the dogleg performs much better and takes far fewer iterations than with Newton steps.

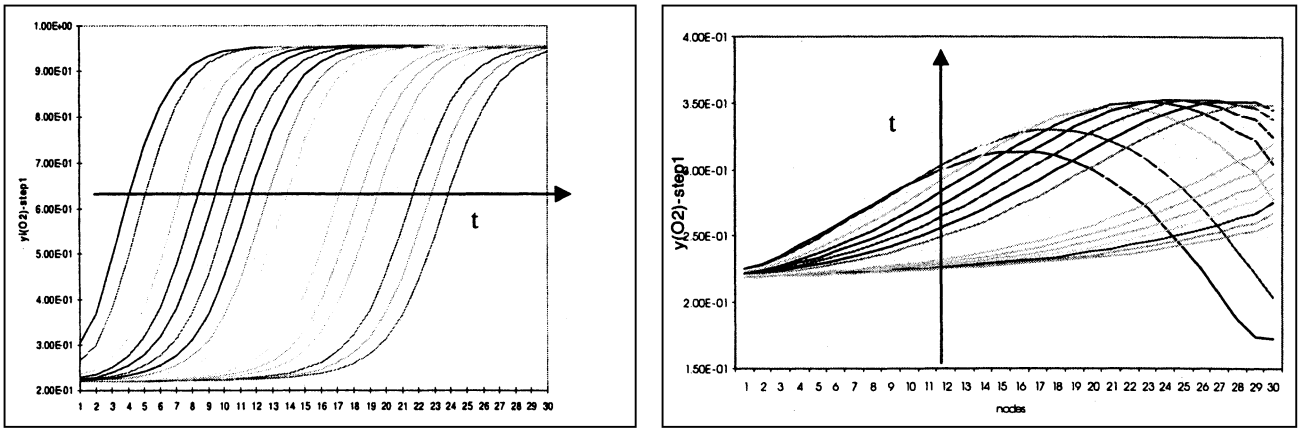
For termination criteria, as pointed out by Nash and Sofer (1996), the scale of $\nabla_x L(x_k, \lambda_k)$ depends on the objective function. Therefore, our convergence test is given by

$$\max\left(\left\|\frac{\nabla_x L(x_k,\lambda_k)}{|f(x_k)|+1}\right\|_\infty, \|c(x_k)\|_\infty\right) < \epsilon \qquad (32)$$

The tolerance ϵ is selected according to the accuracy of function evaluations. If the tolerance for the differential equation integration, that is, the function evaluation, ϵ_I (such as 10^{-6}), then x_k and $\nabla f(x_k)$ can only expect a precision of $\epsilon = \sqrt{\epsilon_I} = 10^{-3}$. Gill et al. (1989) point out that $\epsilon = \sqrt{\epsilon_I}$ is overly stringent and suggest $\epsilon = \sqrt[3]{\epsilon_I} = 10^{-2}$ according to their experience. The basic algorithm for rSQP is summarized in Figure 5.

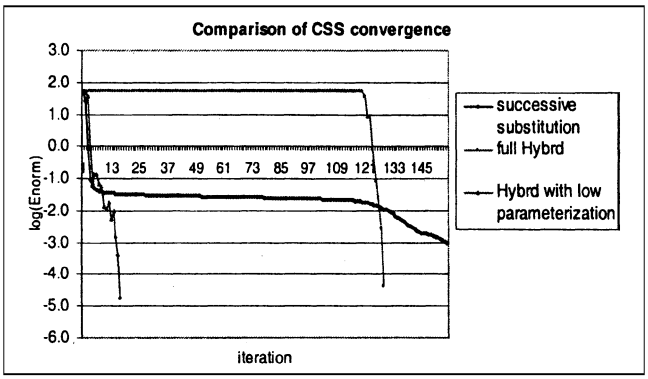
Case Study and Computational Results

Applications of a single-bed 3-step O_2 VSA cycle and two single-bed 6-step O_2 VSA industrial cycles are used to demonstrate the CSS and optimization algorithms. These processes are used to separate O_2 from air using a zeolite molecular sieve.

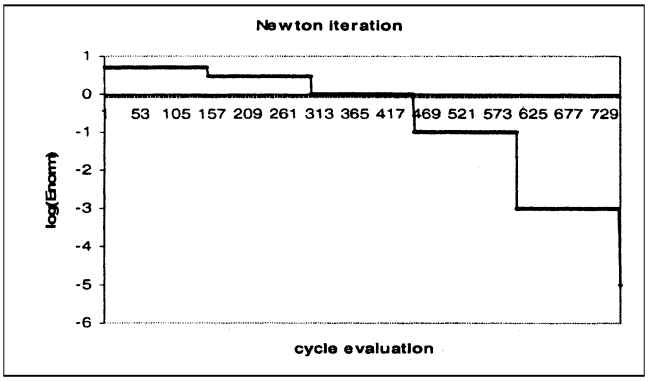


(a) (b)

Figure 8. O_2 mole fraction profile (a) after the first cycle; (b) at CSS, system 1.



(a)



(b)

Figure 9. (a) Comparison of CSS convergence speed with different methods, system 1; (b) CSS convergence with Newton method, system 1.

System 1

Figure 6 shows the flow sheet for a single-bed 3-step nonisothermal VSA cycle. The cycle is operated in three steps: (1) make product (adsorption), (2) evacuation, and (3) repressurization. The duration time for each step is 19, 15, and 5 s, respectively. The bed models are shown in Table 1 with boundary conditions shown in Figure 7.

The adsorption bed is discretized into 30 nodes using a finite volume method. Each node has five differential variables, which are densities and loadings for two components and one temperature variable. The number of state variables for the DAE model is 150. The equilibrium partial pressure is determined from a Langmuir isotherm using a Newton solver in an inner loop. The DAE integration was performed using DASPK 3.0, with the banded structure option (bandwidth = 3 nodes of states), and the finite difference option to generate $\alpha(\partial F/\partial y') + (\partial F/\partial Y)$. A staggered-corrector method is used to calculate the sensitivities, with central differencing to compute the sensitivity equations using 10^{-3} as the perturbation factor.

After simulating the bed performance, we get the bed profiles. Figure 8 shows the O_2 mole fraction profiles after the first cycle and at cyclic steady state. These profiles have been validated using the in-house simulator at APCI (Kumar et al., 1994).

Figure 9(a) compares the CSS convergence speed among successive substitution methods, a full direct-determination method with dogleg implementation, and a parameterized, low-order direct-determination method, with respect to an equivalent cycle evaluation based on CPU time. It takes 156 successive substitution cycles to reach an error norm of 9.86 $\times 10^{-5}$ and one Jacobian evaluation with several Broyden updates to reach 5.46×10^{-5} with *Hybrd*. Also, a second-degree Lagrange polynomial is used to parameterize the bed profiles at CSS. The number of interpolation points is three, and we choose the first, fifteenth, and thirtieth nodes as the interpolation points. When low-order parameterization is used, the convergence is accelerated dramatically and computation time is reduced by 90%. On the other hand, the pure Newton method with accurate sensitivities takes five full Jacobian evaluations to converge to an error norm of 5.22×10^{-5}. As seen by the performance plot in Figure 9b, it is also clear that the competitiveness of a direct-determination approach depends on evaluating an accurate Jacobian quickly.

Next we consider the design problem (Eq. 1). Table 2 lists the selected design variables and constraints. The pressures are measured at the end of the bed at the end of each step. O_2 purity is calculated at the bed outlet according to

$$YP - \frac{\int_{\text{step1}} (\upsilon \rho y(O_2))\, dt}{\int_{\text{step1}} \upsilon \rho dt} = 0 \qquad (33)$$

In order to evaluate the integral, we add two additional differential equations and two new state variables into the DAE system.

$$\frac{d(F \cdot y)}{dt} = \upsilon \rho y(O_2)|_{\text{end\_of\_bed}} \quad \text{and}$$

$$\frac{dF}{dt} = \upsilon \rho|_{\text{end\_of\_bed}}$$

Table 2. Design Results for System 1

	Base Value	3-Pt Interpolation	Full Order
Feed stream (F_1, m^3/s)	11.80	10.35	9.77
Evacuation stream (F_2, m^3/s)	11.80	12.55	12.49
Purge stream (F_3, m^3/s)	11.80	13.45	13.46
Valve constant (CV)	5.086	2.748	2.511
End of step 1 pressure (P_1, kPa)	152.26	172.25	172.25
End of step 2 pressure (P_2, kPa)	49.538	50.66	50.66
End of step 3 pressure (P_3, kPa)	106.77	116.52	116.52
O_2 purity (YP, %)	30.55	35.00	35.00

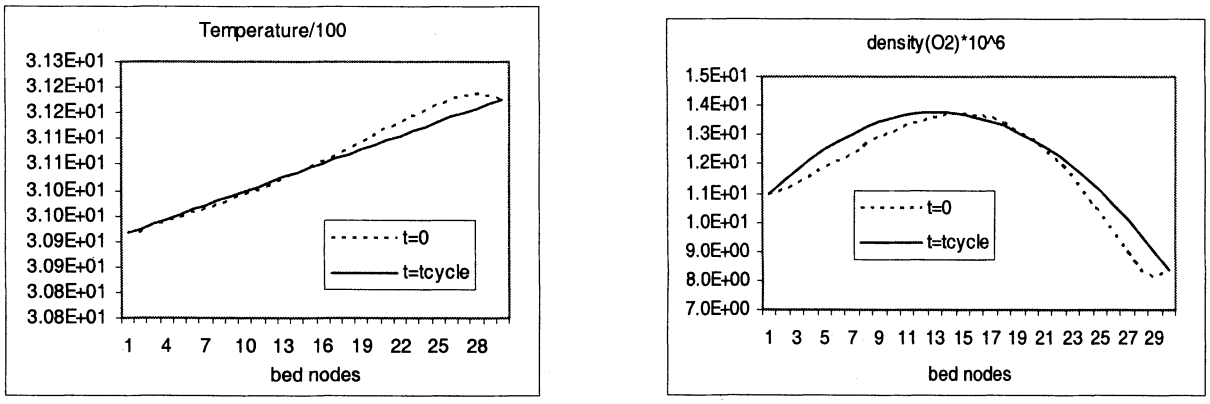


Figure 10. (a) Temperature profile; (b) O_2 density profile for 3-pt interpolation under design conditions, system 1.

Table 3. Optimization Results for System 1

	(a) Decision Variables					
	CV	*P*_tank (kPa)	T_2 (s)	T_3 (s)	*YP* (%)	*RC* (%)
Base value	5.086	113.89	15.0	5.0	30.55	71.94
Design	2.511	113.89	15.0	5.0	35.00	62.24
7-pt optimal	4.249	132.63	100.0	91.15	35.00	96.23
Full optimal (FD)	2.908	30.90	100.0	98.28	35.00	89.83
Full optimal (AC)	2.908	205.69	100.0	98.26	35.00	89.83
	(b) Bed Pressures at Each Step					
	P_1 (kPa)	P_2 (kPa)	P_3 (kPa)			
7-pt interpolation	250.78	11.16	1,641.46			
Full order (FD)	389.90	10.38	2,029.54			
Full order (AC)	389.70	10.37	2,028.53			

Table 4. Convergence Errors [see (Eq. 32)] at the Optimal Solution for System 1

	$\left\|\frac{\nabla_x L(x_k, \lambda_k)}{\lvert f(x_k)\rvert+1}\right\|$	$\lVert c(x_k)\rVert_\infty$	Iter. No.	CPU h
7-Pt interpolation	2.0×10^{-5}	9.5×10^{-5}	81	7.63
Full order (*FD*)	2.1×10^{-6}	1.4×10^{-4}	52	16.13
Full order (*AC*)	7.7×10^{-6}	1.8×10^{-4}	55	40.77

during step 1, and

$$\frac{d(F\cdot y)}{dt} = \frac{dF}{dt} = 0 \tag{34}$$

during steps 2 and 3.

The design targets are $P_1 = 172.25$ kPa, $P_2 = 50.66$ kPa, $P_3 = 116.52$ kPa, and $YP = 35\%$. Applying the direct-determina-

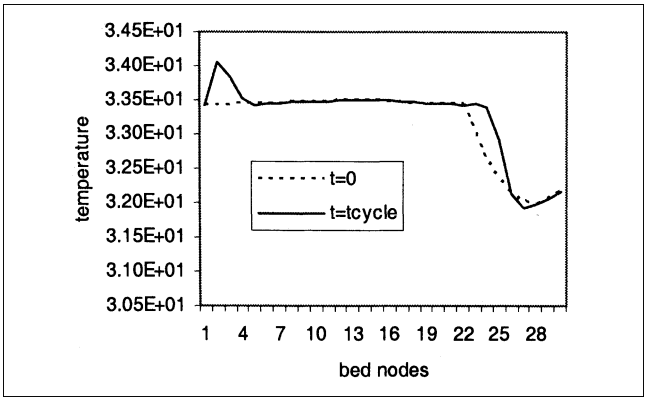


Figure 11. Temperature under optimal conditions for system 1 (7-pt interpolation).

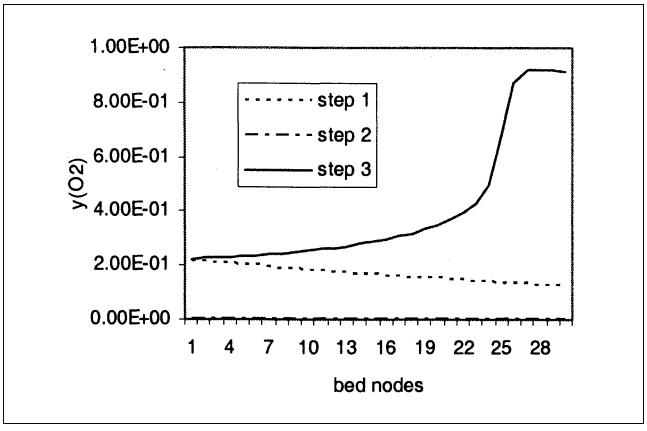


Figure 12. O_2 mole fraction profile under optimal conditions for system 1 (full order).

Table 5. Design Results for System 2

	Base	Design
Feed stream for step 1, 6 (F_1, m^3/s)	4.73	4.63
Evacuation stream for step 3&4 (F_2, m^3/s)	6.90	6.78
Valve constant on step 4 (CV_2)	1.336	0.736
Valve constant on step 6 (CV_4)	1.673	1.485
Valve constant ($CV\__t$)	0.139	0.125
End of step 1, pressure (P_1, kPa)	149.35	151.99
End of step 3, pressure (P_3, kPa)	48.74	50.66
End of step 4, pressure (P_4, kPa)	71.03	60.80
End of step 6, pressure (P_6, kPa)	101.63	101.33
Product purity (YP, %)	85.52	95.00

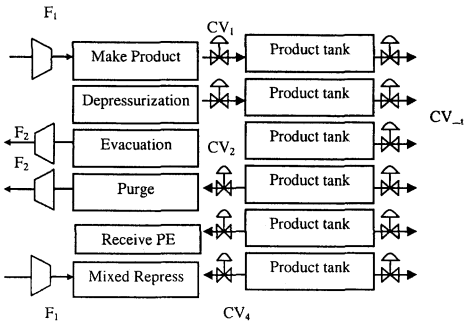


Figure 13. Six-step industrial PSA cycle.

tion approach to the problem (Eq. 1), we achieve the results listed in Table 2. Compared to the base case, less gas is used in (F_1), but more is used for (F_3). This leads to more gas withdrawn during the vacuum step (F_2) and less product retrieved (CV), but with higher product purity.

To compare with a full discretization, we also use a second-degree Lagrange interpolation polynomial to parametrize the bed profiles at CSS. Both the low-order approximation and full-order method succeed in satisfying the design con-

Dual-site Langmuir Adsorption Isotherm

$$n_i^* = m_1 \frac{b_i P_i^*}{1 + \sum_{j=1}^{N} b_j P_j^*} + m_2 \frac{d_i P_i^*}{1 + \sum_{j=1}^{N} d_j P_j^*} \quad \text{where } b = b_0 \exp(\frac{q_i}{R_E T}) \text{ and } d = d_0 \exp(\frac{q_i}{R_E T})$$

Product tank

Mass balance: $V_{tank} \frac{\partial \rho_i}{\partial t} = area * (\rho_i v - \rho_{tank} v_1)$ during step 1 and 2

$V_{tank} \frac{\partial \rho_i}{\partial t} = -area * \rho_{tank} v_1$ during step 3

$V_{tank} \frac{\partial \rho_i}{\partial t} = -area * \rho_{tank} (v_1 + v_2)$ during step 4, 5 and 6

Energy balance: $V_{tank} \frac{\partial u}{\partial t} = area * (hv - h_{tank} v_1)$ during step 1 and 2

$V_{tank} \frac{\partial u}{\partial t} = -area * h_{tank} v_1$ during step 3

$V_{tank} \frac{\partial u}{\partial t} = -area * h_{tank} (v_1 + v_2)$ during step 4, 5 and 6

The bed boundary conditions are:

Step k	Beginning of bed	End of bed
Step 1	Constant flow (F_1)	Determined by valve equation (CV_1)
Step 2	Velocity is zero	Determined by valve equation (CV_1)
Step 3	Constant flow (reverse, F_1)	Velocity is zero
Step 4	Constant flow (reverse, F_2)	Determined by valve equation (CV_2)
Step 5	Velocity is zero	Determined by valve equation (CV_3)
Step 6	Constant flow (F_1)	Determined by valve equation (CV_4)

Figure 14. Additional model equations for system 2 and 3.

straints. For the low-order method, much less computation time is required than for the full-order method, and the accuracy is reasonably good. Figures 10a and 10b show the bed profiles under design conditions given by the low-order method. The variables match only at the interpolation points at the beginning and the end of the cycle.

Finally, we consider the optimization problem (Eq. 2). We choose to maximize O_2 recovery at desired purity (35%) at cyclic steady state. Recovery is calculated by

$$RC = \frac{\int_{\text{step1}} (v\rho y(O_2))\, dt}{\int_{\text{step1\&3}} (v\rho y(O_2))\, dt} \tag{35}$$

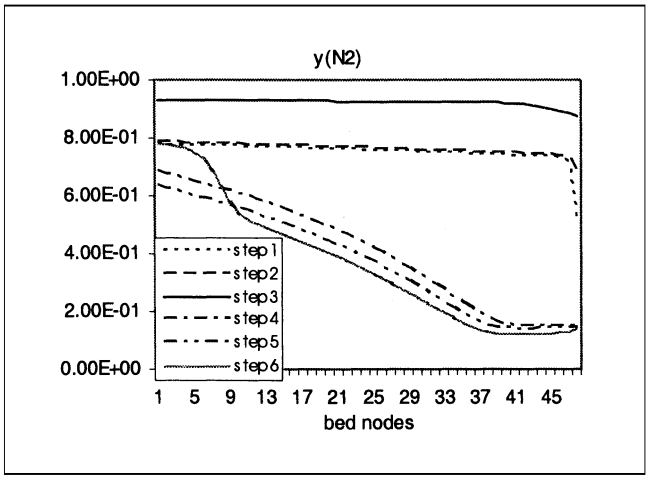
As decision variables, we choose product tank pressure (P_{tank}), the valve constant (CV), and step times for steps 2 and 3 (T_2, T_3). Also in our comparison, we use a 7-point Lagrange polynomial with interpolation points at 1, 7, 15, 18, 22, 26, 30. The upper bounds for T_2 and T_3 are 100 s. Table 3 shows the optimal results. We also compare the results given by finite difference derivatives (FD) and accurate sensitivities generated by DASPK (AC); they are very close, although AC requires twice the CPU time. We also note that low-order approximation leads to inaccurate optimization results. As seen in Figure 11, a higher recovery is achieved from low parameterization at the cost of not satisfying the cyclic steady-state condition. Therefore, caution needs to be taken if low-order methods are to be used for optimization. Table 4 lists the convergence errors (defined by Eq. 32) at the optimal point, while Figure 12 shows the O_2 mole fraction profile under optimal conditions.

Note that at the optimal point, the purity and recovery are insensitive to the tank pressure, P_{tank}. According to the valve equation, any P_{tank} less than P_1 (389.70 kPa) does not affect purity and recovery. Compared with the benchmarks, the optima achieves 18% more recovery than the base conditions and 27% more recovery than the design conditions while satisfying the purity constraints. In order to achieve the maximum recovery possible while maintaining the purity, we now withdraw the product and repressurize at a high level, and both T_2 and T_3 almost hit their upper bounds. This allows more opportunity for adsorption and regeneration of O_2 during the cycle.

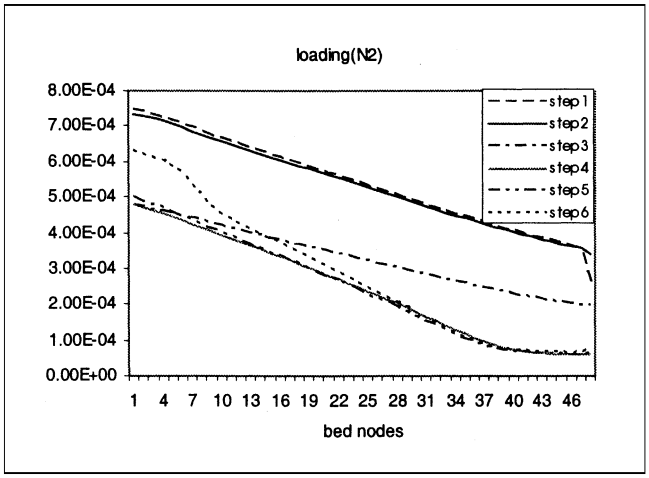
System 2

This is a single-bed 6-step industrial O_2 VSA process. The adsorption bed is continuously packed with one adsorbent. The flow sheet is shown in Figure 13. The duration for each step is 25, 5, 15, 10, 3, 3 s, respectively. A nonisothermal, nonisobaric product tank is included in the flow sheet. The bed models are similar to Table 1, with some additional equations shown in Figure 14.

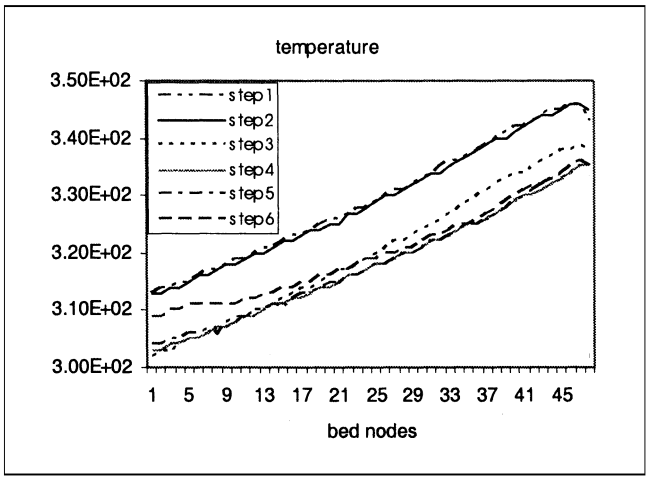
The adsorption bed is discretized with 48 nodes. At each node, there are two algebraic variables, which are the equilibrium partial pressure for two components, and five differential variables, which are the gas density and loading for two components and temperature. There are three additional differential variables, the gas densities and temperature in the



(a)



(b)



(c)

Figure 15. (a) N_2 mole fraction; (b) N_2 loading; and (c) temperature profiles at cyclic steady state for system 2.

Table 6. Optimization Results for System 2

	F_1 (m^3/s)	F_2 (m^3/s)	CV_2	CV_t	T_1 (s)	T_4 (s)	P_1 (kPa)	P_3 (kPa)	P_4 (kPa)	YP, %	Work, MJ/kmol
Design	4.63	6.78	0.736	0.125	25	10	101.99	50.66	60.80	95.00	15.59
Opt (FD)	4.47	6.90	0.733	0.125	24.9	8.19	101.99	51.07	61.20	95.00	14.83
Opt (AC)	3.93	6.90	0.775	0.110	25.6	4.97	101.99	51.27	61.40	95.00	14.35
Upper bound	0	0	3.0	3.0	100	100	N/A	N/A	N/A	N/A	N/A
Lower bound	0.1	0.1	0.3	0.0003	1	1	N/A	N/A	N/A	N/A	N/A

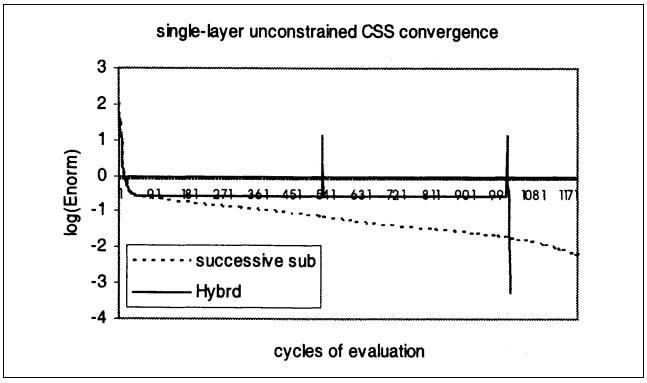


Figure 16. Comparison of convergence speed of successive substitution and direct determination for system 2.

product tank. DASPK is used as a DAE solver to solve this system of 339 state equations. ADIFOR is used here to evaluate $\alpha(\partial F/\partial y')+(\partial F/\partial y)$ and the sensitivity equations. The SparsLinC option is used to generate the sparse Jacobian, the matrix-vector option is used to generate the sensitivity residual, and a staggered-corrector method is used.

The cyclic steady-state profiles are shown in Figure 15, and a comparison of successive substitution with direct determination is shown in Figure 16. It takes 1,500 successive substitution cycles to converge, while direct determination requires only two Jacobian evaluations and several Broyden updates. In Figure 16, we observe some spikes after each Jacobian reevaluation. This occurs as the trust region size adjusts itself and stabilizes the convergence. We also consider a pure Newton method, but this takes extremely small steps toward the solution.

The design targets are $P_1 = 151.99$ kPa, $P_3 = 50.66$ kPa, $P_4 = 60.80$ kPa, $P_6 = 101.33$ kPa, $YP = 95\%$ and O_2 purity is calculated at the tank by

$$YP = \frac{\int_{\text{step1}} (\upsilon\rho y(O_2))\, dt|_{\text{tank}}}{\int_{\text{step1}} \upsilon\rho\, dt|_{\text{tank}}} \tag{36}$$

Table 5 shows design results. Again, direct determination satisfies the design constraints.

Table 7. Convergence Errors (Eq. 32) at the Optimum for System 2

	$\left\|\frac{\nabla_x L(x_k, \lambda_k)}{\lvert f(x_k)\rvert+1}\right\|$	$\|c(x_k)\|_\infty$	Iter. No.	CPU h
Opt (FD)	4.6×10^{-2}	7.4×10^{-5}	57	42.27
Opt (AC)	3.1×10^{-4}	4.9×10^{-5}	34	38.03

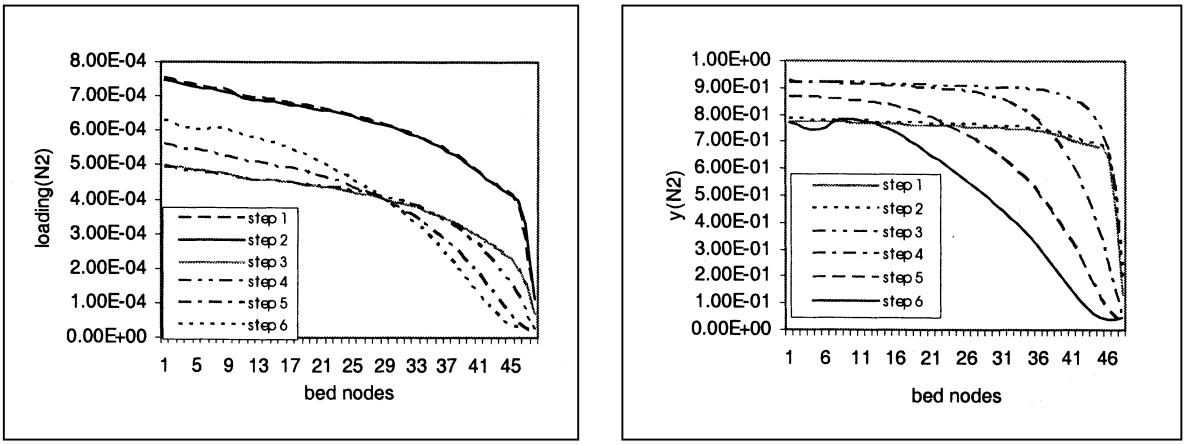


Figure 17. (a) N_2 loading; (b) N_2 mole fraction. Bed profiles under optimal condition for system 2.

Table 8. Design Results for System 3

	F_1	F_2	CV_2	CV_4	$CV_{\_t}$	P_1 (kPa)	P_3 (kPa)	P_4 (kPa)	P_6 (kPa)	YP (%)
Base	4.73	6.90	1.336	1.673	0.138	154.32	49.55	71.03	103.15	74.18
Design	4.45	6.81	0.716	1.503	0.097	151.99	50.66	60.80	101.33	95.00

Table 9. Optimization Results for System 3

	F_1	F_2	CV_2	$CV_{\_t}$	T_1 (s)	T_4 (s)	P_1 (kPa)	P_3 (kPa)	P_4 (kPa)	YP (%)	Work
Design	4.45	6.81	0.716	0.097	25	10	151.99	50.66	60.80	95.00	18.62
Opt (FD)	5.06	6.90	0.766	0.097	18.41	5.73	151.99	50.26	60.39	95.00	18.38
Opt (AC)	3.90	6.90	0.729	0.089	26.40	7.25	151.99	50.36	60.49	95.00	18.14

For optimization, we choose to minimize the specific work while maintaining 95% O_2 purity at cyclic steady state

$$\text{specific\_work} = \frac{\int(|v|\,\rho W_s)\,dt}{\int(v\rho)\,dt} \qquad (\text{MJ/kg}\cdot\text{mol}) \quad (37)$$

where

$$\text{work} = W_s = \frac{\gamma P_i v_i}{\gamma - 1}\left[\left(\frac{P_o}{P_i}\right)^{(\gamma-1)/\gamma} - 1\right]$$

Here P_i and P_o are the pressures at the inlet and outlet of the compressors in steps 1, 3, 4, and 6 in Figure 13. Using Eq. 37, specific work is calculated only if $P_o > P_i$; γ is the ratio of constant pressure and constant volume heat capacities (such as $\gamma = 1.4$ for the O_2/N_2 system); and v_i is the specific volume of gas at compressor inlet conditions. Gas density and velocity are evaluated at the inlet of the bed. Additional constraints on the step pressures are $P_1 \geq 151.99$ kPa and $P_4 = P_3 + 10.13$ kPa. Flow rates F_1, F_2, step times T_1 and T_4, valve constants CV_2 and $CV_{\_t}$ are chosen to be decision variables. Upper and lower bounds for the decision variables are specified in Table 6.

The optimization results are also shown in Table 6. We use the design conditions as the benchmark. Here we obtain an 8% savings in specific work. The decision variables generated from finite difference derivatives (FD) are not as good as those generated from accurate sensitivities (AC), due to round-off errors. Here, the finite difference method calculates sensitivities 1.5 times as fast as DASPK, but rSQP converges faster with fewer iterations with accurate sensitivities from DASPK. Besides, finite difference derivatives are noisier, which is indicated by the larger value of convergence error. Table 7 lists the convergence errors at the optimal solution.

The N_2 profiles are shown in Figure 17. Note that at the optimal point for Opt(AC) in Table 6, specific power is reduced by lowering the feed flow rate (for the given bed volume), maintaining P_1 at its lower bound and reducing the step time for the fourth step where the product is recovered. This is also done at a slightly higher pressure in this step.

System 3

System 3 is very similar to System 2, except that the adsorption bed is packed with two different adsorbents. The adsorbent in the first one-third of the bed is inert, while the adsorbent in the next two-thirds is active. This affects the equilibrium partial pressure, the solid loading on the adsorbent, and the energy balance because of the change in adsorption heat.

The bed profiles at CSS are shown in Figure 18. Compared with System 2, the purity of O_2 in System 3 is not as good because only two-thirds of the bed has active adsorbent. Discontinuities appear at the intersection of two adsorbent layers. This disjunction makes our system more nonlinear and harder to solve via the direct-determination method. Here, direct determination is successful only by using trust region/dogleg and scaling methods simultaneously. Otherwise, Newton's method or the trust region approach with dogleg step diverges frequently. The convergence process given by the direct-determination method is shown in Figure 19. It takes 1,800 successive substitution cycles to converge, while direct determination requires only 3 Jacobian evaluations and

Table 10. Convergence Errors at the Optimal Solution for System 3

	$\left\|\frac{\nabla_x L(x_k, \lambda_k)}{\lvert f(x_k)\rvert + 1}\right\|$	$\|c(x_k)\|_\infty$	Iter. No.	CPU h
Opt (FD)	0.24	7.7×10^{-5}	153	122.05
Opt (AC)	3.2×10^{-3}	8.2×10^{-5}	51	68.65

Table 11. CPU Hours for Case Studies

	Unconstrained CSS		Constrained CSS (design)		Optimization	
	Suc. Sub.	Direct Det.	Suc. Sub.	Direct Det.	Black Box	Sim. Tailor
System 1	0.56	0.62	N/A	1.86	56	40.77
System 2	4.55	2.91	N/A	2.91	455	38.03
System 3	6.91	6.04	N/A	6.04	691	68.65

Table 12. Physical Parameters (in the Order of O_2, N_2)

Name	Symbol	Unit	Case 1	Cases 2 & 3
Affinity parameter	b_0	1/kPa	1.52×10^{-6}, 2.20×10^{-6}	6.98×10^{-7}, 1.48×10^{-7}
	d_0	1/kPa		3.24×10^{-6}, 2.19×10^{-7}
Heat of adsorption	q_i	J/mol	13,494, 17,124	16,716, 28,476, 11,298, 19,866
Mass-transport coefficient	k_i	mol(s·kPa·molsolid)	9.87×10^{-4}, 9.87×10^{-4}	7.40×10^{-3}, 7.40×10^{-3}
Length of adsorption column	l	m	1.83	0.46, 1.37
Diameter of adsorption column	d	m	4.25	3.66
Internal void fraction	ϵ_B	m^3 void/m^3 vessel	0.3854	0.35
External void fraction	ϵ_T	m^3 void/m^3 vessel	0.6137	0.6
Bulk density of solid	ρ_s	kg·mol/m^3	2958	3112
Heat capacity of solid	C_s	J/(mol·K)	0.900	0.838
Diameter of adsorbent	d_p	m	0.0015	0.0023
Saturation capacity of adsorbent	m	mol/kg-mol adsorbent	4.13	1.17, 3.32
Gas constant	R_E	J/(mol·K)	8.32	8.32
Product tank pressure	P_{tank}	kPa	113.89	
Molecular weight	$M_{0,i}$	g/mol	32.0, 28.0	
Coefficients for heat capacity	$c^i_{p,A}$	J/(mol·K)	25.48, 28.91	
	$c^i_{p,B}$	J/(mol·K^2)	15.18×10^{-3}, -1.569×10^{-3}	
	$c^i_{p,C}$	J/(mol·K^3)	-7.15×10^{-6}, -2.49×10^{-6}	
	$c^i_{p,D}$	J/(mol·K^4)	1.31×10^{-9}, -2.86×10^{-9}	
Coefficients for viscosity	$\mu_{0,i}$	mol/(s·m)	4.39×10^{-3}, 4.47×10^{-3}	
	$\mu_{1,i}$	mol/(s·K·m)	5.40×10^{-5}, 4.45×10^{-5}	

Table 13. Initial Conditions

Name	Symbol	Unit	Case 1	Cases 2 & 3
Temperature	T	K	302.60	305.37
Pressure	P	kPa	101.33	101.33
Mole fraction	y_i		0.95, 0.05	0.95, 0.05

Table 14. Boundary Conditions

Name	Symbol	Unit	Case 1	Cases 2 & 3
Flow rate	F	m^3/s	11.80	4.73, 6.90
Valve constant	CV		5.086	3.0, 1.336, 0.903, 1.673
Feed temperature	$T_$feed	K	310.93	310.93
Feed pressure	$P_$feed	kPa	96.87	101.33
Feed mole fraction	$Y_$feed		0.22, 0.78	0.22, 0.78

Table 15. Product Tank (Cases 2, 3)

Product tank volume	V	m^3	141.59
Product tank outflow pressure	P_{tank_t}	kPa	101.33
Product tank valve constant	CV_t		0.139

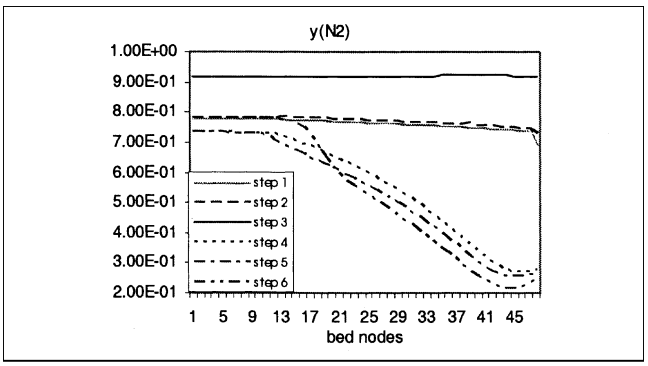
several Broyden updates to converge to CSS. Again we observe some spikes due to the trial trust region sizes, but the trust region approach is able to adjust itself and converge successfully.

The same design variables and constraints are used as in System 2. The results are shown in Table 8. In order to achieve the same purity as System 2, we need to input less flow (F_1), retrieve more (F_2), and set the valve smaller (CV_t). The optimization problem formulation is also similar. Again, F_1 is lowered even further, P_1 remains at its lower bound, and T_4 is reduced, as in System 2. The optimization results are shown in Tables 9 and 10. However, because of the inert adsorbent in the first third of the bed, we only achieve a mere 2.6% savings in specific work when accurate sensitivities from DASPK (AC) are used. The finite difference derivatives (FD) are so noisy that rSQP requires over 120 h and 150 iterations to find an objective function slightly lower than the benchmark. Therefore, the finite difference method is not always reliable and sometimes can yield unacceptable results when models become complex. Figure 20 shows the N_2 profiles under the optimal conditions.

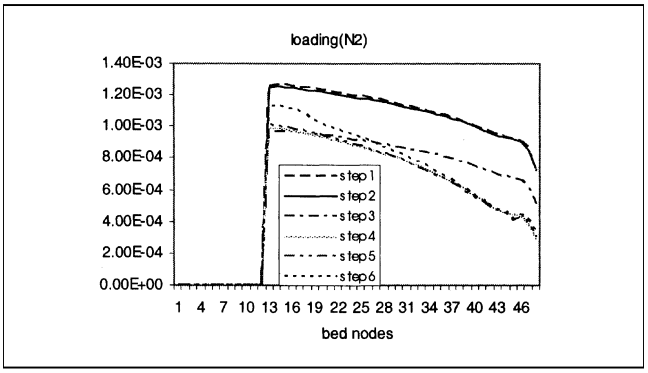
All programs were run on an i686 1GHz Linux machine. Table 11 shows the CPU hours spent for each case study. For the constrained CSS using successive substitution, the performance depends on many tuning variables for the design loops. This is unpredictable, and, thus, no estimate is included in the table. However, we see that by adding design constraints, the computational effort for the direct-determination method rarely increases, which is a big advantage over successive substitution. The cost for optimization using a black-box approach is approximated by multiplying the CPU time for CSS convergence run by the successive-substitution method with 100 experiments that are typically carried out to determine the optimal conditions. While some of these experiments may take less time by restarting at previous CSS points, it is clear from this comparison that the simultaneous tailored approach can significantly improve the optimization efficiency.

Conclusions

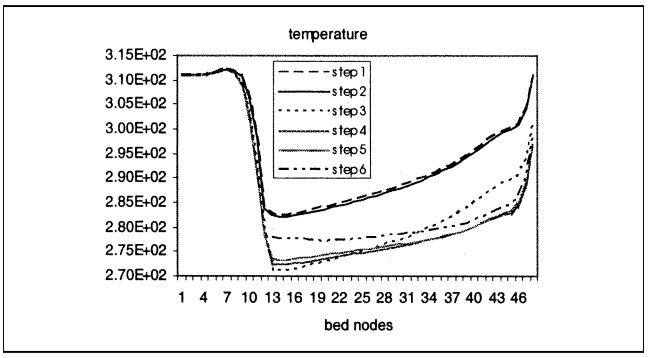
In this work, the performance of successive substitution and a Newton-based approach are compared for converging PSA systems. Using the method of lines, we modified the van Leer flux limiter to smooth the sensitivity calculation, yet still keep its extraordinary ability to track steep fronts accurately. For highly nonlinear and ill-conditioned PSA models, a trust region/dogleg step and scaling are used to ensure robust convergence. Here pure Newton performs much worse and often leads to divergence. For design problems, direct determination requires little additional computational effort and is ef-



(a)



(b)



(c)

Figure 18. (a) N_2 mole fraction; (b) N_2 loading; and (c) temperature profiles at cyclic steady state for system 3.

fective in satisfying multiple control objectives. Thus, it is more advantageous and efficient than successive substitution.

Moreover, a simultaneous tailored approach and state-of-the-art rSQP optimization strategy have been successfully implemented to design optimal PSA processes. We improved the optimization algorithm within rSQP by including a dogleg step so the convergence property has been enhanced for CSS constraints. Low-order methods such as Lagrange polynomial approximation can be used to simplify the model and save computation time. However, the accuracy of design and especially optimization solutions may be sacrificed. We illustrate these approaches using three case studies, which are of a rea-

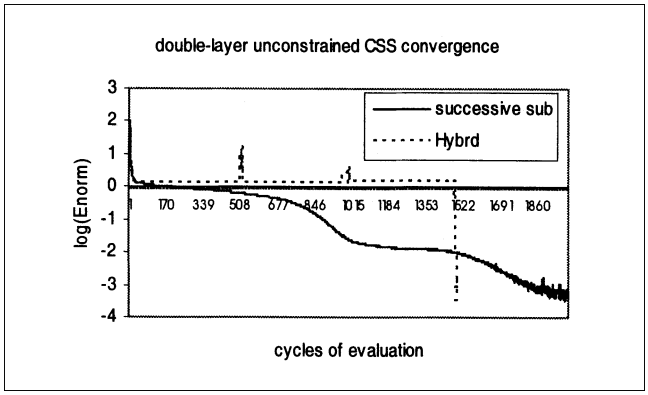
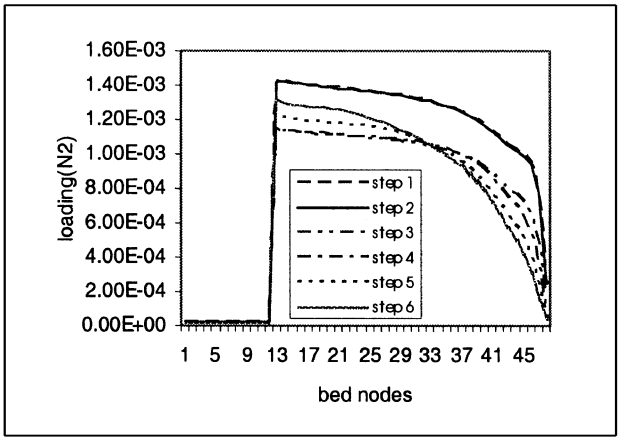
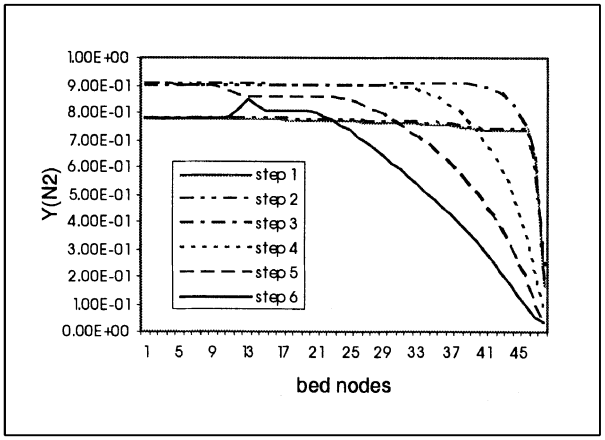


Figure 19. Comparison of convergence speed of successive substitution and direct determination for system 3.



(a)



(b)

Figure 20. (a) N_2 loading; (b) N_2 mole fraction: bed profiles under optimal condition for system 3.

sonably high level of difficulty. Although our case studies are all based on single-bed processes, the same approaches and ideas can be applied directly to multibed processes.

For future work, we will explore optimization techniques for interacting multibed processes. Because of these interactions, the storage and retrieval of profiles and their sensitivities for different steps poses a challenging problem. Finally, since the Jacobian calculation is a bottleneck to our algorithm efficiency, we will use parallel computing to speed up the sensitivity calculation. By spreading sensitivity parameters over different processors, the workload of each processor can be reduced by almost 1/NP of the original load (where NP is the number of processors) and the computation time can be greatly reduced.

Acknowledgments

The authors grateful acknowledge the financial support of the NSF/GOALI program and Air Products and Chemicals Inc. The authors would also like to thank Drs. Ken Anselmo and Mark Daichendt for their input to this research.

Notation

b, d = affinity parameter of adsorbent
c_s = heat capacity of adsorbent
c_p^l = component heat capacity at constant pressure
d_p = adsorbent particle diameter
h = mixture enthalpy
k_i = component mass-transfer coefficient
M = gas mole weight
m_i = saturation capacity of adsorbent
n_i = solid-phase loading
n_i^* = component equilibrium loading
P = total bed pressure
P_i = component partial pressure
P_i^* = component equilibrium partial pressure
q_i = component heat of adsorption
R_E, R_{PV} = gas constants
T = temperature
v = superficial velocity

Greek letters

ϵ_B = interparticle void volume of adsorbent
ϵ_T = total void volume of adsorbent
μ = gas viscosity
ρ_i = component molar density
ρ_s = bulk density of adsorbent

Literature Cited

Biegler, L. T., I. E. Grossmann, and A. W. Westerberg, *Systematic Methods of Chemical Process Design*, Prentice Hall, Upper Saddle River, NJ (1997).

Biegler, L. T., I. E. Grossmann, and A. W. Westerberg, "Notes on Approximation Techniques Used for Process Optimization," *Comput. Chem. Eng.*, **9**(2), 201 (1985).

Bird, R. B., W. E. Stewart, and E. N. Lightfoot, *Transport Phenomena*, Wiley, New York (1960).

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

Boggs, P. T., and J. W. Tolle, "Sequential Quadratic Programming for Large-Scale Nonlinear Optimization," *J. Comput. Applied Math.*, **124**(1), 123 (2000).

Croft, D. T., and M. D. LeVan, "Periodic States of Adsorption Cycles—I. Direct Determination and Stability," *Chem. Eng. Sci.*, **49**, 1821 (1994).

Ding, Y., and M. D. LeVan, "Periodic States of Adsorption Cycles III. Convergence Acceleration for Direct Determination," *Chem. Eng. Sci.*, **56**(17), 5217 (2001).

Feehery, W., J. Tolsma, and P. Barton, "Efficient Sensitivity Analysis of Large-Scale Differential-Algebraic Systems," *Appl. Numer. Math.*, **25**(1), **41** (1997).

Gill, P. E., W. Murray, and M. H. Wright, *Practical Optimization*, Academic Press, New York (1989).

Hirsch, C., *Numerical Computation of Internal and External Flow*, Wiley, New York (1988).

Ko, D., R. Siriwardane and L. T. Biegler, "Optimization of Pressure Swing Adsorption Process using Zeolite 13X for CO_2 Sequestration," *Ind. Eng. Chem. Res.*, **42**, 329 (2003).

Kumar, R., V. G. Fox, D. 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).

Kvamsdal, H. M., and T. Hertzberg, "Optimization of PSA Systems-Studies on Cyclic Steady State Convergence," *Comput. Chem. Eng.*, **21**(8), 819 (1997).

Leis, J. R., and M. A. Kramer, "The Simultaneous Solution and Sensitivity Analysis of Systems Described by Ordinary Differential Equations," *ACM Trans. Math. Soft.*, **14**(1), 45 (1988).

LeVeque, R. J., *Numerical Methods for Conservation Laws*, Birkhäuser, Basel (1992).

Leonard, B. P., "The ULTIMATE Conservative Difference Scheme Applied to Unsteady One Dimensional Advection," *Comput. Methods Appl. Mech. Eng.*, **88**, 17 (1991).

Li, S., and L. Petzold, *Design of New DASPK for Sensitivity Analysis*, Tech. Rep., Univ. of California, Santa Barbara, CA (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).

Maly, T., and L. Petzold, "Numerical Methods and Software for Sensitivity Analysis of Differential Algebraic Equations," *Appl. Numer. Math.*, **20**, 57, (1996).

More, J. J., B. S. Garbow, and K. E. Hillstrom, *User Guide for Minpack-1*, ANL-80-74, Argonne National Laboratory, Argonne, IL (1980).

Nash, S. G., and A. Sofer, *Linear and Nonlinear Programming*, McGraw-Hill, New York (1996).

Nilchan, S., *The Optimization of Periodic Adsorption Processes*, PhD Diss., Imperial College, London (1997).

Nilchan, S., and C. C. Pantelides, "On the Optimisation of Periodic Adsorption Processes," *Adsorption*, **4**, 113 (1998).

Powell, M. J. D., "A Fortran Subroutine for Solving Systems of Nonlinear Algebraic Equations," *Numerical Methods for Nonlinear Algebraic Equations*, P. Rabinowitz, ed., Gordon and Breach, London (1970)

Ritter, J. A., and R. Yang, "Pressure Swing Adsorption: Experimental and Theoretical Study on Air Purification and Vapor Recovery," *Ind. Eng. Chem. Res.*, **30**, 1023 (1991).

Smith, O. J., and A. W. Westerberg, "Acceleration of Cyclic Steady State Convergence for Pressure Swing Adsorption Models," *Ind. Eng. Chem. Res.*, **31**, 1569 (1992).

Smith, O. J., and A. W. Westerberg, "The Optimal Design of Pressure Swing Adsorption Systems," *Chem. Eng. Sci.*, **46**, 2967 (1991).

Stewart, W. E., M. Caracotsios, and J. P. Sorensen, "DDASAC Software Package Documentation," Chemical Engineering Department, Univ. of Wisconsin—Madison, WI (1995).

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

Webley, P. A., and J. He, "Fast Solution-Adaptive Finite Volume Method for PSA/VSA Cycle Simulation; 1 Single Step Simulation," *Comput. Chem. Eng.*, **23**, 1701 (2000).

Webley, P. A, "Optimization of PSA Systems for Air Separation," *6th Proc. Fundamentals of Adsorption Conf.*, Giens, France (1998).

Zhu, W., and L. Petzold, "Parallel Sensitivity Analysis for DAEs with Many Parameters," *Concurr. Pract. Exper.*, **11**(10), 571 (1999).

Manuscript received Mar. 25, 2002, and revision received Dec. 11, 2002.