Adsorption (2008) 14: 567–573
DOI 10.1007/s10450-008-9130-0

Optimisation-based simulation of a pressure swing adsorption process

M.A. Latifi · D. Salhi · D. Tondeur

Received: 25 April 2007 / Revised: 4 October 2007 / Accepted: 18 April 2008 / Published online: 13 May 2008

Abstract In this paper, an optimisation-based approach is developed for the determination of the cyclic steady-sate (CSS) of a pressure swing adsorption process (PSA). It consists in treating the simulation problem as a single dynamic optimisation problem where the performance index is the CSS condition, the decision variables are the state variables at the start of the cycle and the constraints are given by the process model equations with associated initial conditions. The resulting optimisation problem is solved using a gradient-based non linear programming (NLP) method, e.g. SQP method, where the gradients are computed by means of four different methods: finite differences, numerical and analytical sensitivities and adjoint system methods.

Keywords PSA process · Cyclic steady-state · Simulation · Optimisation

1 Introduction

Adsorption processes are commonly used in the separation of gas mixtures as an alternative to traditional separation processes such as distillation and absorption. More specifically, they are involved in operations like air separation, hydrogen purification and dehumidification of gas streams (Tondeur and Wankat 1985; Ruthven et al. 1994). Since PSA processes are widely used in industry, their efficient modelling, simulation and optimisation are of interest and deserve to be investigated. Simulation of PSA processes has been the subject of a large number of research contributions, particularly in convergence improvements to cyclic steady-state (CSS). However, optimisation-based approaches have not been extensively used in simulation of adsorption processes and very few works have been recently devoted to PSA processes (Jiang et al. 2003, 2005; Knaebel et al. 2005). The objective of this paper is to present some results obtained in the simulation of a non isothermal PSA process based on a dynamic optimisation approach.

M.A. Latifi (✉) · D. Salhi · D. Tondeur
Laboratoire des Sciences du Génie Chimique, CNRS-ENSIC,
B.P. 20451, 1 rue Grandville, 54001 Nancy Cedex, France
e-mail: latifi@ensic.inpl-nancy.fr

2 Process model

The system considered here belongs to the recent environmental applications and concerns a PSA process with zeolite 13X as solid adsorbent used for removal of CO_2 from a mixture of CO_2–N_2 where N_2 is inert. The process used consists of four basic steps (classical Skarstrom cycle): compression, high pressure adsorption, blow down and low pressure purge. The process model is similar to those described in (Liu and Ritter 1996; Liu et al. 1998) and is based on the following assumptions:

- A monodimensional model is considered
- The system is non isothermal
- The pressure drop is negligible
- The axial dispersion is negligible
- The gas phase is ideal
- The carrier gas is assumed to be inert
- The fluid velocity is constant
- The mass transfer rate is described by LDF (linear driving force) model

The resulting model is constituted by the following equations:

Table 1 Boundary conditions for the four basic steps

Step I (compression)	Step II (adsorption)
$P = f(t)$	$P = P_H$
$t = 0; y = y_{IV}, q = q_{IV}, T = T_{IV}$	$t = 0; y = y_I, q = q_I, T = T_I$
$z = 0; \forall t : y = y_{feed}, T = T_{feed}$	$z = 0; \forall t : y = y_{feed}, T = T_{feed}, u = u_H$
$z = L; \forall t : u = 0$	
Step III (blowdown)	**Step IV (purge)**
$P = f(t)$	$P = P_L$
$t = 0; y = y_{II}, q = q_{II}, T = T_{II}$	$t = 0; y = y_{III}, q = q_{III}, T = T_{III}$
$z = L; \forall t : \frac{\partial y}{\partial z} = 0, T = T_{ads}, u = 0$	$z = L; \forall t : y = y_{ads}\frac{P_L}{P_H}, T = T_{ads}, u = u_L$

- Carbon dioxide mass balance in the fluid phase:

$$\frac{\partial y}{\partial t} + u\frac{\partial y}{\partial z} + (1-y)\frac{1-\epsilon}{\epsilon}\frac{RT}{P}\rho_s\frac{\partial q}{\partial t} = 0 \tag{1}$$

- Energy balance:

$$\left(\rho_g C_{pg} + \frac{1-\epsilon}{\epsilon}\rho_s C_{ps}\right)\frac{\partial T}{\partial t} + \rho_g C_{pg} u\frac{\partial T}{\partial z} + \frac{1-\epsilon}{\epsilon}\Delta H\rho_s\frac{\partial q}{\partial t} + \frac{2h}{\epsilon r_b}(T - T_0) = 0 \tag{2}$$

- LDF model:

$$\frac{\partial q}{\partial t} = k_a(q^* - q) \tag{3}$$

- Equilibrium relationship for carbon dioxide:

$$q^* = \frac{q_s b P y}{1 + bPy} \tag{4}$$

where $b = \frac{b_0}{R\sqrt{T}}\exp(\frac{-\Delta H}{RT})$

The associated initial and boundary conditions for the four basic steps are given in Table 1.

Equations (1–4) and those in Table 1 form a set of partial differential algebraic equations (PDAEs) which is transformed into a set of partial differential equations (PDEs) since the system index is one (Unger et al. 1995; Brenan et al. 1989).

3 Spatial derivatives approximation

A method of lines methodology (Schiesser 1991) is used to discretise the spatial dimension in order to reduce the system of PDEs to a system of ordinary differential equations (ODEs). Here orthogonal collocation method based on Lagrange polynomials (Villadsen and Michelsen 1978) is applied. The approximations are as follows:

$$y(t,z) \approx \sum_{i=0}^{N+1}\overline{y}_i(t)l_i(z),$$

$$q(t,z) \approx \sum_{i=0}^{N+1}\overline{q}_i(t)l_i(z),$$

$$T(t,z) \approx \sum_{i=0}^{N+1}\overline{T}_i(t)l_i(z) \tag{5}$$

where $l_i(z) = \prod_{j=0, j\neq i}^{N+1}\frac{(z-z_j)}{(z_i-z_j)}$ are the Lagrangian polynomials.

The resulting process model may be written in the following ODEs:

$$\frac{d\overline{y}_i}{dt} + u\sum_{j=0}^{N+1}\overline{y}_j(t)\dot{l}_j(z_i) + (1-\overline{y}_i)\frac{1-\epsilon}{\epsilon}\frac{RT}{P}\rho_s\frac{(1-\epsilon)\rho_s}{\epsilon}\frac{d\overline{q}_i}{dt} = 0, \tag{6}$$

$$\left(\rho_g C_{pg} + \frac{1-\epsilon}{\epsilon}\rho_s C_{ps}\right)\frac{d\overline{T}_i}{dt} + \rho_g C_{pg} u\sum_{j=0}^{N+1}\overline{T}_j(t)\dot{l}_j(z_i) + \frac{1-\epsilon}{\epsilon}\Delta H\rho_s\frac{d\overline{q}_i}{dt} + \frac{2h}{\epsilon r_b}(\overline{T}_i - T_0) = 0, \tag{7}$$

$$\frac{d\overline{q}_i}{dt} = k_a(\overline{q}_i^* - \overline{q}_i), \tag{8}$$

$$\overline{q}_i^* = \frac{q_s\overline{b}_i P\overline{y}_i}{1 + \overline{b}_i P\overline{y}_i}, \tag{9}$$

$$\overline{b}_i = \frac{b_0}{R\sqrt{\overline{T}_i}}\exp\left(-\frac{\Delta H}{R\overline{T}_i}\right) \tag{10}$$

for $i = 1, 2, \ldots, N$.

The collocated boundary conditions for the four basic steps are listed in Table 2.

The resulting system of ODEs (6–9) may be written in the following classical form:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}), \tag{11}$$

$$\mathbf{x}(0) = \mathbf{x}_0 \tag{12}$$

Table 2 Collocated boundary conditions for the four basic steps where $\dot{l}_i(z_j) = \frac{l_i(z_j)}{dz}$

Step I (compression)	Step II (adsorption)
$\forall t$	$\forall t$
$P = f(t)$	$P = P_H$
$\overline{y}_0 = y_{feed}, \overline{T}_0 = T_{feed}, u = 0$	$\overline{y}_0 = y_{feed}, \overline{T}_0 = T_{feed}, u = u_H$
Step III (blowdown)	**Step IV (purge)**
$\forall t$	$\forall t$
$P = f(t)$	$P = P_L$
$\sum_{j=0}^{N+1} \overline{y}_j(t)\dot{l}_j(z_{N+1}), \overline{T}_{N+1} = T_{ads}, u = 0$	$\overline{y}_{N+1} = y_{ads}\frac{P_L}{P_H}, \overline{T}_{N+1} = T_{des}, u = u_L$

where the state vector is given by

$$\mathbf{x}^T = (\overline{y}_0, \overline{y}_1, \ldots, \overline{y}_{N+1}, \overline{q}_0, \overline{q}_1, \ldots, \overline{q}_{N+1}, \overline{T}_0, \overline{T}_1, \ldots, \overline{T}_{N+1})$$

It is constituted by CO_2 concentrations in fluid and solid phases and temperatures at different collocation points.

The initial state is given by

$$\mathbf{x}_0^T = (\overline{y}_0^0, \overline{y}_1^0, \ldots, \overline{y}_{N+1}^0, \overline{q}_0^0, \overline{q}_1^0, \ldots, \overline{q}_{N+1}^0, \overline{T}_0^0, \overline{T}_1^0, \ldots, \overline{T}_{N+1}^0)$$

The solution of the process model equations, i.e. simulation, is dealt with in the next section.

4 Simulation problem definition

The objective of the simulation problem is to determine the cyclic steady-state, i.e., the state vector at the start of the cycle must be equal to the state vector at the end of the cycle. This is formulated as

$$\mathbf{x}(0) = \mathbf{x}(T_{cycle}) \tag{13}$$

The cycle duration is expressed by

$$T_{cycle} = \tau_{comp} + \tau_{ads} + \tau_{blow} + \tau_{des} \tag{14}$$

5 Classical formulation and computation methods

Different methods to solve (13) have been developed and proposed in the literature. The main methods are:

5.1 Fixed-point iteration approach

In this approach the state vector values at the end of a cycle are used at the start of the following cycle. This is formulated as

$$\mathbf{x}^{c+1}(0) = \mathbf{\Psi}[\mathbf{x}^c(0)], \quad c = 1, 2, \ldots, n_{cycles} \tag{15}$$

The substitution process is repeated until convergence.

This method is easy to implement but has only first-order convergence rate. To reach the CSS hundreds of cycles are required.

5.2 Quasi-Newton based methods

In quasi-Newton methods the search for cyclic steady-state is formulated as

$$\mathbf{x}(0) - \mathbf{\Psi}[\mathbf{x}(0)] = 0 \tag{16}$$

where $\mathbf{x}(0)$ is the unknown state vector.

These methods have the advantage of super-linear convergence rate but have also problems in Jacobian matrix computation since numerical methods (mainly finite differences) are used with the first three direct successive substitutions (Smith and Westerberg 1991).

5.3 Newton methods

To overcome the problem of numerical estimation of Jacobian matrix, it was suggested to use sensitivities method $\partial\mathbf{x}(t)/\partial\mathbf{x}(0), t \in [0, T_{cycle}]$ (Croft and LeVan 1994). This allows the application of a Newton method and therefore guarantees the quadratic convergence rate. The problem is that the computational times are very often increased because of the size and non sparsity of the sensitivities matrix (Nilchan and Pantelides 1998).

6 Optimisation-based formulation and computational method

6.1 Problem formulation

The approach developed here consists in treating the simulation problem as a single dynamic optimisation problem where the performance index is the CSS condition, the decision variables are the state variable at the start of the cycle and the constraints are given by the process model equations with associated initial conditions.

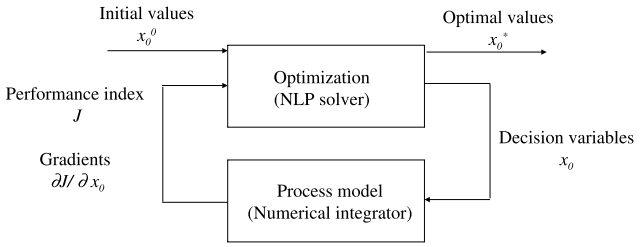

Fig. 1 Optimisation process

The mathematical formulation of the simulation problem is the following:

$$\underset{\mathbf{x}_0}{\text{Min}} J = \frac{1}{2}\mathbf{e}^T\mathbf{e} \quad \text{where } \mathbf{e} = \mathbf{x}(0) - \mathbf{x}(T_{cycle}) \tag{17}$$

subject to constraints (11–12).

6.2 Computational method used

The computational method used consists in the estimation of initial values of decision variables which are used in process model integration. The performance index and the gradients of both performance index and constraints with respect to decision variables are computed and provided to a gradient-based non linear programming (NLP) solver, which in turn estimates a new vector of decision variables. The process is repeated until convergence where the optimal values of decision variables are obtained (Fig. 1).

It is important to notice that in any gradient-based optimisation method, the convergence and its rate strongly depend on the accuracy of gradients computations. In this work four methods are used and compared. They are based on the following general definition of the performance index:

$$\underset{\mathbf{p}}{\text{Min}} J = G[\mathbf{x}(t_f), \mathbf{p}] + \int_0^{t_f} F[\mathbf{x}(t), \mathbf{p}]dt \tag{18}$$

where $\mathbf{x}_0$ is substituted by $\mathbf{p}$ in order to clearly distinguish the state vector $\mathbf{x}$ from the vector of decision variables $\mathbf{x}_0$.

6.2.1 *Finite differences method*

The approximation of the gradient of the performance index J with respect to a parameter p_i by means of finite differences consists in perturbating J with a finite amount Δp_i of p_i as follows

$$\frac{\partial J}{\partial p_i} \approx \frac{J(p_i + \Delta p_i) - J(p_i)}{\Delta p_i} \tag{19}$$

where $\frac{\Delta p_i}{p_i} \approx 1\%; i = 1, 2, \ldots, 3(N+2)$.

6.2.2 *Sensitivity method*

The differentiation of the performance index J with respect to a parameter p_i leads to the following equation:

$$\frac{\partial J}{\partial p_i} = \frac{\partial G}{\partial \mathbf{x}^T}\mathbf{s}_i\bigg|_{t=t_f} + \frac{\partial G}{\partial p_i} + \int_0^{t_f}\left(\frac{\partial F}{\partial \mathbf{x}^T}\mathbf{s}_i + \frac{\partial F}{\partial p_i}\right)dt \tag{20}$$

where $\mathbf{s}_i = \frac{\partial \mathbf{x}}{\partial p_i}$, $i = 1, 2, \ldots, 3(N+2)$, is the vector of sensitivities of $\mathbf{x}$ with respect to p_i, deduced from:

$$\dot{\mathbf{s}}_i = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\mathbf{s}_i + \frac{\partial \mathbf{f}}{\partial p_i}; \quad \mathbf{s}_i(0) = \mathbf{s}_{i0} \tag{21}$$

where $\mathbf{s}_i$ is the vector $(s_1^i, s_2^i, \ldots, s_{3(N+2)}^i)^T$.

The partial derivatives of G, F and $\mathbf{f}$ with respect to $\mathbf{x}$ and p_i are computed using DASPK code (Brown et al. 1994; Li and Petzold 1999). When they are computed numerically, the resulting sensitivities are referred to as *numerical sensitivities*. Their analytical computation is carried out using *Adifor* package (Bischof et al. 1992). In this case the resulting sensitivities are referred to as *analytical sensitivities*. The analytical sensitivities approach is similar to the method developed by (Croft and LeVan 1994), referred to as "direct determination method". The only difference is that the authors solved directly the CSS condition, i.e. $e[x(0)] = x(0) - x(T_{cycle}) = x(0) - F[x(0)] = 0$, by means of a Newton method.

The computational algorithm using the finite differences or the sensitivity method is the following

(i) Estimation of the vector of parameters $\mathbf{p}$
(ii) Forward integration of state equations (11–12) and sensitivity equations (21)
(iii) Computation of gradients by means of (19) or (20)
(iv) Estimation of a new vector of parameters $\mathbf{p}$ by means of an NLP solver and go to step (ii)
(vi) Iteration until convergence

6.2.3 *Adjoint system method*

Consider the simple formulation of a dynamic optimisation problem where the performance index is given by (18) and the process model is described by the system of ODEs (11–12). The gradients of the performance index and constraints with respect to parameters are given by:

$$\frac{\partial J}{\partial p_i} = \frac{\partial G}{\partial p_i} + \int_0^{t_f}\frac{\partial H}{\partial p_i}dt \tag{22}$$

where

$$H = F(\mathbf{x}, \mathbf{p}) + \lambda^T\mathbf{f}(\mathbf{x}, \mathbf{p}) \tag{23}$$

is the Hamiltonian function of the performance index.

The vector of co-state or adjoint variables λ is defined as:

$$\frac{d\lambda}{dt} = -\frac{\partial H}{\partial \mathbf{x}} \tag{24}$$

and the corresponding terminal conditions are expressed as:

$$\lambda(t_f) = \left.\frac{\partial G}{\partial \mathbf{x}}\right|_{t=t_f} \tag{25}$$

$i = 0, 1, \ldots, 3(N+2)$.

The computation algorithm using the adjoint system method is the following

(i) Estimation of the vector of parameters **p**
(ii) Forward integration of state equations (11–12)
(iii) Backward integration of adjoint or co-state equations (24–25)
(iv) Computation of gradients by means of (22)
(v) Estimation of a new vector of parameters **p** by means of an NLP solver and go to step (ii)
(vi) Iteration until convergence

Table 3 Physical parameters of the model

Parameter	Unit	Value
k_a	s^{-1}	9×10^{-3}
ρ_s	kg/m^3	720
ϵ	–	0.40
τ	s	360
L	m	1.2
u_H	m/s	0.2
u_L	m/s	0.6
P_H	atm	4.0
P_L	atm	1.3
C_{pg}	J/kg/K	29.1
ΔH	J/mol	34,230
r_b	m	0.0135
T_0	K	293.15
C_{ps}	J/kg/K	924.0
q_s	mol/kg	2.98
h	$W/m^2/K$	45.0
b_0	$m^3/mol/K^{0.5}$	3.88×10^{-8}
y_{feed}	% (vol)	5
ρ_g	kg/m^3	1.3

7 Simulation results

7.1 Data used

The pressure profiles in compression and blowdown steps are:

$$P(t) = P_H + (P_L - P_H)\left[\frac{t^2}{\tau_{comp}^2} - \frac{2t}{\tau_{comp}} + 1\right], \tag{26}$$

$$P(t) = P_L + (P_H - P_L)\left[\frac{t^2}{\tau_{blow}^2} - \frac{2t}{\tau_{blow}} + 1\right] \tag{27}$$

The pressure profiles used in this work are similar to those in (Kapoor and Yang 1989) where the pressure is recorded and fitted by a quadratic function of time. On the other hand, we assumed that the durations of adsorption and desorption steps are equal ($\tau_{ads} = \tau_{des} = \tau$). In the same way, the durations of compression and blowdown steps are the same and equal to ten times less than the duration of adsorption and desorption steps ($\tau_{comp} = \tau_{blow} = 0.1\tau$).

The number of collocation points N is 15.

The rest of the physical parameters are given in Table 3.

7.2 Results and discussion

The simulations are carried out on a 1.6 GHz Athlon MP 1900+ computer. The NLP solver used is NLPQL developed by Schittkowski (1985) and DASSL code is used as the integrator (Brenan et al. 1989). The sensitivities (numerical and analytical) are computed using the DASPK code (Brown et al. 1994; Li and Petzold 1999). Finally, the adjoint system method is developed within the dynamic optimisation package DYNO (Fikar and Latifi 2002).

Table 4 presents the optimal values of the performance index, i.e. the CSS condition, and their corresponding CPU times for the four methods of gradients computation.

It can be seen that the four methods lead to the same values of CSS condition, but the adjoint method has by far the fastest convergence rate especially when compared to analytical sensitivity method. The difference is due to the high dimensions of the process model and of the vector of decision variables. It is important to notice that in the sensitivity method each decision variable requires to solve an additional system of ODEs with the same dimension as the

Table 4 Methods of gradients computation in simulation

	Finite differences method	Sensitivity method		Adjoint system method
		Numerical	Analytical	
CPU time (s)	581	315	62	24
Performance index	0.91×10^{-10}	0.96×10^{-10}	0.93×10^{-10}	0.90×10^{-10}

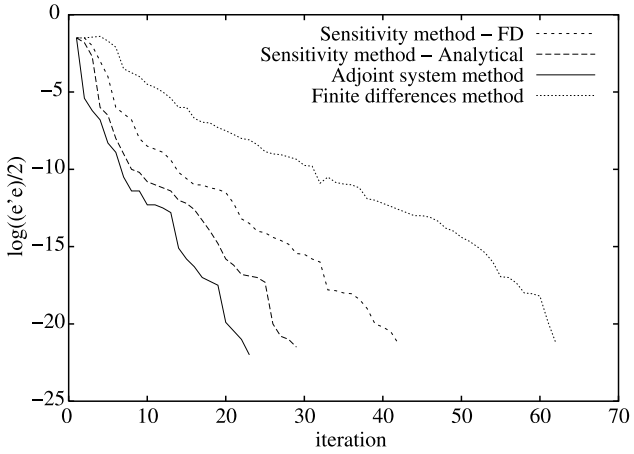

Fig. 2 CSS condition *vs* iteration number

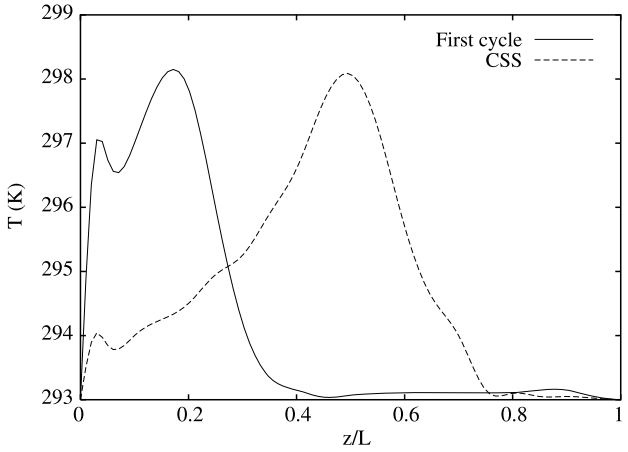

Fig. 5 Temperature profiles in the fluid after the adsorption step *vs* dimensionless bed length

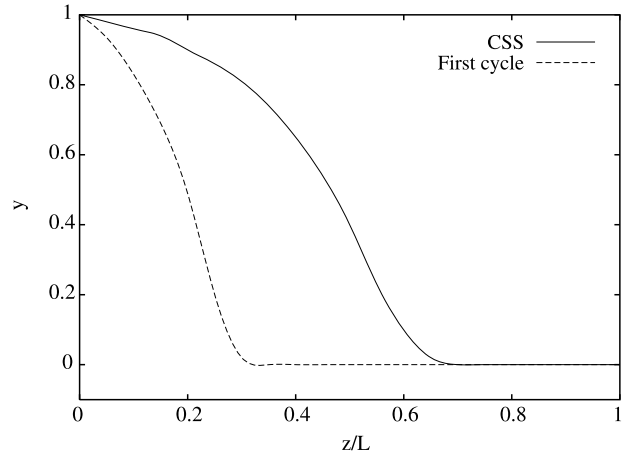

Fig. 3 CO_2 Dimensionless concentration profiles in the fluid after the adsorption step *vs* dimensionless bed length

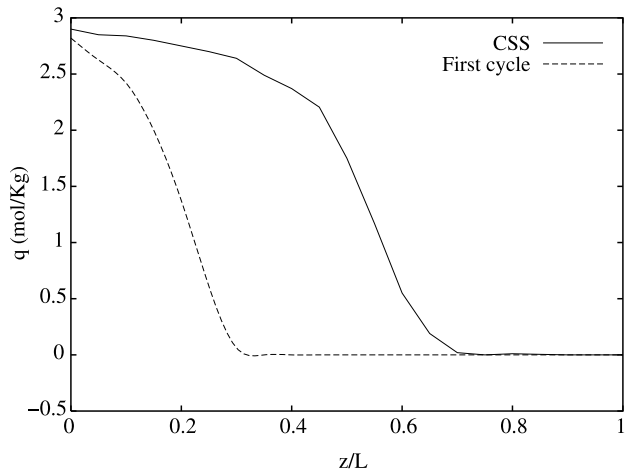

Fig. 4 CO_2 Concentration profiles in the solid phase after the adsorption step *vs* dimensionless bed length

process model. It is evident that this needs additional computational time which is not the case in the adjoint system method.

In Fig. 2 the comparison of different methods of gradients computation is presented as the variations of the CSS condition versus iteration number. This figure confirms the performance of the adjoint system method. Figures 3, 4 and 5 present respectively the concentrations of carbon dioxide in the fluid and solid phases and the fluid phase temperature after the adsorption step versus dimensionless bed length. Two profiles are reported in each figure, one corresponds to the first cycle or iteration (dashed line) and the other to CSS (solid line).

8 Concluding remarks

An optimisation—based approach has been developed for the determination of CSS condition of a PSA process used for separation of the binary mixture CO_2/N_2. The approach is based on simultaneous treatment of simulation and optimisation as a single optimisation problem. The NLP solver used is a gradient-based method (i.e., SQP method). The adjoint system method was specifically developed and used for gradients computation and compared to finite differences and sensitivity (numerical and analytical) methods. It was shown that this method has the fastest convergence rate. The approach is not limited by the number of components involved provided that appropriate kinetic, hydrodynamic, thermodynamic and thermokinetic models are used.

9 Nomenclature

C_p	Heat capacity	J/kg/K
ΔH	Heat of adsorption	J/mol
h	Heat transfer coefficient	$W/m^2/K$
k_a	Mass transfer coefficient	s^{-1}
L	Bed length	m
N	Number of collocation points	–
P	Pressure	atm
q	Amount adsorbed	mol/kg
q^*	Equilibrium amount adsorbed	mol/kg
r_b	Bed radius	m
t	Time	m
T	Temperature	K
u	Intersticial velocity	m/s
y	Gas phase mole fraction	–
z	Axial position	m

Greek symbols

ρ	Density	kg/m^3
ϵ	Bed porosity	–
τ	Time duration	s

Subscripts

ads	Adsorption
blow or L	Blowdown
comp or H	Compression
des	Desorption
g	Gas
s	Solid

References

Bischof, C., Carle, A., Corliss, G., Griewank, A., Hovland, P.: Adiforgenerating derivative codes from FORTRAN programs. Sci. Program. **1**, 1–9 (1992)

Brenan, K.E., Campbell, S.E., Petzold, L.R.: Solution of Initial Value Problems in Differential-Algebraic Equations. North-Holland, New York (1989)

Brown, P.N., Hindmarsh, A.C., Petzold, L.R.: Using Krylov methods in the solution of large-scale differential-algebraic systems. SIAM J. Sci. Comput. **15**, 1467–1488 (1994)

Croft, D.T., LeVan, M.D.: Periodic states of adsorption cycles—I. Direct determination and stability. Chem. Eng. Sci. **49**, 1821–1829 (1994)

Fikar, M., Latifi, M.A.: User's guide for FORTRAN dynamic optimisation code DYNO. Technical report, Laboratoire des Sciences du Génie Chimique, CNRS, Nancy, France (2002)

Jiang, L., Biegler, L.T., Fox, V.G.: Simulation and optimization of pressure swing adsorption systems for air separation. AIChE J. **49**, 1140–1157 (2003)

Jiang, L., Biegler, L.T., Fox, V.G.: Design and optimization of pressure swing adsorption systems with parallel implementation. Comput. Chem. Eng. **29**, 393–399 (2005)

Kapoor, A., Yang, R.T.: Kinetic separation of methane-carbon dioxide mixture by adsorption on molecular sieve carbon. Chem. Eng. Sci. **44**, 1723–1733 (1989)

Knaebel, S.P., Ko, R.D., Biegler, L.T.: Simulation and optimization of a pressure swing adsorption system: recovering hydrogen from methane. Adsorption **11**, 615–620 (2005)

Li, S., Petzold, L.R.: Design of new DASPK for sensitivity analysis. Technical report, University of California, Santa Barbara, CA, USA (1999)

Liu, Y., Ritter, J.A.: Pressure swing adsorption-solvent vapor recovery: process dynamics and parametric study. Ind. Eng. Chem. Res. **35**, 2229–2312 (1996)

Liu, Y., Delgado, J., Ritter, J.A.: Comparison of finite differences techniques for simulating pressure swing adsorption. Adsorption **4**, 337–344 (1998)

Nilchan, S., Pantelides, C.C.: On the optimisation of periodic adsorption processes. Adsorption **4**, 113–147 (1998)

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

Schiesser, W.E.: The Numerical Method of Lines. Academic Press, New York (1991)

Schittkowski, K.: NLPQL: a FORTRAN subroutine solving constrained nonlinear programming problems. Ann. Oper. Res. **5**, 485–500 (1985)

Smith, O.J., Westerberg, A.W.: The optimal design of pressure swing adsorption systems. Chem. Eng. Sci. **46**, 2967–2976 (1991)

Tondeur, D., Wankat, P.C.: Gas purification by pressure swing adsorption. Sep. Purif. Methods **14**, 157–212 (1985)

Unger, J., Kroner, K., Marquardt, W.: Structural analysis of differential-algebraic equation systems—theory and applications. Comput. Chem. Eng. **19**, 867–882 (1995)

Villadsen, J., Michelsen, M.L.: Solution of Differential Equation Models by Polynomial Approximation. Prentice-Hall, Englewood Cliffs (1978)