Engineering Optimization
Vol. 00, No. 00, Month 200x, 1–24

RESEARCH ARTICLE

A Multi-objective Genetic Algorithm for the Design of Pressure Swing Adsorption

Giovanna Fiandaca & Eric S Fraga,
Centre for Process Systems Engineering
Department of Chemical Engineering, University College London (UCL)
e.fraga@ucl.ac.uk

Stefano Brandani, Institute for Materials and Processes
School of Engineering and Electronics, University of Edinburgh
(*Received 00 Month 200x; final version received 00 Month 200x*)

Pressure Swing Adsorption (PSA) is a cyclic separation process, more advantageous over other separation options for middle scale processes. Automated tools for the design of PSA processes would be beneficial for the development of the technology, but their development is a difficult task due to the complexity of the simulation of PSA cycles and the computational effort needed to detect the performance at cyclic steady state.

We present a preliminary investigation of the performance of a custom multi-objective genetic algorithm (MOGA) for the optimisation of a fast cycle PSA operation, the separation of air for N_2 production. The simulation requires a detailed diffusion model, which involves coupled nonlinear partial differential and algebraic equations (PDAEs). The efficiency of MOGA to handle this complex problem has been assessed by comparison with direct search methods. An analysis of the effect of MOGA parameters on the performance is also presented.

1. Introduction

During the past three decades there has been an increasing demand for separation and purification technologies arising for a wide range of industries, such as the chemical, the petrochemical, the pharmaceutical and the electronic gas industries (Sircar 2006). Operations range from air separation for O_2 and/or N_2 production, purification of H_2, separation of hydrocarbons and CO_2 capture from flue gases. The available processes for gas separation are absorption, membrane separation, adsorption and cryogenic distillation (Aaron and Tsouris 2005). For middle scale operations, Pressure Swing Adsorption (PSA) is the most convenient option (Ruthven *et al.* 1993). In fact, PSA is a largely established technology, and several hundreds of PSA plants are operating all over the world (Sircar 2006).

Despite the rapid growth in practical applications of PSA and the growing accuracy of bed models, the design and optimisation still remain an experimental effort (Sircar 2006). This is due to the inherent complexity and variety of PSA processes and to the computational effort required by their simulation. Although many studies have addressed the design of PSA cycles via optimisation (Nilchan and Panthelides 1998, Biegler *et al.* 2004, Ding *et al.* 2002, Cruz *et al.* 2005), few of them have considered fast cycles and multiple-bed operations, features necessary to reduce the costs of the operation and increase efficiency. Furthermore, few studies have addressed the multi-objective optimisation of PSA operations, which would help identify the trade-offs between the different aspects of the performance. However, in view of the large number of degrees of freedom, a mathematical programming approach to the opti-

ISSN: 0305-215X print/ISSN 1029-0273 online

DOI: 10.1080/0305215YYxxxxxxx
http://www.informaworld.com

misation of PSA processes is desirable (Nilchan and Panthelides 1998, Biegler *et al.* 2004) to allow the enhancement of the performance of PSA cycles, and subsequently to expand the application of the process.

Section 2 introduces the difficulties linked to the simulation of PSA cycles, while section 3 provides an overview of the optimisation approaches used in literature for their design. Our case study is introduced in section 4.1. The process investigated is the separation of air for N_2 production. The model is presented in section 4.1, while the design problem and its characterisation are in section 4.2. The optimisation approach is analysed in section 5: a custom multi-objective algorithm is presented, whose main feature is the definition of a fitness function especially designed to broaden the extent of the Pareto front. The efficiency of the algorithm is proven by comparison with direct search methods and random search in section 6.1. The effect of the parameters of the multi-objective genetic algorithm (MOGA) is studied in section 6.2. Finally, the design information obtained is presented together with the validation against literature data.

2. Pressure Swing Adsorption

Pressure swing adsorption (PSA) is a cyclic operation for the separation of gaseous mixtures, introduced by Skarstrom (Ruthven *et al.* 1993) as a four-step operation: pressurisation, adsorption at high-pressure, counter-current blowdown to low pressure and low-pressure purge step, or desorption. The process involves two beds, as illustrated in fig.1. The selectivity in a PSA process depends on differences in either the adsorption equilibrium (equilibrium-controlled) or the adsorption rate (kinetically-controlled) between the components to be separated. During the adsorption step, the less strongly (or the less rapidly) adsorbed species pass through the bed and are collected as a refined product. The extracted product consists of the adsorbed species removed during the desorption step.

The main difficulties which arise in PSA design are the complexity of the simulation and the identification of cyclic steady state (CSS). The simulation requires the development of specialised algorithms for solving PSA bed models, described by partial differential-algebraic equations (PDAEs) in space and time, derived from heat, mass and momentum balances plus transport and equilibrium equations.

Often, the mass transfer model is based on the linear driving force (LDF) approximation (Jiang *et al.* 2003, Ko and Moon 2002). Assuming a linear driving force for mass transfer between the fluid and solid phases simplifies the model by eliminating the need to describe concentration profiles within the solid. The full diffusion model and the LDF approximation are equivalent when the profile of concentration within the particle is parabolic, a condition which holds for slow cycles. Unfortunately, the LDF approximation fails to describe the dynamics of fast cycle operations, needed to reduce capital and operating costs. A PSA cycle is fast when the diffusional time constant R^2/D, where R is the radius of the particle and D the diffusivity, is large with respect to the cycle time. At fast cycles the concentration profile is flat at the core of the particle but varies rapidly in the outer region of the adsorbing particle (Ahn and Brandani 2005). In this case, a detailed diffusion model is necessary which requires the solution of coupled nonlinear partial differential and algebraic equations in time and space.

Different methods have been developed to detect the performance at CSS. In the successive substitution method (Sankararao and Gupta 2007, Todd *et al.* 2003), the CSS is determined by repeated dynamic simulations, starting from a given initial state until CSS is reached. This is computationally demanding. Alternatively, the equations describing the system can be simultaneously discretised in the space and

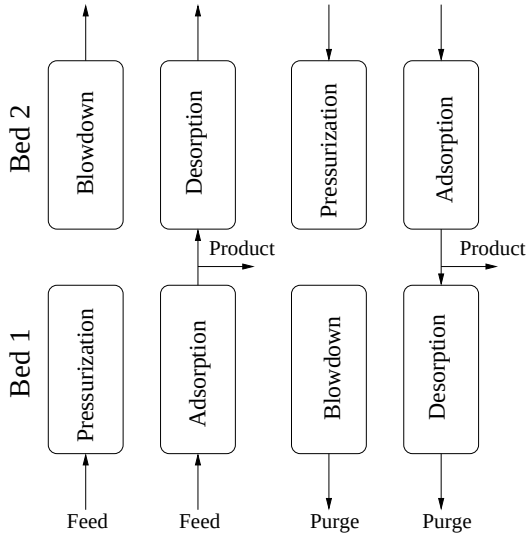


Figure 1. 2 beds/ 4 steps Skarstrom cycle

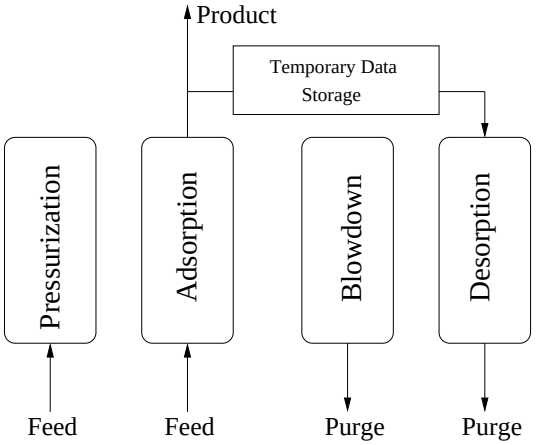


Figure 2. Model of PSA cycle to allow faster convergence to cyclic steady state

time and the periodicity condition is imposed as a constraint (Ko and Moon 2002, Nilchan and Panthelides 1998). However, this simultaneous approach can suffer from convergence issues (Nilchan and Panthelides 1998). Alternatively, the problem of detecting CSS can be treated as an optimisation problem itself (Ding *et al.* 2002, Jiang *et al.* 2003), as illustrated in the next paragraph. We adopted a "unibed" approach to accelerate convergence to CSS, which will be illustrated in section 4.1.

3. Optimisation strategies

A number of approaches for solving the PSA design problem exists and can be grouped into the following categories (Biegler *et al.* 2004): 1) simplified optimisation methods, 2) black box optimisation, 3) equation-based optimisation and 4) simultaneous tailored approximation. Simplified optimisation methods (Smith and Westerberg 1991, Lewandowski *et al.* 1998) consist of the development of a simple PSA model with fine-tuning using pilot-plant data, actual plant data or a more detailed model. Although such models are often useful, case-by-case studies are hard to transfer among different PSA systems.

In the black box optimisation approach (Kvamsdal and Hertzberg 1997, Rajasree and Moharir 2000), the optimiser may only use directly the objective function

value and possibly the constraints. If gradient information is required, this must be estimated numerically at potentially high computational cost.

In the third method, the bed equations, the objective function, inequality constraints and the CSS condition are completely discretised (Nilchan and Panthelides 1998) in time and space and solved simultaneously. The approach is efficient for simple systems but convergence problems arise with complex systems and steep concentration waves. In these cases a fine spatial and temporal grid is required, but this leads to a non-manageable optimisation problem. Besides, for complicated models, the solver may fail due to error accumulation caused by complete discretisation (Ko *et al.* 2003).

The simultaneous tailored optimisation (Ding *et al.* 2002, Jiang *et al.* 2003) is a combination of black box optimisation and equation-based optimisation where the bed model is treated as a black box but the CSS condition is incorporated as a constraint directly. Since satisfaction of the CSS is required only at the optimum, the time-consuming CSS loop is eliminated. However, the optimiser may have difficulties in finding feasible design points, which satisfy the CSS constraint.

The design of PSA processes has also been handled by heuristic analysis (Jain *et al.* 2003) to develop easy-to-use rules to be used as a guide in PSA process design.

All of the above consider only single objective design problems. However, the design problem is inherently multi-objective and only two previous papers have attempted tackling the multi-objective problem (Ko and Moon 2002, Sankararao and Gupta 2007). A multi-objective optimisation strategy based on the summation of weighted objective functions (SWOF) method has been proposed by Ko and Moon (2002). The SWOF method minimises a combination of objectives, but it works only if the Pareto curve is convex. In our case, this cannot be guaranteed as not only is the model non-convex, but the feasible space is also non-convex (cf. section 4.2). A "modified MOSA-aJG" method has been used with a detailed diffusion model for mass transfer (Sankararao and Gupta 2007). MOSA-aJG is an expansion of the simple simulated annealing (SSA) for multi-objective optimisation, where "MOSA" stands for multi-objective simulated annealing, and "aJG" is a jumping-gene (JG) adaptation. CSS is identified using a successive substitution method and the simulation requires 24 h on a Pentium 4, 2.99 GHz computer for each cycle. Although the optimisation method used (simulated annealing) is a stochastic method, the results in the article are not accompanied by any statistical analysis on the performance of the optimiser and it is therefore impossible to know what to expect from this method in general.

The paucity of alternatives for solving the multi-objective design problem for fast cycle PSA needs is addressed in this paper. The performance of a custom multi-objective genetic algorithm is evaluated. The case study is the separation of air for N_2 production.

4. Case Study

4.1. *The model*

The problem of interest is the separation of air on an activated carbon sieve for N_2 production using a 2 bed PSA process. The configuration investigated is the 4-steps Skarstrom cycle. This separation is kinetically controlled by microporous diffusion. Fast cycles are necessary to exploit the different adsorption rates of the components to produce N_2. The design problem is the simultaneous optimisation of the recovery and purity of N_2 in the product stream. One constraint is the maximum value of pressure that can be reached. Another constraint is the objectives can only

be evaluated when the operation has reached CSS. Identifying the CSS requires the development of a model for the dynamic simulation of the process if we are to avoid convergence problems, as noted above.

The main assumptions are perfect mixing, ideal gas phase, isothermal operation, spherical adsorbent particles and negligible pressure drop. The above assumptions are reasonable for air separation on a carbon molecular sieve (Hassan *et al.* 1986, 1987).

A series of 6 CSTRs (continuous stirred tank reactors) has been used to simulate each of the 2 beds. A CSTR is an ideal continuous reactor whose content is continuously stirred and, hence, homogeneous. As a consequence, the product of a CSTR has the same composition as the fluid inside the reactor. A more realistic model for an adsorption column is the dispersed plug flow reactor, where the composition of the fluid changes from one point to another. This more complex behaviour can be modelled using a series of CSTRs (Levenspiel 1962). This simplification reduces the computational requirements since the mass balance of the fluid phase for a CSTR is not dominated by a convective term as in the case of a plug flow reactor (Cruz *et al.* 2005). The fluid mass balance in a fixed bed reactor with dispersed plug flow is given by:

$$-D_L \frac{\partial}{\partial z}\left(c\frac{\partial y_i}{\partial z}\right) + \frac{\partial}{\partial z}(cv_i) + \frac{\partial c_i}{\partial t} + \frac{(1-\varepsilon)}{\varepsilon}\frac{\partial \overline{q_i}}{\partial t} = 0 \tag{1}$$

where D_L is the Fickian axial dispersion coefficient, v is the interstitial fluid velocity, $y_i(t)$ is the molar fraction of species i, c and c_i are the concentrations of the fluid phase and of the species i respectively, ε is the bed voidage, and $\overline{q_i}$ is the average concentration of species i in the solid.

The first term of eq. 1 takes into account axial dispersion, the second term takes into account advection. The third and fourth terms express the build up in the fluid and the solid phase respectively. The mass balance in a CSTR is given by:

$$\frac{\varepsilon V}{R_g T}\frac{d(y_i\,P)}{dt} + (1-\varepsilon)V\frac{d\overline{q}_i}{dt} = F_{in}\,y_{i,in} - F_{out}\,y_i \tag{2}$$

where V is the volume of the reactor, T the temperature, P the pressure, R_g is the universal gas constant. F_{in} and F_{out} and the inlet and outlet flow rates respectively, while $y_{i,in}$ is the molar fraction of species i in the inlet.

In a CSTR there is no diffusion in the z direction as perfect mixing is assumed, so the axial diffusive term $-D_L\frac{\partial^2 c_i}{\partial z^2}$ is not present in eq. 2. The advection term $\frac{\partial}{\partial z}(cv_i)$ is substituted by the finite difference $(F_{in}\,y_{i,in} - F_{out}\,y_i)$. Eq. 1 is usually a parabolic or hyperbolic equation (when the diffusive term is negligible) controlled by advection, and very hard to solve numerically (Cruz *et al.* 2005). Using eq. 2, in lieu of eq. 1, allows a faster solution of the mass balance equation.

The model of each CSTR is summarised in Table 1 where $i, j \in \{O_2, N_2\}$.

The variables are the solid concentration $q_i(t,r)$, the molar fraction $y_i(t)$, the gas concentration $c_i(t) = \frac{Py_i}{R_g T}$, the outlet flow rate $F_{out}(t)$, and the pressure $P(t)$. The constants are the temperature, T, the radius of the adsorbent particle, R_p, the universal gas constant, R_g, the saturation limit, $q_{i,s}$, the Langmuir constant, b_i, the constant intrinsic mobility, D_{i0}, the bed voidage, ε, the liquid film mass transfer coefficient, k_f, the volume of the reactor, V, the inlet flow rates F_{in}, and $y_{i,in}$, the molar fraction in the inlet flow. Data for the simulation are from (Ruthven *et al.* 1993). The system of PDAEs has been discretised in space using the central finite difference operator and the adaptive $4th/5th$ order Runge-Kutta method is used to

Table 1. Model equations for N_2 separation in a CSTR unit.

Mass balances:	$\frac{\varepsilon V}{R_g T}\frac{d(y_i P)}{dt} + (1-\varepsilon)V\frac{d\overline{q}_i}{dt} = F_{in}\, y_{i,in} - F_{out}\, y_i$	
Diffusion:	$\frac{\partial q_i}{\partial t} = \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 D_i \frac{\partial q_i}{\partial r}\right)$	(3)
Continuity:	$\sum_i y_i = 1$	
Boundary conditions:	$\left(\frac{\partial q_i}{\partial r}\right)_{r=0} = 0$	(4)
	$\left(\frac{\partial q_i}{\partial r}\right)_{r=R_p} = k_f(c_i - c_{i_{R_p}})$	(5)
Equilibrium:	$q_i(R_p) = \frac{q_{i,s} b_i c_{i,R_p}}{1 + \sum b_i c_{i_{R_p}}}$	(6)
Initial conditions:	$y_i = y_{i,0}$	
	$q_i = q_i^*(y_{i,0})$	

solve the resulting system of ordinary differential equations. Ten points have been used for the discretisation of the radius.

The mass balance of the solid particle is expressed by eq. 3. Although it might be possible to use the LDF approximation by fitting the parameters of the model with experimental data (Ruthven *et al.* 1993), the resulting simulation would not be representative of the operation outside the range of parameters experimentally investigated. Accordingly, we use a Fickian equation to describe the mass balance in the particle (eq.3). The diffusivity D_i has been described by

$$D_i = \frac{D_{i0}}{1-\theta_i-\theta_j}\left((1-\theta_j) + \theta_i \frac{\partial q_j/\partial r}{\partial q_i/\partial r}\right) \tag{7}$$

where $\theta_i = \frac{q_i(t,R_p)}{q_{i,s}}$. This expression was proposed by Hagbood (cf. Ruthven *et al.* (1993)) and applies to binary Langmuir systems where the saturation limits of the two species are the same: $q_{A,s} = q_{B,s}$.

The boundary condition, eq. 4, is the "symmetry condition". It implies the symmetry of the concentration profile with respect to the center of the particle (r=0), or in other words a zero-flux at the center of the particle. This situation holds because of the symmetry of the flux of matter from all the directions towards (in the case of adsorption) and from (in the case of desorption) the center of the particle. The boundary condition at the surface of the particle, eq. 5, specifies that the gradient of the concentration of the adsorbate phase at the surface be proportional to the difference between the concentration of the fluid phase in the bulk and the value in equilibrium with the surface concentration in the solid. The equilibrium is described by the Langmuir isotherm, eq. 6.

During the pressurisation step, the operating pressure increases from P_{low} to P_{high}, while it decreases from P_{high} to P_{low} during the blowdown step. The lower pressure, P_{low} is a process specification. During the constant pressure steps (adsorption at

high pressure, and desorption at low pressure) the unknowns are the concentrations of the two species in the solid and gas phases, q_i and y_i respectively, and the outlet flow rate F_{out}. During blowdown, the same set of unknowns (q_i, y_i, F_{out}) holds, but the value of the inlet flow rate is set to be zero. The dependence of the pressure with time during blowdown was described by an exponential function, eq. 8, where $\overline{P}$ and $\bar{t}$ are the average values of pressure and time over the blowdown operation, and k is equal to 10 in order to have a steep change in P. Eq. 8 reproduces well the fact that pressure changes are usually fast, and it is more realistic than a linear trend with time (Ruthven *et al.* 1993). The dependence is

$$P(t) = \overline{P} + (P_{low} - P_{high}) \left\{1 - exp\left[k(t - \bar{t})\right]\right\} \tag{8}$$

The balances in the pressurisation steps are described by the same set of equations, but F_{out} is known and is equal to zero, while the pressure has to be calculated. P_{high} is a function of the duration of the pressurisation step, t_{press}, and of the inlet feed rate, F_{in}, and it will be determined by the mass balances which describes the operation.

We adopted the method to achieve a faster convergence to CSS proposed in (Kumar *et al.* 1994): since at CSS each bed undergoes identical process steps in a sequential manner, it is possible to simulate a multibed cycle using only one bed. As in Kumar *et al.* (1994), we store in temporal effluent arrays the flow and composition from a process step to use it later when the bed is undergoing the appropriate process step. The resulting model is illustrated in fig.2. Although the method does not provide a correct description of the transition to CSS, it represents a powerful tool for the design of the operation since it allows a faster and correct prediction of the performance at CSS. This reduces the computational requirements sufficiently to allow the use of successive substitution for detecting the CSS.

4.2. *The design problem*

The multi-objective design problem is to identify the trade-off between the recovery of N_2 achieved and the purity of the N_2 actually recovered. There are 4 design parameters: $t_c \in [15, 250]$ s is the cycle time which defines the duration of the four steps, r_S the split ratio (described below), $F_{in} \in [15, 100]$ mol/s the inlet flow rate and S the schedule (also described below). The constraints for the optimisation problem are the evaluation of the PDAEs to CSS and a maximum value of the high pressure, $P_{high} < 7$ atm.

The recovery of N_2 is defined as the fraction of N_2 in the product stream, F_{prod}, with respect to the amount fed to the system over the cycle:

$$\text{recovery}_{N_2} = \frac{2 * \int_{t_{ads}} (F_{prod}\, y_{N_2})\ dt}{\int_{t_c} (F_{in}\, y_{N_2,in})\ dt}$$

where $y_{N_2,in}$ is the molar fraction of N_2 in the inlet flow, F_{in}; t_{ads} is the duration of the adsorption step.

The purity is given by the concentration of N_2 in F_{prod} and is evaluated by the following integral:

$$\text{purity}_{N_2} = \frac{\int_{t_{ads}} (F_{prod}\, y_{N_2})\ dt}{\int_{t_{ads}} F_{prod}\, dt}$$

The product stream flow rate, F_{prod}, is the fraction of the outlet flow rate with-

such

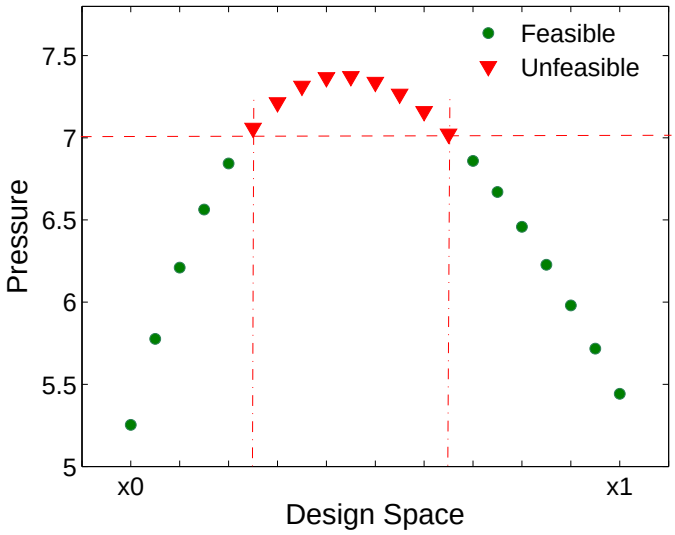


Figure 3. As we move from point x_0 to point x_1 in the design space, the pressure assumes values which violate the constraint on the maximum pressure allowed, hence determining a non-convexity in the feasible design region.

drawn during the adsorption step (the *raffinate*). This fraction is expressed by the split ratio, $r_S \in [0.20, 0.85]$, for the splitter which separates the raffinate into the product stream and the purge to be sent to the other bed:

$$F_{prod} = r_S F_{out,ads}$$

The schedule, $S \in [0, 50]\%$, expresses the fraction of the cycle time occupied by each of the adsorption and desorption steps: $t_{ads} = t_{des} = St_c$, where the subscripts "ads" and "des" refer to the adsorption and desorption steps, respectively. Consequently, $t_{press} = t_{blow} = (1 - S)t_c$, where "$press$" and "$blow$" refer to the pressurisation and blowdown steps, respectively.

A preliminary investigation of the design and objective space has been carried out to identify the most appropriate approach to the solution of the optimisation problem. This analysis showed that the objective function is non-smooth and non-convex. Interestingly, also the design space in non-convex, as shown below. Let us consider two design points, $x_0 = [231.9, 0.84, 65.3, 44.5]$ and $x_1 = [123.2, 0.33, 29.9, 21]$. As we move from x_0 to x_1 in the design space, the values of all the design variables (t_c, r_S, F_{in} and S) decrease, while t_{press} increases from 12.75 to 35.7 s. As F_{in} decreases, the value of P_{high} decreases as well. Conversely, the increase of t_{press} makes the pressure increase. As a result, the value of P has a maximum as we move in the space from x_0 to x_1. The value of this maximum is 7.37, above the P_{max} imposed as a constraint. Hence, there will be a non-feasible region in the area around this maximum which corresponds to a non-convexity in the design space, as shown in fig. 3.

Another important characteristic of the problem is the computational demand of the simulation. The model proposed in 4.1 is detailed enough to describe the behaviour of the PSA operation yet less computationally demanding than a full column simulation. However, the evaluation of the objectives still makes the optimisation problem complex, possibly requiring several minutes of computation depending on the resources available and the particular case study considered. To give an indication of the computational demands for this design problem, an objective function

evaluation, based on convergence to cyclic steady state requiring, on average, 6 cycles, takes approximately 90 s on a 2.20 GHz Pentium processor.

5. A multi-objective genetic algorithm (MOGA) procedure

The preliminary investigation of the design problem showed that the profiles of the objective functions are non-smooth and non-convex. The design space is non-convex. The computational requirements of the simulation model, together with the non-smooth behaviour observed, preclude the use of gradient based optimisation methods. Numerical differentiation would be required, leading to a loss of accuracy and to a deterioration in the performance of the optimiser (Biegler *et al.* 2004).

Accordingly, our initial attempts to solve the design problem were based on direct search optimisation methods (DSMs), as they are derivative-free optimisation algorithms designed for non-smooth optimisation problems.

For single objective optimisation, the results were promising (Fiandaca *et al.* 2007). However, the performance of DSMs for the multi-objective optimisation problem is very poor, as illustrated in section 6.1. Furthermore, for the multi-objective design problem, the use of single objective DSMs is problematical due to the need for a weighted combination for the objective function. As the number of objectives to be considered simultaneously increases, the number of discrete optimisation problems that must be solved using a weighted objective function increases exponentially. An alternative to direct search or gradient based methods is the class of evolutionary stochastic optimisation algorithms. These methods are similar to DSMs in that they do not require gradient information. They are, however, able to generate a Pareto front directly. Genetic algorithms have been extensively applied for the multi-objective optimisation of simulated moving beds (SMBs), a continuous countercurrent chromatographic separation processes (Yu *et al.* 2003, Zhang *et al.* 2003). SMBs processes have similar complexity to a PSA process, described by PDAEs with comparable nonlinear behaviour. Therefore, we consider a genetic algorithm for tackling the multi-objective design problem. The choice of a genetic algorithm is also motivated by our previous positive experience with simulation based black box models in process design (Barakat *et al.* 2008).

The implementation of a genetic algorithm for any new problem requires the definition of the following elements: a representation of (hopefully) feasible solutions, the crossover and mutation operators, a selection procedure together with an appropriate fitness function, and the properties of the evolution of the population. For the MOGA used in this work, the solution representation consists simply of real-valued design variables. A multi-point crossover operator is defined and mutation consists of selecting a single design variable and assigning it a randomly chosen value from the domain for that variable. We have used tournament selection with a tournament size of 2. The population policy is one of replacement with elitism. The key property of the MOGA used, specific to this problem, is the definition of the fitness function.

The aim, for any multi-objective optimisation problem, is to identify the Pareto front. An approximation to this front is a finite sized Pareto set. The goal, for our specific design problem, is to generate as broad a Pareto set as possible, stretching the end points to identify the extremes of the trade-offs between the criteria to give the engineer sufficient information to make design decisions early in the design process (Zitzler *et al.* 2004). This desire has been addressed in a number of ways in the literature. Schaffer (1985) presented a Vector Evaluated Genetic Algorithm (VEGA). In this algorithm, a population is subdivided into subgroups, governed by different objective functions. Since the search directions of VEGA are exclusively parallel to the axes of the objective space, the algorithm is able to find mainly the extreme

solutions on the Pareto front. A Niched Pareto algorithm, proposed by Horn *et al.* (1994), incorporates the concept of Pareto domination in the selection procedure and spreads the population along the Pareto front by applying a niching pressure. Murata and Ishibuchi (1995) presented a MOGA with various direction search, which uses a weighted sum of multiple objective functions to formulate a scalar fitness function. They compared the MOGA with the VEGA (Schaffer 1985) and the Niched Pareto algorithm (Horn *et al.* 1994), showing that the VEGA outperforms the other two algorithms in the detection of points belonging to the extremes of the Pareto front, while failing to detect points at intermediate positions. Deb (2001) presented a steady-state multi-objective evolutionary algorithm (MOEA) which attempts to maintain the spread while converging to the Pareto front. However, no proof of convergence for the method has been provided. Subsequently, Laumanns and co-workers (Laumanns *et al.* 2002) proposed an algorithm which caters for convergence to the Pareto optimal set while aiming to cover the whole range of non-dominated solutions, based on the concept of ϵ-dominance. Laumanns *et al.* (2001) analysed the performance of different algorithms using a volume based approach (cf. Zitzler and Thiele (1999)), with some modification: a reference volume between the origin and an utopia point – defined as the profit sums of all items in each objective – is taken into account. The aim is to minimise the fraction of that space which is not dominated by any of the final archive members. This is considered by the authors the most appropriate scalar indicator since it combines both the distance of solutions (toward the utopian trade-off surface) and the spread of solutions. Density based selection can further improve the algorithm performance by a broader distribution of solutions along the trade-off surface. Herrero *et al.* (2007) presented the ϵ-MOGA method, designed to achieve a reduced but well-distributed representation of the Pareto front. The algorithm adjusts the limits of the Pareto front dynamically, preventing the loss of solutions which belong to the ends of the Pareto front.

Two key requirements can be extracted from the previous works. First, the need for an elitism operation which ensures that the extreme points are not lost in the evolutionary procedure. Second, that there be a driving force to extend the Pareto set outwards. Our approach is based on elitism applied to the whole Pareto set (subject to size constraints mentioned below) combined with a fitness function chosen to emphasise points that may have genetic material that could extend the Pareto set at the extreme ends.

Elitism, from generation to generation, is implemented by copying over the complete Pareto set from the old population to the new. The only qualification is that if the Pareto set is the whole population, only half of the members, chosen randomly, of the Pareto set are copied over to the new population intact.

The fitness of a design point in the population is based on a modified measure of the distance of that point to the approximation to the Pareto front. Solutions with lower values are fitter. For our design problem, with two criteria, the distance of a dominated point to the Pareto front approximation is the minimum of the distance of that point to each of the points in the Pareto set and the distance to infinite projections from the end points parallel to the two axes. The aim of the latter is to give emphasis to those solutions which may be far from any points in the current Pareto front approximation but which may help in generating new solutions that would extend the breadth of the Pareto front. The procedure to assign the fitness is illustrated in fig.4.

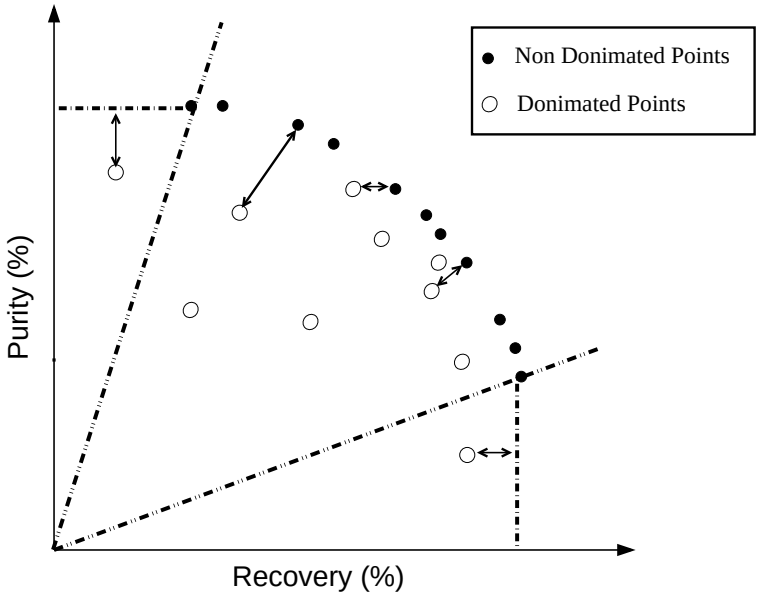


Figure 4. Illustration of the procedure to assign the fitness.

6. Analysis of the MOGA

The design problem has been attempted a number of times to assess the average performance. All codes have been run in MATLAB 7.1.

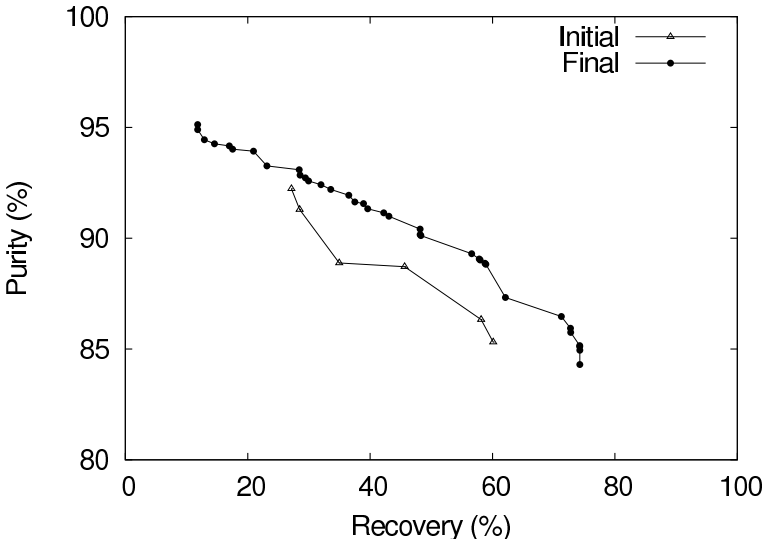
The data presented in fig. 5 have been generated using the values summarised in Table 2. Fig. 5(a) illustrates the initial and final Pareto sets for a typical run. The initial set is from the initial randomly generated population and, in this case, consists of 6 points. The final Pareto set has 36 points, evenly distributed across the front, out of a population of 40. We see an improvement when compared with the initial set, across the front, not just in objective function values but also in the breadth of this approximation to the front.

Table 2. Parameters used in MOGA

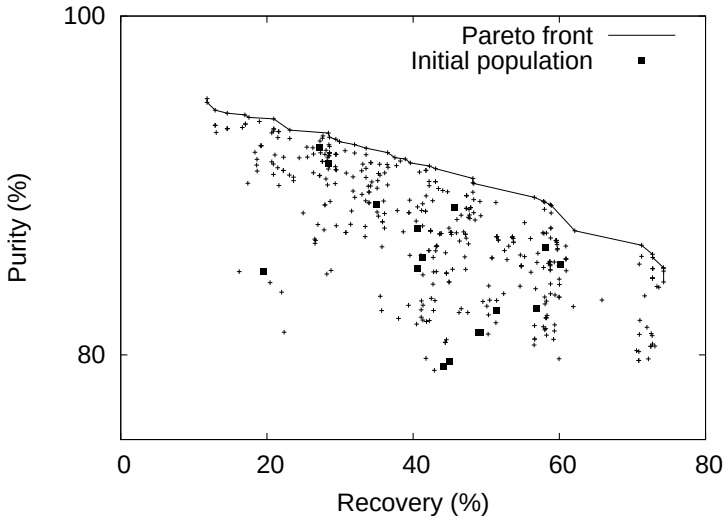
Crossover rate	cr	0.7
Mutation rate	mr	0.1
Population size	n	40
Number of generations	ng	50
Tournament size		2

The effectiveness of the multi-objective optimisation procedure in searching the objective function space is shown in fig. 5(b). The figure shows a fairly even distribution of points across the two criteria with an increasing density towards the Pareto front. More importantly for our design goals, when the full set of design points generated is compared with the initial random population generated by the MOGA procedure, the fitness function defined above has been effective in generating a broad Pareto front which helps the engineer gain a better understanding of the trade-offs involved.

As genetic algorithms are stochastic, the assessment of the performance of the MOGA requires a statistical analysis of the results. For every given set of parameters, the average Pareto fronts and the standard deviations have been determined by performing a Gaussian progress regression (Rasmussen and Williams 2006) of 10 Pareto fronts. We obtained a good fit to our data (the Pareto fronts generated) by using as a covariant function the sum of a Matern covariance function with a



(a) Representative initial and final Pareto sets.



(b) A graphical representation of all the design points generated during the search with the initial population highlighted and the final Pareto front approximation drawn.

Figure 5. Analysis of the performance of the multi-objective genetic algorithm. All the parameters used are listed in Table 2.

shape parameter of 3/2 and an independent noise. The process is illustrated in fig. 6, generated using the set of parameters in Table 2. This analysis allows us to gain an insight on the effect of the different parameters of the MOGA on both the average Pareto front obtained, as well as on the dispersion of the data. In the course of the paper, only average Pareto fronts obtained through regression will be shown, unless the standard deviation is significant for the analysis. The parameter values used are those in Table 2, unless otherwise noted. This statistical approach has been used to compare the performance of MOGA and DSMs (section 6.1), and to study the effect of the parameters of the MOGA on the resulting Pareto front (section 6.2).

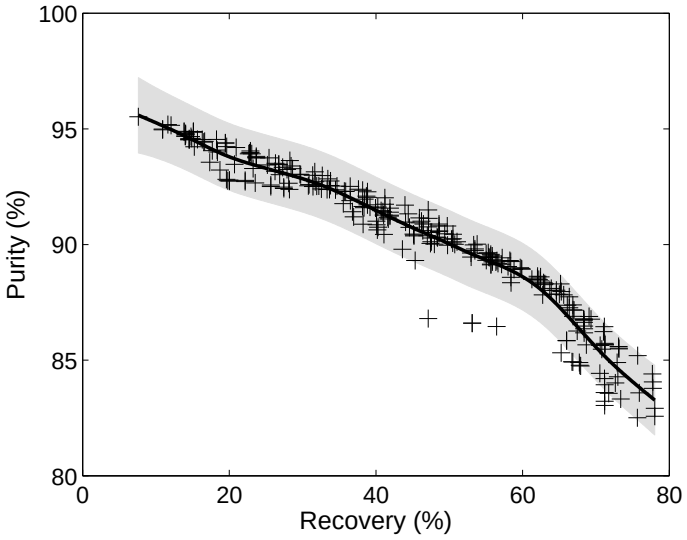


Figure 6. Average performance of 10 optimisation runs. All the parameters used are listed in Table 2.

6.1. *Comparison between MOGA, random search and Direct Search Methods*

The capability of the MOGA to efficiently detect the Pareto front has been assessed by comparison with random search and DSMs. The aim of the comparison with a random search algorithm is to verify that the evolutionary procedure implemented in the MOGA proposed (i.e. definition of fitness evaluation, selection, mating and mutating procedures) is effective. A random search algorithm has been included also in the comparison among evolutionary algorithms proposed by Zitzler *et al.* (2000).

The average Pareto fronts obtained with MOGA (averaged over 10 runs, using approximately 600 evaluations each) have been compared to the average of 10 sets of non-dominated points, obtained from 10 sets of 600 randomly generated points. The Pareto front obtained with the MOGA shows a big improvement with respect to the homologous random one, and a larger number of non-dominated points is detected by the MOGA. The performance of the two methods in the high purity region is comparable, with a slight dominance of the solutions found by the random search: this shows that it is easier to detect solutions with high purity/low recovery and the impact of the evolutionary procedure in this area is less crucial.

We also compared the performance of the custom MOGA and of some standard DSMs for the solution of our multi-criteria problem. When the objective function is non-smooth as in the optimisation problem of interest, two possible approaches are direct search methods and evolutionary algorithms (cf. section 5). We compared the performance of these two classes of optimisers to verify that evolutionary algorithms are effectively the best option for the case of interest. Although a better performance of MOGA could be considered predictable, there are cases reported in literature (Meza *et al.* 1996) where the performance of a genetic algorithm and a DSM is comparable.

DSMs are derivative-free algorithms, which use exclusively function values in their attempt to find the optimum. At each iteration DSMs explore the objective function in a linearly independent set of n directions, as an alternative to information coming from the gradient (Lewis *et al.* 2000). DSMs were originally designed to solve single objective problems. Hence, it is necessary to combine the two objectives to be maximised, i. e. recovery and purity, into a single objective function. The combined expression must allow a different weight to be given to each objective so that differ-

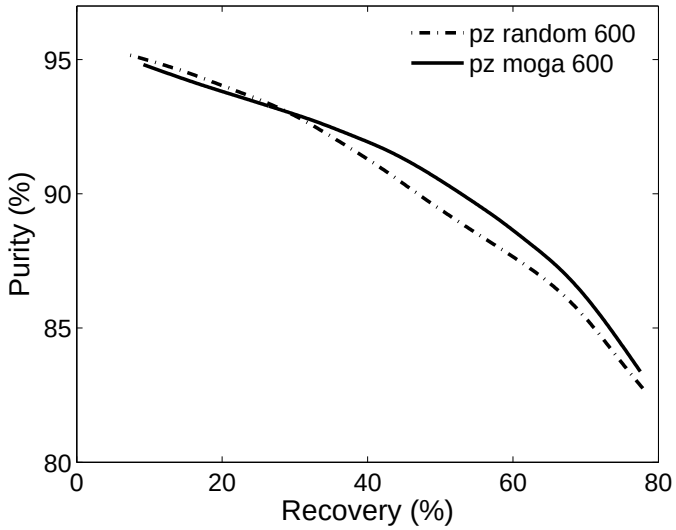


Figure 7. Comparison between the Pareto front obtained with 600 random evaluations and the Pareto front obtained with the same number of evaluation by MOGA (with ng=30 and n=40 and other parameters as in Table 2).

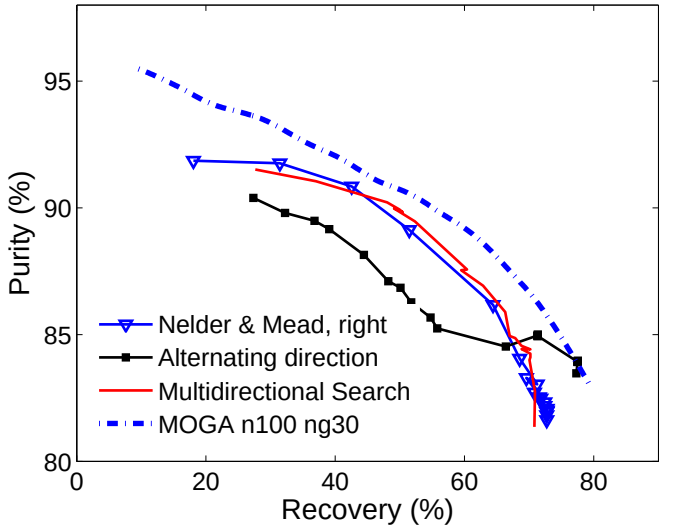


Figure 8. Comparison of the average performances of MOGA and DSMs

ent trade-offs between the two variables can be optimised. The composite criterion to be maximised has been expressed as:

$$f(\overline{x}, \lambda) = recovery_{N_2}(\overline{x}) - \lambda * [recovery_{N_2}(\overline{x}) - purity_{N_2}(\overline{x})] \tag{9}$$

where $0 \leq \lambda \leq 1$, and $\overline{x}$ represents the vector of design variables. The methods used were Nelder and Mead, with both regular and right simplex, alternating directions and multidirectional search (Kelley 1999). As DSMs are deterministic methods, we used a statistical procedure to assess the performance of each direct search algorithm different from the procedure used for the MOGA. We randomly selected ten points within the feasible region of the design problem, e.g. $\overline{x}_{0,i}$, where $i = 1, ..., 10$. Starting from these points, each of DSM has generated a Pareto front by optimising the function f, eq. 9, with λ varying from 0 (maximisation of recovery) to 1 (maximisation of purity), with regular increment of 0.05. The performance of every

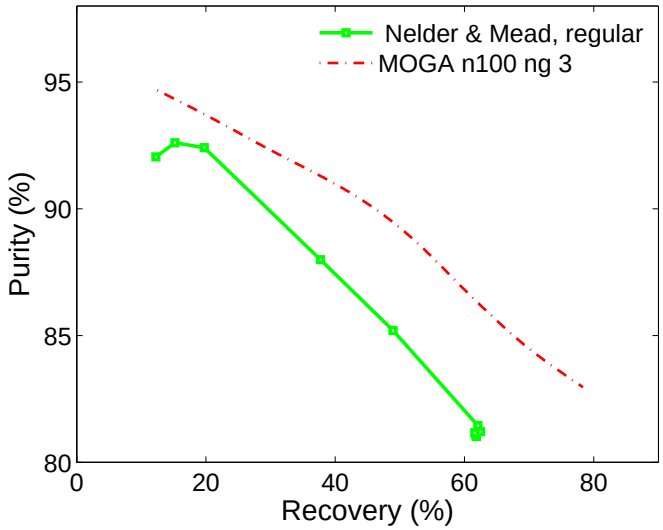


Figure 9. Comparison of the average performances of MOGA and Nelder & Mead, regular simplex

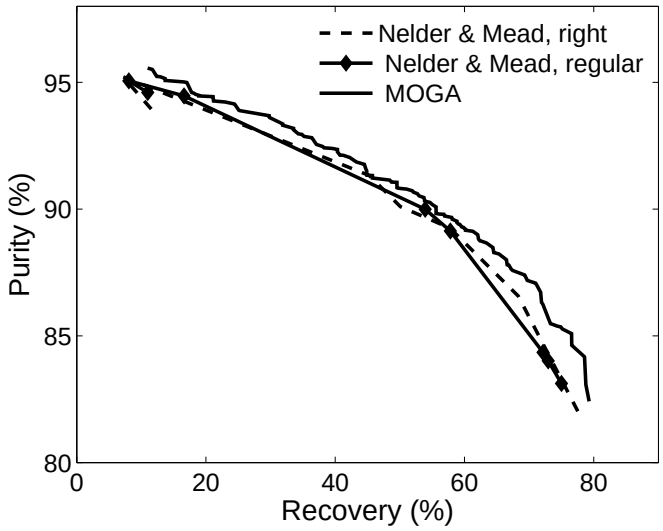


Figure 10. Comparison of the best performances of the DSMs and MOGA (with n 100, ng 30 and other parameters as in Table 2).

method is then given by the average of the 10 Pareto fronts it generated, and the relative standard deviation.

As shown in fig. 8 and fig. 9, the average performance of MOGA is by far the best compared to those of all the four DSMs used. The comparison has been made between the average performance of each method using a comparable number of iterations. This information is summarised in Table 3. In all cases, MOGA achieved better results using a lower number of function evaluations, which demonstrates the efficiency of the algorithm. In fact, higher values of purity (for a given recovery) have been detected, and a larger breadth of the Pareto curve has been achieved by MOGA. The distribution of points along the Pareto fronts generated by alternating direction methods and multidirectional search methods is quite uniform, while the Nelder & Mead methods have generated more points in the high recovery region. The comparison among the best results achieved by the different methods shows that all the algorithms find an approximation to the effective Pareto front, but the

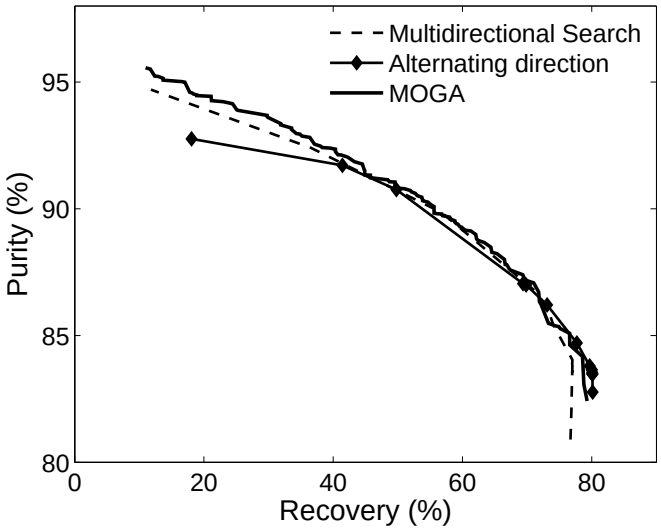


Figure 11. Comparison of the best performances of the DSMs and MOGA (with n 100, ng 30 and other parameters as in Table 2).

Table 3. Total number of function evaluations used by the different methods to produce the Pareto fronts of figures 8,9,10 and 11. The parameters used by MOGA for the comparison are as in Table 2 unless otherwise stated.

Method	N Eval per Pareto
Nelder & Mead regular	286
Nelder & Mead right	1,916
Alternating Directions	12,190
Multidirectional Search	4,000
MOGA n 100, ng 30	1,800
MOGA n 100, ng 3	250

approximation of MOGA is more complete (figs. 10, 11). The difference between the average and the best performance of DSMs revealed the sensitivity of DSMs to the starting point. As described before, we started each method from 10 points randomly chosen within the design space. We obtained better results in the high recovery region from starting points already belonging to that area, and the same occurred for the high purity area. This means that a pre-knowledge of the design space would be required to obtain a good approximation of the Pareto set with any of DSMs, which represents a significant drawback with respect to MOGA.

6.2. *Impact of algorithm parameters*

The parameters for the MOGA, e.g. the population size, mutation and crossover rates and the number of generations, will have an impact on the quality of the approximation to the Pareto front that we obtain. We changed the parameters of MOGA one at the time, while keeping the others to the values summarised in Table 2. As explained in section 6, only average Pareto fronts obtained through regression will be shown, unless the standard deviation is significant for the analysis.

Fig. 12 shows that as the number of generations (ng) increases, the Pareto front improves till convergence, which is reached in 35 generations.

The effect of the mutation rate (mr) is illustrated in fig. 13: as the mutation rate increases we get better results in the central area of the Pareto front, while the effect is not clear at the extremes of the curve. We do not observe a direct effect of the mutation rate on the deviation of the data.

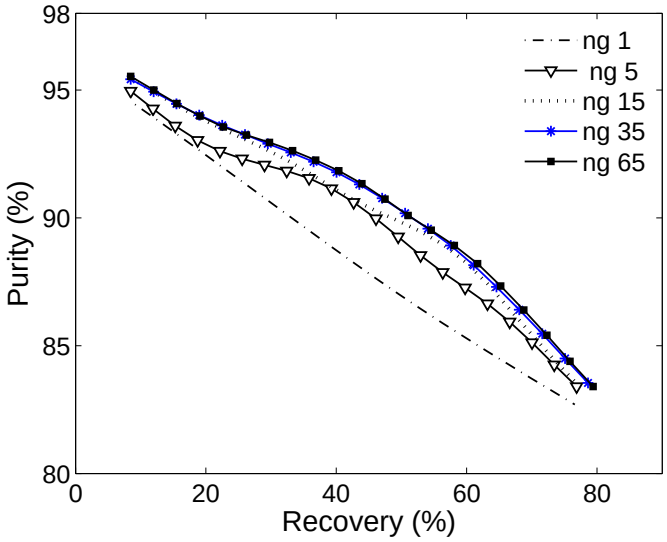


Figure 12. Influence of the number of generations (ng) allowed. Convergence is reached in 35 generations.

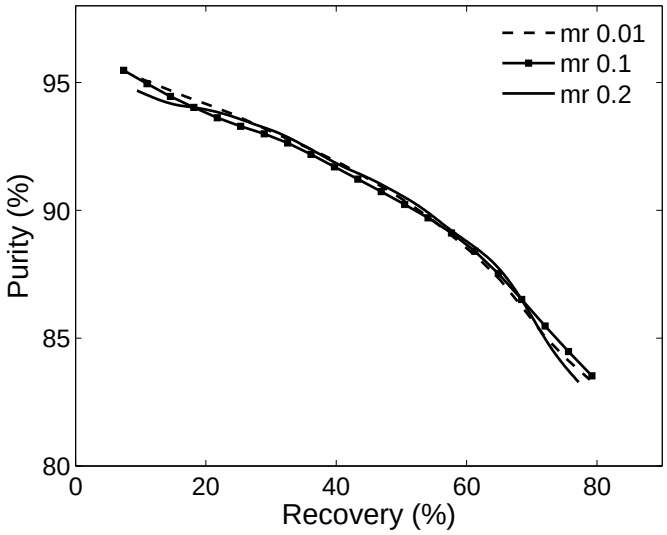


Figure 13. Influence of the mutation rate (mr), with mr=0.01, mr=0.1, mr=0.2.

The tournament size (ts) has no effect on the final performance of the algorithm in the range of values investigated (i. e. ts=2, 4, 6). However, a slight difference in the rate of convergence is noticeable in fig. 14, as at the same number of generations the Pareto front detected with tournament size 4 dominates the other, and is more extended in the high recovery region.

The most influential parameter appears to be the size of the population (n) used: in fig. 15 as the size of the population increases the Pareto front is pushed further, while the standard deviation of the data decreases (fig. 16). As shown in fig. 17, increasing the size of the population we are able to investigate more accurately the search space: although the same trends are indicated in fig.17 (a) and (b), the clumping of solutions is more evident in fig.17 (a), whereas the solutions found with a larger population are more distributed within the search space. However, such an

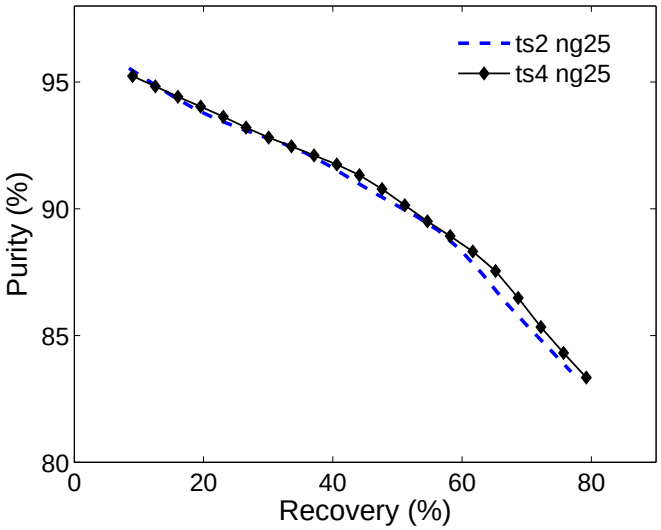


Figure 14. Average Pareto front produced with tournament size 2, 4 at 25 generations.

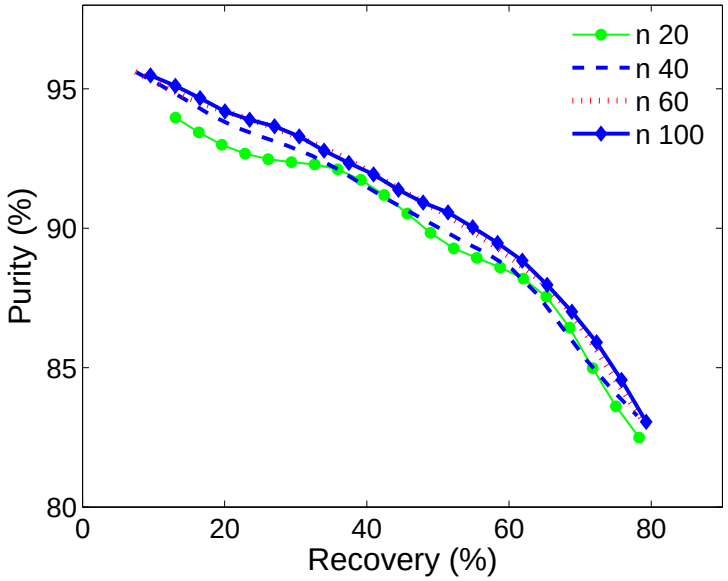


Figure 15. Influence of the population size (n), with n=20, n=40, n=60, n=100

improvement is counterbalanced by a higher computational requirement: the number of function evaluations needed grows from an average of 950 with $n = 40$, to 1410 with $n = 60$, to 2480 with $n = 100$ and $ng = 50$.

7. Design problem analysis

Analysis of the Pareto sets identified in all the runs show clear clumping of the solutions. This manifests itself in a parallel co-ordinate visualisation of the Pareto set shown in fig.17. In 17(b), although there are 72 solutions comprising the Pareto set, there are only 9 distinct values of the cycle time, t_c, and 8 for the feed flow rate, F_{in}. To a lesser degree, the same clumping is observed for the split ratio, r_S, and

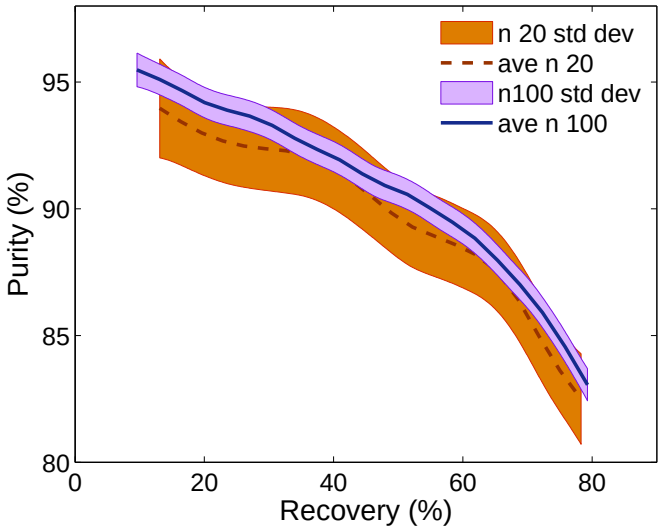


Figure 16. Influence of the population size on the standard deviation from the Pareto front

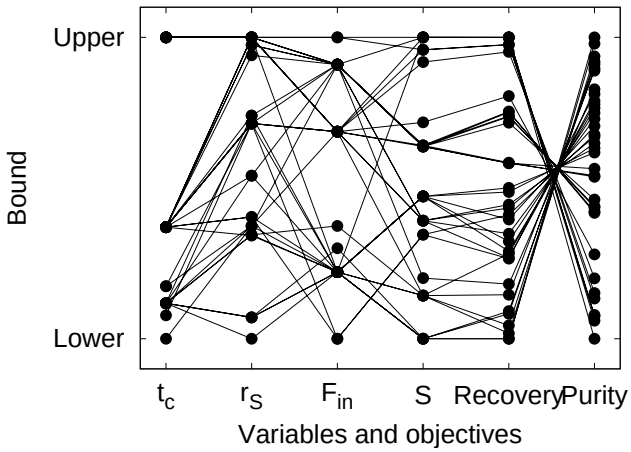
schedule, S, design variables. Also apparent from the clustered intersection points for the lines connecting the two criteria is the inverse relationship between these two criteria, as expected. Overall, the results illustrate the complex relationship between the design variables and the objectives for the design problem. There appear to be hyper-slices in the 4 dimensional design space, defined by values of t_c and F_{in}, which correspond to families of good solutions for particular trade-offs between the two criteria. An almost linear relationship between the two criteria is highlighted not only in the parallel co-ordinate visualisation but also in the Pareto graph.

The cycle time ranges in $\in [137, 248]$, $r_S \in [0.33, 0.82]$, $F_{in} \in [30.5, 87.9]$ and the schedule $\in [20, 44.5]$ %. While an almost linear relation holds between purity and recovery, no design variable shows monotonic behaviour as the Pareto front is traversed.

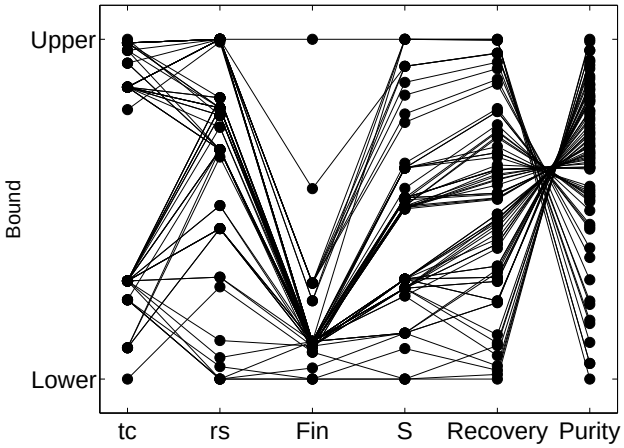
We can compare our results with those presented by (Hassan *et al.* 1986). They also considered air separation on a molecular sieve using a 4-step Skarstrom cycle. To summarise, the following trends have been described by Hassan *et al.* (1986) and observed in the results of our optimisation runs:

- As the split ratio increases, the recovery increases and the purity decreases. This is as expected. The regeneration of the adsorbent (i.e. lower F_{purge}) is less effective with increasing split ratio. Also, the amount of product withdrawn, F_{prod}, increases with the split ratio. The effect of the split ratio is exploited in a modified Skarstrom cycle commonly adopted for air separation, where the purge step is eliminated (i.e. $r_S = 1$) to maximise recovery, while a pressure equalization step is introduced to gain an optimal regeneration of the bed as well as to save compression work (Hassan *et al.* 1987, Ruthven *et al.* 1993).
- Hassan *et al.* (1986)]explored the effect of the purge to feed ratio. This is equivalent to looking at $(1 - r_S)/F_{in}$ for our model. Again, we find agreement in our results: as the purge/feed ratio increases, we get a lower recovery and a higher purity. This is shown clearly in fig. 18.

Hassan *et al.* (1986) explored the effect of the schedule in a restricted range $[75, 90]$ % and they found no appreciable effect on the performance of the operation. In our case, however, the use of detailed modelling and advanced computational tools has



(a) 36 different solutions are represented in this graph, obtained with a population of 40 individuals.



(b) 72 different solutions are represented in this graph, obtained with a population of 100 individuals.

Figure 17. Pareto set visualised using a parallel co-ordinate representation. Variable domains and objective function value ranges have been normalised for presentation.

allowed a wider range of S to be explored. We have observed that the longer the t_{ads} for a given t_c, the higher the recovery and the lower the purity.

Although we do not observe a monotonic relation between the cycle time and the objectives, we do note that lower t_c values correspond to lower recoveries and higher purities. To some extent, peaks in t_c correspond to valleys in F_{in} as the two variables balance out to allow the pressure to satisfy the operating constraint.

8. Conclusions

The problem of designing a PSA process has been addressed. The specific case study is the separation of air for N_2 production. The model is based on a realistic mass transfer equation. Despite simplifications, such as the replacement of the adsorption column by a series of CSTRs, the resulting model is computationally challenging.

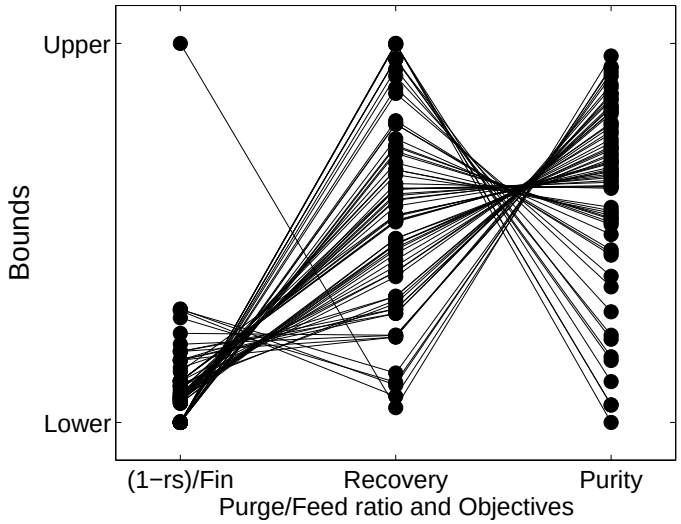


Figure 18. Influence of the Purge/Feed ratio on the objectives

This is especially the case when the intended use is within a design framework.

The design problem proposed is the maximisation of both purity and recovery of N_2, in a 4 dimensional design space. The evaluation of the objectives requires the solution of the dynamic model to cyclic steady state (CSS). A method to achieve an accelerated convergence to CSS, proposed by (Kumar *et al.* 1994), has been adopted. This has allowed us to use the method of successive substitution to identify the CSS, necessary for the calculation of the evaluation criteria. Analysis of the shape of the objective function and of the corresponding feasible region has shown that the objective function is non-smooth and non-convex. Interestingly, the feasible region itself is non-convex. Together, these properties pose challenges to any optimiser.

Two classes of optimisers can cope with the non-smoothness of the objective function: evolutionary algorithms and direct search methods (DSMs). Hence, we have evaluated the performance of a custom multi-objective genetic algorithm (MOGA) to generate approximations to the Pareto front. A targeted fitness function has been defined which emphasises not just points close to the current Pareto set but also those which are close to an infinite extension to that set parallel to the criteria axes. This fitness function encourages the evolutionary procedure to broaden the extent of the front. The aim is to help a design engineer identify the trade-offs between the different criteria and thereby choose the appropriate design point or even a region in the design space for further investigation. A statistical analysis has been carried out to evaluate the average performance of the MOGA. It has been shown the MOGA is by far more efficient and reliable than DSMs in approximating the Pareto front for the problem of interest. The results of the comparison showed that on average DSMs are not able to detect a good approximation of the Pareto front. Moreover, the success of DSMs is very sensitive to the starting point of the optimisation, so that a pre-knowledge of the design problem would be required to obtain a good approximation o the Pareto set.

The results of the case study show good agreement with experimental results (Hassan *et al.* 1986). Furthermore, the analysis of the performance of the MOGA indicates that it has been successfully applied, generating a Pareto front which has sufficient breadth and diversity to demonstrate this agreement. The comparison with DSMs has shown the MOGA is more efficient and reliable. However, this study has been a preliminary investigation into the suitability of multi-objective evolutionary

algorithms for the design of PSA processes. The success of this preliminary investigation now motivates us to investigate the use of existing and validated evolutionary algorithms, such as NSGA-II (Deb *et al.* 2002).

References

Aaron, D. and Tsouris, C., 2005. Separation of CO_2 from flue gas: a review. *Separation Science and Technology*, 40 (1-3), 321–348.

Ahn, H. and Brandani, S., 2005. A New Numerical Method for Accurate Simulation of Fast Cyclic Adsorption Processes. *Adsorption*, 11 (2), 113–122.

Barakat, T., Fraga, E.S., and Sørensen, E., 2008. Multi-objective Optimisation of batch separation processes. *Chem. Eng. & Processing*, 47 (12), 2303–2314.

Biegler, L.T., Jiang, L., and Fox, V.G., 2004. Recent Advances in Simulation and Optimal Design of Pressure Swing Adsorption Systems. *Separation and Purification Reviews*, 33 (1), 1–39.

Cruz, P., Magalhães, F.D., and Mendes, A., 2005. On the Optimization of Cyclic Adsorption Separation Processes. *AIChE Journal*, 51 (5), 1377–1395.

Deb, K., 2001. *Multi-objective optimization using evolutionary algorithms*. Chester,UK: J. Wiley & Son Inc.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T., 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *Evolutionary Computation, IEEE Transactions*, 6 (2), 182–197.

Ding, Y., Croft, D.T., and LeVan, M.D., 2002. Periodic States of Adsorption cycles IV. Direct Optimization. *Chem. Eng. Science*, 57, 4521.

Fiandaca, G., Fraga, E.S., and Brandani, S., 2007. Development of Computer Aided Tools for PSA Processes. *In*: *Proceedins of 07AIChE Annual Meeting*, p. 150.

Hassan, M.M., Raghavan, N.S., and Ruthven, D.M., 1987. Pressure Swing Air Separation on a carbon molecular-sieve. 2. Investigation of a modified cycle with pressure equalization and no purge. *Chem. Eng. Science*, 42 (8), 2037–2043.

Hassan, N.M., Ruthven, D.M., and Raghavan, N.S., 1986. Air Separation by Pressure Swing Adsorption on a Carbon Molecular Sieve. *Chem. Eng. Science*, 41 (5), 1333–1343.

Herrero, J.M., Martinez, M., Sanchis, J., and Blasco, X., 2007. Vol. 4507 of *Lecture Notes in Computer Science*, Well distributed Pareto Front by Using the ϵ-MOGA Evolutionary Algorithm. *In*: *Computational and Ambient Intelligence.*, 292–299 Springer Berlin / Heidelberg.

Horn, J., Nafpliotis, N., and Goldberg, D.E., 1994. A Niched Pareto Genetic Algorithm for Multiobjective Optimization. *In*: *Proceedings of the First IEEE Conference on Evolutionary Computation, IEEE World Congress on Computational Intelligence*, 82–87.

Jain, S., Moharir, A., Li, P., and Wozni, G., 2003. Heuristic Design of Pressure Swing Adsorption: A Preliminary Study. *Separation and Purification Technoloogy*, 33 (1), 25–43.

Jiang, L., Biegler, L.T., and Fox, V.G., 2003. Simulation and Optimization of Pressure Swing Adsorption System for Air Separation. *AIChE Journal*, 49 (5), 1140.

Kelley, C.T., 1999. *Iterative Methods for Optimization*. Frontiers in Applied Mathematics SIAM.

Ko, D. and Moon, I., 2002. Multiobjective optimization of cyclic adsorption processes. *Ind. Eng. Chem. Res.*, 42 (2), 93–104.

Ko, D., Siriwardane, R., and Biegler, L.T., 2003. Optimization of a Pressure-Swing Adsorption Process Using Zeolite 13X for CO_2 Sequestration. *Ind. Eng. Chem. Res.*, 42 (2), 339–348.

Kumar, R., *et al.*, 1994. A versatile process simulator for adsorptive separations. *Chem. Eng. Science*, 49 (18), 3115–3125.

Kvamsdal, H.M. and Hertzberg, T., 1997. Optimization of PSA systems - Studies on cyclic steady state convergence. *Computers & Chemical Engineering*, 21 (8), 819–832.

Laumanns, M., *et al.*, 2002. Combining Convergence and Diversity in Evolutionary Multiobjective Optimization. *Evolutionary Computation*, 10 (3), 263–282.

Laumanns, M., Zitzler, E., and Thiele, L., 2001. Vol. 1993 of *Lecture Notes in Computer Science*, On the Effects of Archiving, Elitism, and Density Based Selection in Evolutionary Multi-objective Optimization. *In*: *Evolutionary Multi-Criterion Optimization.*, 181–196 Springer Berlin / Heidelberg.

Levenspiel, O., 1962. *Chemical Reaction Engineering*. J. Wiley & Son Inc.

Lewandowski, J., Lemcoff, N.O., and Palosaari, S., 1998. Use of neural networks in the simulation and optimization of pressure swing adsorption processes. *Chemical Engineering & Technology*, 21 (7), 593–597.

Lewis, R.M., Torczon, V., and Trosset, M.W., 2000. Direct Search Methods: Then and Now. *Journal of Computational and Applied Mathematics*, 124 (1-2), 191–207.

Meza, J.C., Judson, R.S., Faulkner, T.R., and Treasurywala, A.M., 1996. A Comparison of a Direct Search Method and a Genetic Algorithm for Conformational Searching. *Journal of Computational Chemistry*, 17 (9), 1142–1151.

Murata, T. and Ishibuchi, H., 1995. MOGA: multi-objective genetic algorithms. *In*: *International Conference on Evolutionary Computation, IEEE*, 289–924.

Nilchan, S. and Panthelides, C.C., 1998. On the Optimization of Periodic Adsorption Processes. *Adsorption*, 4 (2), 113–147.

Rajasree, R. and Moharir, A.S., 2000. Simulation Based Synthesis, Design and Optimization of Pressure Swing Adsorption Processes. *Computers & Chemical Engineering*, 24 (11), 2493–2505.

Rasmussen, C.E. and Williams, C.K.I., 2006. *Gaussian Processes for Machine Learning*. The MIT Press Online at `http://www.gaussianprocess.org/` with links to code as well.

Ruthven, D.M., Farooq, S., and Knaebel, K.S., 1993. *Pressure Swing Adsorption*. VCH.

Sankararao, B. and Gupta, S.K., 2007. Multi-objective Optimization of Pressure Swing Adsorbers for Air Separation. *Ind. Eng. Chem. Res.*, 46 (11), 3751–3765.

Schaffer, J.D., 1985. Multiple Objective Optimization with Vector Evaluated Genetic Algorithms. *In*: *Proceedings of the 1st International Conference on Genetic Algorithms* Lawrence Erlbaum Associates, Inc., 93–100.

Sircar, S., 2006. Basic Research needs for the Design of Adsorptive Gas Separation Processes. *Ind. Eng. Chem. Res.*, 45 (16), 5435–5448.

Smith, O.J. and Westerberg, A.W., 1991. The Optimal Design of Pressure Swing Adsorption Systems. *Chem. Eng. Science*, 46 (12), 2967–2976.

Todd, R., Buzzi-Ferraris, G., Manca, D., and Webley, P.A., 2003. Improved ODE Integrator and Mass Transfer Approach for Simulating a Cyclic Adsorption Process. *Computers & Chemical Engineering*, 27 (6), 883–899.

Yu, W., Hidajat, K., and Ray, A.K., 2003. Application of multiobjective optimization in the design and operation of reactive SMB and its experimental verification. *Ind. Eng. Chem. Res.*, 42 (26), 6823–6831.

Zhang, Z., Mazzotti, M., and Morbidelli, M., 2003. Multiobjective optimization of simulated moving bed and Varicol processes using genetic algorithm. *Journal of Chromatography A*, 989 (1), 95–108.

Zitzler, E., Deb, K., and Thiele, L., 2000. Comparison of Multiobjective Evolutionary

Algorithms: Empirical Results. *Evolutionary Computation*, 8 (2), 173–195.

Zitzler, E., Laumanns, M., and Bleuler, S., 2004. A tutorial on evolutionary multiobjective optimization. *In*: *Metaheuristics for Multiobjective Optimisation* Springer-Verlag, 3–38.

Zitzler, E. and Thiele, L., 1999. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *Evolutionary Computation, IEEE Transactions*, 3 (4), 257–271.