Jussi Hakanen – Yoshiaki Kawajiri – Kaisa Miettinen – Lorenz T. Biegler

# INTERACTIVE MULTI-OBJECTIVE OPTIMIZATION OF SIMULATED MOVING BED PROCESSES USING IND-NIMBUS AND IPOPT

HELSINKI SCHOOL OF ECONOMICS
WORKING PAPERS
W-404

Jussi Hakanen* – Yoshiaki Kawajiri** –
Kaisa Miettinen*** – Lorenz T. Biegler**

# INTERACTIVE MULTI-OBJECTIVE OPTIMIZATION OF SIMULATED MOVING BED PROCESSES USING IND-NIMBUS AND IPOPT

* University of Jyväskylä,
Dept. of Mathematical Information Technology

** Carnegie Mellon University,
Dept. of Chemical Engineering

***Helsinki School of Economics,
Dept. of Business Technology

August
2006

HELSINGIN KAUPPAKORKEAKOULU
HELSINKI SCHOOL OF ECONOMICS
WORKING PAPERS
W-404

HELSINGIN KAUPPAKORKEAKOULU
HELSINKI SCHOOL OF ECONOMICS
PL 1210
FI-00101 HELSINKI
FINLAND

ISSN 1235-5674
(Electronic working paper)
ISBN-10: 952-488-055-5
ISBN-13: 978-952-488-055-8

Helsinki School of Economics -
HSE Print 2006

# Interactive Multi-Objective Optimization of Simulated Moving Bed Processes using IND-NIMBUS and IPOPT

Jussi Hakanen[*], Yoshiaki Kawajiri[†], Kaisa Miettinen[§], Lorenz T. Biegler[†]

Dept. of Mathematical Information Technology
P.O. Box 35 (Agora), FI-40014 University of Jyväskylä, Finland
e-mail: jussi.hakanen@mit.jyu.fi

[†] Carnegie Mellon University
Dept. of Chemical Engineering
Pittsburgh, PA, 15213 USA
e-mail: kawajiri@cmu.edu, biegler@cmu.edu

[§] Helsinki School of Economics
P.O. Box 1210, FI-00101 Helsinki, Finland
e-mail: kaisa.miettinen@hse.fi

**Abstract**

In this paper, efficient optimization techniques are used to solve multi-objective optimization problems arising from Simulated Moving Bed (SMB) processes. SMBs are widely used in many industrial separations of chemical products and they are very challenging from the optimization point of view. With the help of interactive multi-objective optimization, several conflicting objectives can be considered simultaneously without making unnecessary simplifications as has been done in previous studies. The optimization techniques used are the interactive NIMBUS® method and the IPOPT optimizer. To demonstrate the usefulness of these techniques, the results of solving an SMB optimization problem with four objectives are reported.

**Keywords:** Multiobjective optimization, interactive methods, NIMBUS, interior point optimization, IPOPT, Simulated Moving Bed processes

[*] Corresponding author

# 1 Introduction

Optimization of real-world industrial processes is often computationally demanding. It may require considerable numbers of evaluations of the process model during the optimization procedure, in order to obtain an optimal solution for a single objective function. On the other hand, real-world optimization problems typically have several conflicting objectives that need to be considered simultaneously, and it is important to have optimization tools that require only few process model simulations in order to obtain optimal or satisfactory solutions quickly.

In real-world problems, the problem is too often simplified by considering only a single criterion as an objective function, although there usually are several criteria that should be optimized at the same time. By considering different criteria simultaneously, one can obtain better understanding of the whole problem and its interdependencies. In multi-objective optimization (MOO) [11], we do not have a single well-defined optimal solution but, instead, a set of mathematically equivalent compromise solutions called Pareto optimal solutions where no objective can get better values without impairment in at least one of the others. In order to obtain the best solution for the problem in question, we often need a decision maker (DM), who can express preference information about different Pareto optimal solutions. Some MOO methods try to approximate the whole set of Pareto optimal solutions. For many real-world problems these methods can be too slow because they need to generate a large number of Pareto optimal solutions. Besides, with more than two objectives, displaying the solutions to the DM is not trivial and it is not easy for the DM to select one of many alternatives as the final solution. Instead, interactive MOO methods aim at decreasing the cognitive burden set on the DM and let the DM to direct the solution process towards the most preferred solution.

In this study, we consider MOO of a challenging periodic adsorption process for the separation of valuable chemical products. In particular, Simulated Moving Bed (SMB) processes have been applied to many important separations in sugar, petrochemical, and pharmaceutical industries. Because they operate dynamically, in periodic cycles, systematic optimization of SMBs remains a challenging problem. Prior to the application of the full discretization approach where both spatial and temporal variables are discretized, optimization of SMB processes was known to be an expensive and challenging problem. In a recent paper on SMB (single-objective) optimization [18], required CPU times for single-objective optimization problems were reported to be 13-26 hours on a Pentium II machine. Moreover, Submarani, Hidajat and Ray [17] used a genetic algorithm to produce the whole pareto set. Although they did not report the computational time, we believe this method is also quite computationally expensive. Both of these papers above used single-discretization method, where only the spatial domain is discretized. In addition, there exist a number of conflicting objectives associated with these processes including productivity, product quality, utilization of desorbent (solvent) and generation of waste streams.

Several recent papers deal with single objective optimization of SMBs (see, for example, [2, 7, 9, 18]). Recently, MOO algorithms have been applied for periodic separation processes by Ko and Moon [10], for the case of gas separation, and by Subramani et al. [17], for SMB processes. Ko and Moon used a modified sum of weighted objective functions to obtain a representation of the Pareto optimal set. Their approach is valid only for two objective functions. On the other hand, Submarani et al. applied non-dominated sorting genetic algorithm (NSGA) to a problem where they had two or three objective functions. NSGA tries to approximate the set of Pareto optimal solutions, but it cannot guarantee the Pareto optimality of the solutions obtained.

In order to accelerate the process optimization, Kawajiri and Biegler [9] have developed an efficient full discretization approach combined with a large-scale nonlinear programming method for the optimization of SMBs. More recently, they extended this approach to a superstructure SMB formulation and used an $\epsilon$-constrained method to solve the bi-objective problem, where throughput and desorbent consumption were considered as objective functions [8].

To summarize, we can say that so far, SMB processes have been considered with 1, 2 or 3 objectives, only. In addition, all of the MOO approaches used have been based on the idea of trying to approximate the whole Pareto optimal set and, thus, need to generate lots of Pareto optimal solutions in order to get a good approximation. Therefore, they are computationally inefficient in the sense Pareto optimal solutions generated since, usually, the ultimate aim is to find a single best solution as the final solution to be implemented.

In this paper, we approach SMB problems in a completely new way and apply interactive MOO to problems arising from SMB processes. We consider the case with four conflicting objective functions which brings new insight into the problem and avoids unnecessary simplifications. Thus, we can obtain a better understanding of the whole process than in previous studies. We are especially interested in efficient and user-friendly solution methods for our computationally demanding problems. Therefore, we use an interactive process design tool IND-NIMBUS, based on the interactive multi-objective optimization method NIMBUS® [11, 14], together with an interior point optimizer IPOPT [19] to solve these problems. The reason for using IPOPT is that the optimizers currently implemented in IND-NIMBUS [5] are not well suited for large-scale optimization problems like SMBs.

This paper is organized as follows. First, we describe the optimization tools used, that is, NIMBUS® with IND-NIMBUS and IPOPT, in Section 2. Then, we give an overview of SMBs and present the multi-objective SMB problem to be solved in Section 3. In Section 4, we describe the interactive solution procedure as well as show and discuss the results obtained. Finally, Section 5 is devoted to concluding remarks and future work.

# 2 Interactive process design tool

## 2.1 IND-NIMBUS

IND-NIMBUS is an interactive tool developed for solving MOO problems arising, for example, in industry. It is based on the interactive MOO method NIMBUS® [11, 12, 14], in other words, IND-NIMBUS is an implementation of the NIMBUS® method. The general idea of the NIMBUS® method is to help the DM in finding the most satisfactory compromise between conflicting objectives without generating too many Pareto optimal solutions.

A benefit of interactive MOO methods is that the DM can guide the solution procedure and is able to learn about the behaviour of the problem. (S)he can study the interrelationships of the objective functions and obtain a wider understanding of their effects on the whole problem. Therefore, as the DM gains more and more understanding of the behaviour of the problem, (s)he can adjust her/his preferences accordingly. Another benefit is that interactive methods are computationally efficient in the sense of Pareto optimal solutions generated, because only a small number of Pareto optimal solutions usually needs to be computed. Furthermore, the DM can concentrate only on those solutions that are of interest to her/him. On the other hand, interactive methods require the DM to take part in the solution procedure continuously and, thus, the DM has to be willing to devote time in the solution procedure. One can expect the DM to find the most preferred solution when (s)he can actively take part in the solution procedure. It is important that the method is easy to use and that the DM can easily answer the questions proposed to her/him and, therefore, we use the NIMBUS® method. Other interactive methods are described in [11] and references therein.

NIMBUS® converts the original multiple objectives together with preference information coming from the DM into a new problem with a single objective function. This new problem can then be solved with appropriate solvers developed for scalar-valued problems. NIMBUS® is particularly well suited for problems involving more than two objective functions. IND-NIMBUS has been previously applied to several industrial optimization problems, including the design of papermaking processes [4, 5].

We consider the following MOO problem. The vector-valued objective function $\boldsymbol{f} = (f_1, \ldots, f_k)^T$ consists of $k$ real-valued objective functions $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$ that are to be optimized simultaneously. The decision variables $\boldsymbol{x} \in \mathbb{R}^n$ belong to the feasible region $S$. The objective vectors $\boldsymbol{z} = \boldsymbol{f}(\boldsymbol{x})$ are in the objective space $\mathbb{R}^k$. We assume that the objective functions are conflicting, that is, all of them do not attain their optima at the same decision vector $\boldsymbol{x}$. Thus, the MOO problem considered is of the form

$$\begin{aligned} &\text{minimize} \quad \{f_1(\boldsymbol{x}), \ldots, f_k(\boldsymbol{x})\} \\ &\text{subject to} \quad \boldsymbol{x} \in S. \end{aligned} \tag{1}$$

For the simplicity of presentation, we assume that all the objective functions in this section are to be minimized. If some objective function $f_i$ is to be maximized, it is equivalent to minimize $-f_i$. Minimization of the vector-valued objective function $\boldsymbol{f}$ is understood in the sense of Pareto optimality: A feasible decision vector $\boldsymbol{x}^*$ is called a Pareto optimal solution if there does not exist another feasible decision vector $\boldsymbol{x}$ such that $f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x}^*)$ for all $i = 1, \ldots, k$ and $f_j(\boldsymbol{x}) < f_j(\boldsymbol{x}^*)$ for at least one $j$. Provided that the problem is correctly specified, the final solution of a rational DM is always Pareto optimal and, thus, we can restrict our consideration to Pareto optimal solutions.

We can summarize the interactive NIMBUS® algorithm as follows: First, a Pareto optimal starting point is generated together with additional information regarding the ranges of each objective function in the set of Pareto optimal solutions. The ranges are defined by the ideal and nadir objective vectors. The ideal objective vector $\boldsymbol{z}^*$ consists of the minima of individual objective functions subject to constraints while the values for the nadir objective function $\boldsymbol{z}^{nad}$ can be approximated by using the payoff table [11]. Note, that the payoff table can give a poor approximation for the nadir objective vector [11]. We show the ideal and nadir objective vectors to the DM in order to inform her/him what is possible to achieve in the problem in question.

As a starting point, we use the neutral compromise solution [21], which is approximately in the middle of the Pareto optimal set. At every iteration, the DM is shown the values of the objective functions in the current Pareto optimal solution ($\boldsymbol{f}(\boldsymbol{x}^c)$) and (s)he is asked to indicate how the current solution should be improved. This is done with the help of classifying the objective functions into up to five different classes, that is, the functions $f_i$ whose value

- should be improved as much as possible ($i \in I^{imp}$),
- should be improved until some specified aspiration level $\bar{z}_i$ ($i \in I^{asp}$),
- is satisfactory at the moment ($i \in I^{sat}$),
- can impair up to some specified bound $\epsilon_i$ ($i \in I^{bound}$) and
- can change freely ($i \in I^{free}$).

Note, that $I^{imp} \cup I^{asp} \cup I^{sat} \cup I^{bound} \cup I^{free} = \{1, \ldots, k\}$. The aspiration levels and the upper bounds are given by the DM. In order to be able to produce a new Pareto optimal solution, the classification is acceptable if $I^{imp} \cup I^{asp} \neq \emptyset$ and $I^{bound} \cup I^{free} \neq \emptyset$, because one must let some objective function to impair in order to improve some other objective function. Note, that the definition of the classes does not take into account whether individual objectives are to minimized or maximized.

Then, according to the classification information provided by the DM, several new solutions are generated that try to follow the preferences of the DM as well as possible. This is realized by forming several single objective subproblems in the synchronous NIMBUS [14]. The number of the new solutions to be generated can be between 1 and 4 as requested by the DM. Different subproblems can produce different solutions with the same preference information and, therefore, it is better to let the DM select the most preferred of them [13]. These subproblems are then solved with a suitable single objective optimizer. In this paper, for the first time we use the IPOPT optimizer [19]. Note, that the solutions of the subproblems are always Pareto optimal (see [14] and references therein). The solutions obtained and the current Pareto optimal solution are then shown to the DM, who selects the most preferred one. In IND-NIMBUS, different types of visualizations of the Pareto optimal solutions are provided to the DM in order to aid the comparison and selection. There is also a possibility to generate intermediate Pareto optimal solutions between any two Pareto optimal solutions obtained. Then, if the DM is satisfied with the solution obtained, we terminate the solution procedure, otherwise we ask the DM to make another classification in the solution (s)he selected.

There are four different subproblems used in the NIMBUS$^{\circledR}$ method, the standard subproblem formulated by the developers of NIMBUS$^{\circledR}$ and three subproblems based on reference points. In all the subproblems, the latter part of the objective function is called an augmentation term and it guarantees that the solution obtained is Pareto optimal (for further details see [11] and [14]). The augmentation parameter $\rho > 0$ is a small coefficient. The standard subproblem is of the form

$$\begin{array}{ll} \text{minimize} & \max\limits_{i\in I^{imp},\, j\in I^{asp}} \left[ \frac{f_i(\boldsymbol{x})-z_i^*}{z_i^{nad}-z_i^{**}}, \frac{f_j(\boldsymbol{x})-\bar{z}_j}{z_j^{nad}-z_j^{**}} \right] + \rho \sum\limits_{i=1}^{k} \frac{f_i(\boldsymbol{x})}{z_i^{nad}-z_i^{**}} \\ \text{subject to} & f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x}^c) \text{ for all } i \in I^{imp} \cup I^{asp} \cup I^{sat}, \\ & f_i(\boldsymbol{x}) \leq \epsilon_i, \text{ for all } i \in I^{bound}, \\ & \boldsymbol{x} \in S, \end{array} \tag{2}$$

where $z_i^{**} = z_i^* - \sigma$ and $\sigma > 0$ is a small positive constant.

The additional subproblems are based on reference points. From the classification information provided by the user, we can form a reference point $\hat{\boldsymbol{z}}$, where $\hat{z}_i = z_i^*$ for $i \in I^{imp}$, $\hat{z}_i = \bar{z}_i$ for $i \in I^{asp}$, $\hat{z}_i = f_i(\boldsymbol{x}^c)$ for $i \in I^{sat}$, $\hat{z}_i = \epsilon_i$ for $i \in I^{bound}$ and $\hat{z}_i = z_i^{nad}$ for $i \in I^{free}$. These additional subproblems are very similar in structure but in comparisons [13] they have proven to generate different enough solutions. Here, we present only one of them. The mathematical formulation of the subproblem based on achievement (scalarizing) functions [20] is

$$\begin{array}{ll} \text{minimize} & \max\limits_{i=1,\ldots,k} \left[ \frac{f_i(\boldsymbol{x})-\hat{z}_i}{z_i^{nad}-z_i^{**}} \right] + \rho \sum\limits_{i=1}^{k} \frac{f_i(\boldsymbol{x})}{z_i^{nad}-z_i^{**}} \\ \text{subject to} & \boldsymbol{x} \in S. \end{array} \tag{3}$$

Other subproblems are based on the satisficing trade-off method [15] and the GUESS method [1].

All of the above-mentioned subproblems are nonsmooth regardless of the properties of the original objective and constraint functions. Because all the functions in our application are differentiable and in order to use the IPOPT optimizer to solve the subproblems above, we reformulate them. Note, that we use IPOPT because it is well suited for large-scale optimization problems like SMBs. Because our subproblems are all min-max type, we can in a standard way introduce a new variable $\delta \in \mathbb{R}$ and get corresponding smooth variants of problems (2) and (3):

$$\begin{array}{ll} \underset{\boldsymbol{x},\delta}{\text{minimize}} & \delta + \rho \sum\limits_{i=1}^{k} \frac{f_i(\boldsymbol{x})}{z_i^{nad}-z_i^{**}} \\ \text{subject to} & \frac{f_i(\boldsymbol{x})-z_i^*}{z_i^{nad}-z_i^{**}} \leq \delta, \quad i \in I^{imp}, \\ & \frac{f_j(\boldsymbol{x})-\bar{z}_j}{z_j^{nad}-z_j^{**}} \leq \delta, \quad j \in I^{asp}, \\ & f_i(\boldsymbol{x}) \leq f_i(\boldsymbol{x}^c) \text{ for all } i \in I^{imp} \cup I^{asp} \cup I^{sat}, \\ & f_i(\boldsymbol{x}) \leq \epsilon_i, \text{ for all } i \in I^{bound}, \\ & \boldsymbol{x} \in S, \quad \delta \in \mathbb{R} \end{array} \tag{4}$$

and

$$\begin{aligned} \underset{\boldsymbol{x},\delta}{\text{minimize}} \quad & \delta + \rho \sum_{i=1}^{k} \frac{f_i(\boldsymbol{x})}{z_i^{nad} - z_i^{**}} \\ \text{subject to} \quad & \frac{f_i(\boldsymbol{x}) - \hat{z}_i}{z_i^{nad} - z_i^{**}} \leq \delta, \quad i = 1, \ldots, k \\ & \boldsymbol{x} \in S, \quad \delta \in \mathbb{R}, \end{aligned} \tag{5}$$

respectively.

The method for generating intermediate solutions is the following. If the DM wants to generate $P$ intermediate solutions between $\boldsymbol{f}(\boldsymbol{x}^1)$ and $\boldsymbol{f}(\boldsymbol{x}^2)$, we first set $\boldsymbol{d} = \boldsymbol{x}^2 - \boldsymbol{x}^1$. Then, we calculate $P$ different vectors $\boldsymbol{f}(\boldsymbol{x}^1 + t_j\boldsymbol{d})$, where $t_j = j/(P+1)$ and $j = 1, \ldots, P$. The vectors obtained are projected onto the Pareto optimal set by applying subproblem (3) or (5) with $\boldsymbol{f}(\boldsymbol{x}^1 + t_j\boldsymbol{d})$ as the reference point for $j = 1, \ldots, P$.

There is also an implementation of the NIMBUS® method operating on the Internet since 1995 called WWW-NIMBUS. WWW-NIMBUS is freely available in `http://nimbus.it.jyu.fi/` for academic purposes.

## 2.2 Interior point optimization

IPOPT is a large-scale nonlinear optimization package [19] based on a Newton-based interior point (barrier) algorithm with filter line-search method. For a general optimization problem

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & f(\boldsymbol{x}) \\ \text{subject to} \quad & \boldsymbol{c}(\boldsymbol{x}) = 0, \\ & \boldsymbol{x}^l \leq \boldsymbol{x} \leq \boldsymbol{x}^u, \end{aligned} \tag{6}$$

where $\boldsymbol{x} \in \mathbb{R}^n$, $f : \mathbb{R}^n \to \mathbb{R}$, and $\boldsymbol{c} : \mathbb{R}^n \to \mathbb{R}^m$, IPOPT applies an interior penalty formulation to convert and solve problem (6) as a sequence of barrier problems

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & \varphi_\mu(\boldsymbol{x}) := f(\boldsymbol{x}) - \mu \sum_{i=1}^{n} \ln(x_i - x_i^l) - \mu \sum_{i=1}^{n} \ln(x_i^u - x_i) \\ \text{subject to} \quad & \boldsymbol{c}(\boldsymbol{x}) = 0, \end{aligned} \tag{7}$$

for a decreasing sequence of barrier parameters $\mu > 0$ converging to zero. We assume, that the functions in problem (6) are twice continuously differentiable. It can be shown, that under mild regularity assumptions the solutions of the barrier problems (7) will converge to the solution of the original problem (6) as $\mu \to 0$. Note, that in IPOPT, all inequality constraints are converted internally into equality constraints by using slack variables [3].

To solve the barrier problem (7) for a fixed parameter $\mu$, a damped Newton's method is applied to primal-dual equations of problem (6) [19], that is,

$$\begin{aligned} \nabla f(\boldsymbol{x}) + \nabla \boldsymbol{c}(\boldsymbol{x})\boldsymbol{\lambda}^T - W^l + W^u &= 0 \\ \boldsymbol{c}(\boldsymbol{x}) &= 0 \\ (X - X^l)W^l\boldsymbol{e} - \mu\boldsymbol{e} &= 0 \\ (X^u - X)W^u\boldsymbol{e} - \mu\boldsymbol{e} &= 0, \end{aligned} \tag{8}$$

where $X = diag(\boldsymbol{x})$, $W^l = diag(\boldsymbol{w}^l)$, $W^u = diag(\boldsymbol{w}^u)$ and $\boldsymbol{e} = (1, ..., 1)^T$. The vectors $\boldsymbol{\lambda} \in \mathbb{R}^m$, $\boldsymbol{w}^l \in \mathbb{R}^n$ and $\boldsymbol{w}^u \in \mathbb{R}^n$ are the Lagrange multipliers of the equality constraints, lower bounds and upper bounds, respectively. After applying Newton's method to equations (8), the new search direction for variables and the Lagrange multipliers can be solved from equations that result from linearizations of equations (8) at the current iterate.

Next, a suitable step size is determined along the new search direction using a filter line search [19]. The filter line search uses a bi-criteria reduction, where either the objective function value or constraint violation is reduced. Essentially, the step size is acceptable if it leads to a sufficient reduction in either of these two criteria. In addition, the method maintains a "filter" that defines a prohibited region for the line search and prevents the algorithm from cycling. Further details of this method can be found in [19].

The IPOPT algorithm has been shown to be globally and superlinearly convergent under weaker assumptions than other barrier methods. IPOPT has been applied to thousands of test problems and applications and has been adopted by a widespread user community. A key advantage is the ability to use second derivative information efficiently. More information and background on IPOPT can be found on `http://projects.coin-or.org/Ipopt`.

# 3 Simulated moving bed processes

## 3.1 Chromatographic separation and simulated moving beds

Efficient purification techniques are crucial in chemical process industries. In production of pharmaceuticals, it is often necessary to purify a product to nearly 100% for regulatory reasons. In food and sugar industries, removal of undesired components that degrade product quality is essential. One of the most common separation techniques is distillation, where the feed mixture is vaporized. However, this is not applicable to substances with an extremely high boiling point, or thermally unstable products such as proteins. For such products, liquid chromatographic separation has been widely used, where the feed mixture is separated making use of the difference in the migration speed of components in liquid. In liquid chromatographic processes, a small amount of feed mixture is supplied to an end of a column which is packed with adsorbent particles, and then pushed toward the other end with desorbent (water, organic solvent, or mixture of these). Unlike distillation, this technique does not require heat. Instead, a slight difference in the affinity for the adsorbent is necessary, which makes it a powerful method for separations of mixtures with similar chemical structures.

Simulated Moving Bed (SMB) chromatography emerged from industry in the 1960s as a realization of continuous and counter-current operation of chromatographic separation [16]. An SMB unit consists of multiple columns which are packed with adsorbent particles. The columns are connected to each other making a circulation loop, as shown in Figure 1. In the SMB process of Figure 1, there are altogether eight columns. The feed mixture is inserted into the process in the upper left corner, while desorbent input is in the lower right corner. The two products raffinate and extract are collected in the upper right corner and lower left corner, respectively. Feed mixture and desorbent are supplied between columns continuously. At the same time, two products, raffinate and extract, are withdrawn from the loop also continuously. The two inlet and two outlet streams are switched in the direction of the liquid flow at a regular interval, or Steptime. Because of the four inlet/outlet streams, there are four liquid velocity zones in the SMB loop as shown in Figure 1. The operation of SMB is uniquely determined by the five parameters: four zone velocities, $u^I, \ldots, u^{IV}$, and Steptime $t_{step}$.

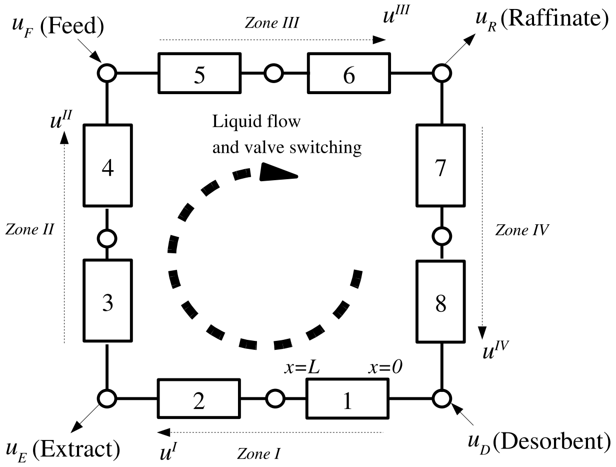

Figure 1: A Schematic diagram of a SMB process.

Due to such a sequence of switching, numerical methods based on steady-state assumptions cannot be directly applied to SMB. Instead, from start-up to shut-down, the four streams proceed around the SMB loop, repeating the same operation cycle. After the start-up period, which typically takes 50 to 100 cycles, the chemical components in the columns to be separated have profiles that repeat in every cycle, thus leading to a Cyclic Steady State (CSS). Even at CSS, the concentration profiles of chemical components still propagate through the columns. Therefore, treatment of CSS requires dynamic simulation and optimization techniques.

## 3.2 Modeling of SMB

To model each chromatographic column, mass transfer and equilibrium of chemical components are taken into an account. In this study, we assume that the equilibrium between the components and adsorbent is described by a linear isotherm. For the mass transfer, we employ the Linear Driving Force (LDF) model, where the mass transfer of components from liquid phase to adsorbent phase is described by a linear equation [9]:

$$
\begin{aligned}
\epsilon_b \frac{\partial C_i^j(x,t)}{\partial t} + (1-\epsilon_b)\frac{\partial q_i^j(x,t)}{\partial t} + u^p \frac{\partial C_i^j(x,t)}{\partial x} &= 0, \\
(1-\epsilon_b)\frac{\partial q_i^j(x,t)}{\partial t} &= K_{appl\,i}(C_i^j(x,t) - C_i^{j,eq}(x,t)), \\
q_i^j(x,t) &= K_i C_i^{j,eq}(x,t), \\
C_i^j(0,t) &= C_i^{j,in}(t), \\
i = 1, ..., N_c, j = 1, ..., N_{Col}, p = I, II, III, IV,
\end{aligned}
\tag{9}
$$

where $0 \leq x \leq L$, $L$ is the length of each column, $t > 0$, $\epsilon_b$ is the void fraction, $C_i^j(x,t)$ is the concentration in the liquid phase of component $i$ in column $j$, $q_i^j(x,t)$ is the concentration in the solid phase, $u^p$ is the superficial liquid velocity in zone $p$, $C_i^{j,in}(t)$ is the inlet concentration, $C_i^{j,eq}(x,t)$ is the equilibrium concentration in the liquid phase, $K_{appl\,i}$ is the mass transfer coefficient, and $K_i$ is the equilibrium constant . The subscripts $i$ correspond to chemical components, superscript $j$ is the column index, and $p$ the zone number, $I$, $II$, $III$, and $IV$, as shown in Figure 1. Furthermore, $N_c$ is the total number of chemical components, $N_{Column}$ is the number of columns, and $N_p$ is the number of columns in zone $p$, with $N_I + N_{II} + N_{III} + N_{IV} = N_{Column}$.

The connections between chromatographic columns are described by the following equations:

*Desorbent inlet port:*

$$u^{IV} + u_D = u^I, \tag{10}$$
$$C_i^{1,in} u^I = C_i^{N_{Column}}(L,t) u^{IV}, \tag{11}$$

*Extract outlet port:*

$$u^I - u_E = u^{II}, \tag{12}$$
$$C_i^{N_I}(L,t) = C_i^{N_{I+1},in}(t), \tag{13}$$

*Feed inlet port:*

$$u^{II} + u_F = u^{III}, \tag{14}$$
$$C_i^{N_I+N_{II}}(L,t) u^{II} + C_{F,i} u_F = u^{III} C_i^{N_I+N_{II}+1,in}(t), \tag{15}$$

*Raffinate outlet port:*

$$u^{III} - u_R = u^{IV}, \tag{16}$$
$$C_i^{N_I+N_{II}+N_{III}}(L,t) = C_i^{N_I+N_{II}+N_{III}+1,in}(t), \tag{17}$$

where $C_{F,i}$ is the feed concentration and $u_D$, $u_E$, $u_F$ and $u_R$ are the superficial velocities (flow rates divided by cross sectional area of column) of the desorbent, extract, feed, and raffinate, respectively.

At the CSS, the concentration profiles of chemical components are identical at the beginning and at the end of a cycle. Since SMB repeats the same operation for all of the columns, the profiles at the beginning of a step are identical to those shifted by the length of a column at the end of the step:

$$\begin{aligned} C_i^j(x,0) &= C_i^{j+1}(x,t_{step}), \quad j=1,...,N_{Column}-1, \\ q_i^j(x,0) &= q_i^{j+1}(x,t_{step}), \quad j=1,...,N_{Column}-1, \\ C_i^{N_{Column}}(x,0) &= C_i^1(x,t_{step}), \\ q_i^{N_{Column}}(x,0) &= q_i^1(x,t_{step}), \end{aligned} \tag{18}$$

where $t_{step}$ is the Steptime. The quality of the products can be measured by purities $PurE_i$ of chemical component $i$ for the extract product:

$$PurE_i = \frac{\int\limits_0^{t_{step}} C_i^{N_I}(L,t)dt}{\sum\limits_{k=1}^{Nc} \int\limits_0^{t_{step}} C_k^{N_I}(L,t)dt}. \tag{19}$$

Recoveries $RecE_i$ measure how much of component $i$ goes from the feed stream to the extract stream:

$$RecE_i = \frac{u_E \int\limits_0^{t_{step}} C_i^{N_I}(L,t)dt}{u_F \int\limits_0^{t_{step}} C_{F,i}dt} = \frac{u_E \int\limits_0^{t_{step}} C_i^{N_I}(L,t)dt}{t_{step}u_F C_{F,i}}. \tag{20}$$

Finally, the zone velocities must be between upper and lower bounds, $u^u$ and $u^l$, respectively:

$$u^l \le u^p \le u^u, \ p = I, II, III, IV. \tag{21}$$

The SMB model above is fully discretized both in spatial and temporal domains, that is, the state variables ($C_i^j(x,t), q_i^j(x,t)$, etc) are fully-discretized in space and time and, therefore, are converted into algebraic variables. The spatial domain is discretized via a central difference method, thus converting the PDEs to ODEs. The resulting ODEs are then discretized in the temporal domain using a Runge-Kutta discretization, Radau collocation on finite elements. More information on this discretization can be found in [9]. The multi-objective SMB problem considered in this paper is based on this model and it is presented next.

### 3.3 Multi-objective SMB problem

The SMB problem to be considered is for the separation of fructose and glucose. The values for the parameters in the SMB model are from [6, 9] and they are shown in Table 1. The SMB optimization problem is large having 33 997 decision variables and 33 992 equality constraints. Thus, the degrees of freedom of the SMB problem is 5.

| parameter | value | parameter | value |
|---|---|---|---|
| $\epsilon_b$ | 0.389 | | |
| $K_1$ | 0.518 | $K_{apps\ 1}$ [1/s] | $6.84 \times 10^{-3}$ |
| $K_2$ | 0.743 | $K_{apps\ 2}$ [1/s] | $6.84 \times 10^{-3}$ |
| $C_{F,1}$ [%] | 50.0 | $C_{F,2}$ [%] | 50.0 |
| $u_l$ [m/h] | 0.1 | $u_u$ [m/h] | 8.00 |
| $L$ [m] | 2.0 | $N_{Column}$ | 4 |
| $N_c$ | 2 | | |

Table 1: Parameters of fructose/glucose separation

For the MOO of SMB, we use four different objective functions for this problem: maximize throughput, T, ($u_F$ in eq. (14)), minimize consumption of solvent in the desorbent stream, D, ($u_D$ in eq. (10)), maximize product purity, P, (eq. (19) for the component $i = 2$), and maximize recovery of the valuable component in the product, R, (eq. (20) for the component $i = 2$). Thus, our vector-valued objective function is $\boldsymbol{f} = (T, D, P, R)^T$ and the multi-objective SMB problem is of the form

$$\begin{array}{ll} \text{maximize} & u_F \\ \text{minimize} & u_D \\ \text{maximize} & PurE_2 \\ \text{maximize} & RecR_2 \\ & \\ \text{subject to} & (9) - (21). \end{array} \tag{22}$$

These objectives have a number of clear conflicts. For example, if throughput is increased, the feed mixture is more likely to contaminate the products, that is, the purity decreases. Moreover, with uncertain prices and raw material costs, a multi-objective approach can be an extremely useful tool for SMB design.

By imposing upper bounds on objective functions, we eliminate part of feasible region that gives impractical objective values; it is unlikely that the DM is satisfied with solutions of extremely low throughput, high desorbent consumption, low purity, or low recovery. Restricting the feasible region is also important for numerical stability. For example, if throughput approaches zero, then concentrations of the chemical components in the columns become zero, making the denominators of equations (19) and (20) very small numbers. For the reasons above, we set the following bounds for the objective functions: $T \geq 0.4$, $D \leq 10.0$, $P \geq 90.0$ and $R \geq 70.0$.

The discretized multiobjective optimization problem (22) is constrained by bilinear equality and bilinear inequality constraints. The different objective functions are either linear (Throughput and Desorbent) or bilinear (Purity and Recovery). As a result, the optimization problem is non-convex.

We use IND-NIMBUS coupled with IPOPT to solve this four objective SMB problem. When we use the reformulated scalarizing functions for this SMB problem, the regularity assumptions for IPOPT are fulfilled. The SMB model has been implemented in AMPL modelling language. The multiple objectives are handled by the computationally efficient NIMBUS® method while the single objective subproblems are solved with the efficient IPOPT optimizer. With this approach, we minimize the time used in generating new solution candidates for the DM and maintain the interactive nature of the solution procedure. During the iterative solution procedure, the DM gets valuable information about the interrelationships between the conflicting objectives.

# 4 Multi-objective optimization of SMBs

## 4.1 Solution procedure

Next, we describe the solution procedure. The interactive solution procedure described in text does not fully correspond to the actual situation because the DM can continuously utilize the visualizations provided by IND-NIMBUS. To illustrate this, a screenshot of the classification window of IND-NIMBUS is shown in Figure 2. In IND-NIMBUS, the classification is made by clicking different parts of the bars representing the objective functions in the current Pareto optimal solution (single solution in the left in Figure 2). The value of each objective function is represented with a colored bar that originates from left and right for the objectives to be minimized and maximized, respectively. In both cases, the interpretation is the same: the less color, the better is the corresponding objective function value. All the solutions generated can be seen on the right and interesting solutions can be taken in the set Best candidates at any time.

The decision maker involved in this solution procedure was an expert in the field of SMBs. The ideal and nadir objective vectors were $\boldsymbol{z}^* = (0.891, 0.369, 97.2, 90.0)^T$ and $\boldsymbol{z}^{nad} = (0.400, 2.21, 90.0, 70.0)^T$, respectively. In other words, those are the approximated ranges for the Pareto optimal set. Note, that we are minimizing desorbent consumption (the second objective function) and maximizing all the others, as mentioned previously.

As a starting point for MOO, we obtained a neutral compromise solution $\boldsymbol{f}(\boldsymbol{x}^1) = (0.569, 1.58, 92.5, 76.9)^T$. First, the DM wanted to improve purity and throughput and he was willing to compromise desorbent consumption and recovery. Thus, he made the first classification ($I^{imp}$={P}, $I^{asp}$={T}, $\bar{z}_T = 0.715$, $I^{bound}$={D,R} with $\epsilon_D = 1.78$ and $\epsilon_R = 74.5$). The number of new solutions to be generated was set to four and we obtained four different solutions ($\boldsymbol{f}(\boldsymbol{x}^2) = (0.569, 1.56, 93.3, 74.5)^T$, $\boldsymbol{f}(\boldsymbol{x}^3) = (0.553, 1.43, 94.8, 70.0)^T$, $\boldsymbol{f}(\boldsymbol{x}^4) = (0.412, 1.07, 97.0, 70.0)^T$ and $\boldsymbol{f}(\boldsymbol{x}^5) = (0.570, 1.52, 93.9, 72.4)^T$).

From the solutions obtained, the DM preferred $\boldsymbol{f}(\boldsymbol{x}^3)$ because it had good values for purity and throughput. Next, he wanted the values of purity and throughput to remain un-

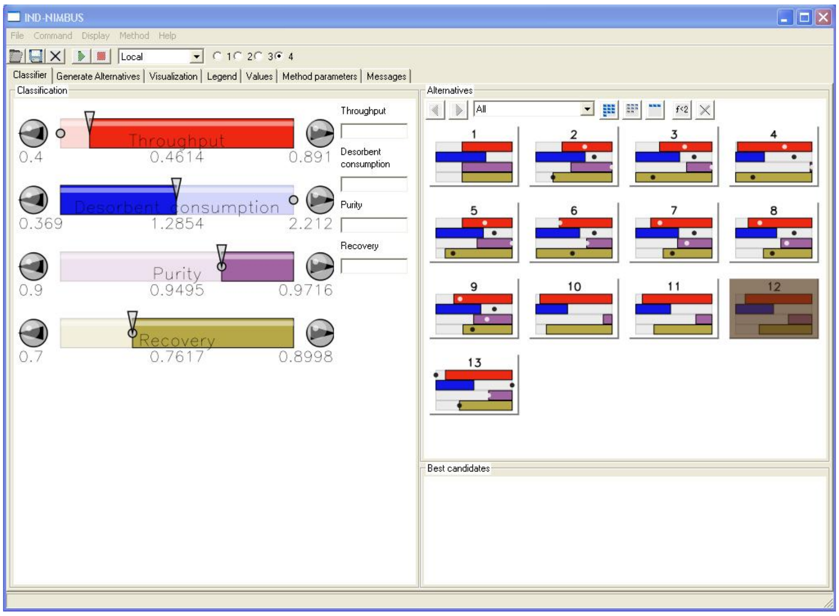

Figure 2: A screenshot of IND-NIMBUS.

changed and study whether recovery could be improved by compromising desorbent consumption ($I^{asp}$={R}, $\bar{z}_R = 0.796$, $I^{sat}$={P,T}, $I^{bound}$={D} with $\epsilon_D = 1.78$). The number of new solutions to be generated was again set to four and three different solutions were obtained this time ($\boldsymbol{f}(\boldsymbol{x}^7) = (0.497, 1.41, 93.9, 77.2)^T$, $\boldsymbol{f}(\boldsymbol{x}^8) = (0.481, 1.36, 94.2, 77.3)^T$ and $\boldsymbol{f}(\boldsymbol{x}^9) = (0.515, 1.46, 93.5, 77.1)^T$).

In all of the three new solutions, recovery improved but the values for purity and throughput impaired. This means, that recovery could not be improved while keeping the values of purity and throughput in the levels they had in solution $\boldsymbol{f}(\boldsymbol{x}^3)$. At this point, the DM preferred the solution $\boldsymbol{f}(\boldsymbol{x}^8)$ because it had better values for recovery and desorbent consumption than $\boldsymbol{f}(\boldsymbol{x}^3)$.

Next, he wanted to generate intermediate solutions between $\boldsymbol{f}(\boldsymbol{x}^4)$ and $\boldsymbol{f}(\boldsymbol{x}^8)$ in order to study the effect of improving purity. The number of intermediate solutions was set to 3. The following intermediate solutions were obtained: $\boldsymbol{f}(\boldsymbol{x}^{10}) = (0.426, 1.14, 96.3, 72.8)^T$, $\boldsymbol{f}(\boldsymbol{x}^{11}) = (0.443, 1.21, 95.6, 74.8)^T$ and $\boldsymbol{f}(\boldsymbol{x}^{12}) = (0.461, 1.29, 95.0, 76.2)^T$. Within these new solutions, the DM preferred $\boldsymbol{f}(\boldsymbol{x}^{12})$ over $\boldsymbol{f}(\boldsymbol{x}^8)$ because it had better value for purity.

Finally, he wanted to study whether it would be possible to improve throughput by compromising desorbent consumption. However, the classification could not produce different solutions while keeping the values for purity and recovery which means that it is a limitation of the process itself. Therefore, as the final solution the DM selected $\boldsymbol{f}(\boldsymbol{x}^{12})$ which had throughput 0.461 m/h, desorbent consumption 1.29 m/h, purity 95.0% and recovery 76.2%.

## 4.2 Results and discussion

The solutions obtained during the solution process are shown in Table 2. The solutions selected at each step of the solution procedure are shown in bold face, while the final solution is shown in both bold face and italics. In addition, the action of the DM (classification or generating intermediate solutions) in each step of the solution procedure is shown in the table.

Figure 3 shows examples of different visualizations provided by IND-NIMBUS. The visualizations help the DM in the decision making procedure. The solutions visualized in Figure 3 are the neutral compromise solution (1) and the solutions preferred at each step of the solution procedure (3, 8 and 12). (In IND-NIMBUS, the DM can choose those visualizations that seem most informative among 7 different types.)

The interactive solution procedure gave the DM wider understanding of the relationships in the problem when he was able to consider all the four objective functions simultaneously. This problem could have been solved also by maximizing total profit which is a scalar-valued function depending on all four objectives used here. However, defining the total profit function is difficult and, therefore, our approach gives us a flexible way to consider this problem without making any unnnecessary simplifications. In addition, the DM found IND-NIMBUS easy to use and he considered the visualizations of IND-NIMBUS to be very helpful in comparing different Pareto optimal solutions obtained. The final solution obtained had good values for purity and recovery. On the other hand, the value for throughput could have been better, but it was impossible to improve without compromising purity and recovery.

The combination of IND-NIMBUS and IPOPT turned out to be robust when solving the SMB problem. The SMB optimization problem is challenging with a large number of variables and constraints and it could be difficult to solve. However, during the solution procedure, IPOPT was able to solve all the subproblems produced by IND-NIMBUS. On the other hand, we needed to compute only 13 Pareto optimal solutions which is a very small number when compared to the methods that try to approximate the whole Pareto optimal set. Producing a single Pareto optimal solution took 16.4 IPOPT iterations (27.6 objective function evaluations) and 65.8 CPU seconds on average.

# 5 Conclusions

For the first time, we have applied an interactive approach together with efficient optimization techniques in solving multi-objective optimization problems arising from Simulated Moving Bed processes. We used interactive NIMBUS® method in multi-objective optimization while single objective subproblems generated by the NIMBUS® method were solved with the interior point optimizer IPOPT. As an example of the efficiency of our approach, we solved an SMB optimization problem having four conflicting objective functions. Considering four conflicting objective functions in the SMB problem simultaneously was a novel approach and we were able to get a better understanding of the SMB problem when compared to previous studies where usually one or two objectives were considered.

| Solution | T [m/h], max | D [m/h], min | P [%], max | R [%], max |
|---|---|---|---|---|
| Ideal | 0.891 | 0.369 | 97.2 | 90.0 |
| Nadir | 0.400 | 2.21 | 90.0 | 70.0 |
| $\boldsymbol{f}(\boldsymbol{x}^1)$ | 0.569 | 1.58 | 92.5 | 76.9 |
|  | $I^{asp}, \bar{z}_T = 0.715$ | $I^{bound}, \varepsilon_D = 1.78$ | $I^{imp}$ | $I^{bound}, \varepsilon_R = 74.5$ |
| $\boldsymbol{f}(\boldsymbol{x}^2)$ | 0.569 | 1.56 | 93.3 | 74.5 |
| $\boldsymbol{f}(\boldsymbol{x}^3)$ | **0.553** | **1.43** | **94.8** | **70.0** |
| $\boldsymbol{f}(\boldsymbol{x}^4)$ | 0.412 | 1.07 | 97.0 | 70.0 |
| $\boldsymbol{f}(\boldsymbol{x}^5)$ | 0.570 | 1.52 | 93.9 | 72.4 |
|  | $I^{sat}$ | $I^{bound}, \varepsilon_D = 1.78$ | $I^{sat}$ | $I^{asp}, \bar{z}_R = 0.796$ |
| $\boldsymbol{f}(\boldsymbol{x}^7)$ | 0.497 | 1.41 | 93.9 | 77.2 |
| $\boldsymbol{f}(\boldsymbol{x}^8)$ | **0.481** | **1.36** | **94.2** | **77.3** |
| $\boldsymbol{f}(\boldsymbol{x}^9)$ | 0.515 | 1.46 | 93.5 | 77.1 |
|  | 3 interm. | solutions between | $\boldsymbol{f}(\boldsymbol{x}^4)$ | $\boldsymbol{f}(\boldsymbol{x}^8)$ |
| $\boldsymbol{f}(\boldsymbol{x}^{10})$ | 0.426 | 1.14 | 96.3 | 72.8 |
| $\boldsymbol{f}(\boldsymbol{x}^{11})$ | 0.443 | 1.21 | 95.6 | 74.8 |
| $\boldsymbol{f}(\boldsymbol{x}^{12})$ | ***0.461*** | ***1.29*** | ***95.0*** | ***76.2*** |

Table 2: Pareto optimal objective function values for the SMB problem.

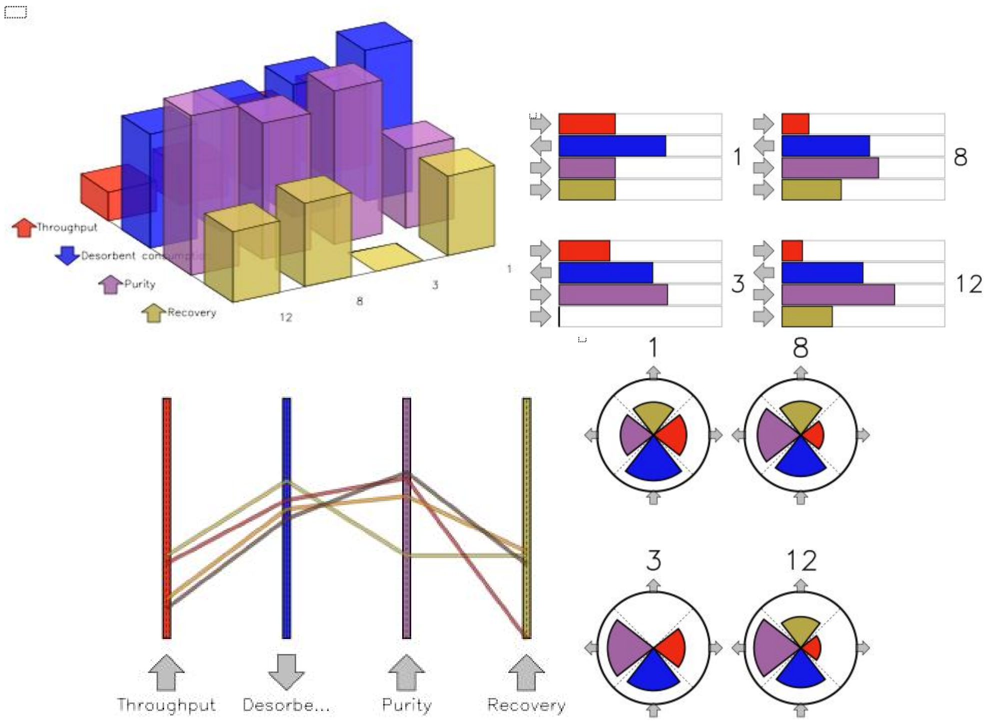

Figure 3: Different visualizations of solutions obtained by IND-NIMBUS.

The results show that a number of interesting SMB designs can be obtained through this approach. With our tools, this computationally very demanding SMB problem could be succesfully handled and new insight obtained. We were able to combine independently developed efficient optimizers (IND-NIMBUS and IPOPT) and obtain novel results for SMB problems without unnecessary simplifications by restricting the number of objectives. Therefore, this can be seen as a significant first step and a good starting point for following studies.

For future work, we intend to apply IND-NIMBUS to the superstructure formulation of SMB instead of the standard formulation studied in this paper. Kawajiri and Biegler reported a better trade-off in the bi-objective optimization study (throughput and desorbent consumption as objectives) with the superstructure formulation than other formulations [8]. In addition, a superstructure SMB formulation of this problem can generate novel designs [8]. In particular, depending on the emphasis of different objectives (e.g., productivity vs. solvent consumption) the superstructure approach evolves from conventional designs to complex designs with multiple feeds and operating profiles. Moreover, the combination of IPOPT with IND-NIMBUS, allows these designs to be generated and analyzed quickly by the DM. For these reasons, applying IND-NIMBUS to the superstructure SMB would be interesting idea. Using superstructure formulation, the total number of variables is about the same in the standard formulation, but the degrees of freedom will increase by one order of magnitude. In addition, the solution times for the single objective optimization problems with the superstructure formulation increase [8].

# Acknowledgements

The authors wish to thank Dr. Marko M. Mäkelä from the University of Jyväskylä for his valuable comments. Funding from the Jenny and Antti Wihuri Foundation is gratefully acknowledged.

# References

[1] J.T. Buchanan. A Naïve Approach for Solving MCDM Problems: the GUESS Method. *Journal of the Operational Research Society*, 48:202–206, 1997.

[2] G. Dünnebier, J. Fricke, and K.-U. Klatt. Optimal Design and Operation of Simulated Moving Bed Chromatographic Reactors. *Industrial & Engineering Chemistry Research*, 39:2290–2304, 2000.

[3] P.E. Gill, W. Murray, and M.H. Wright. *Practical Optimization.* Academic Press Inc., London, 1981.

[4] J. Hakanen, J. Hakala, and J. Manninen. An Integrated Multiobjective Design Tool for Process Design. *Applied Thermal Engineering*, 26:1393–1399, 2006.

[5] J. Hakanen, K. Miettinen, M.M. Mäkelä, and J. Manninen. On Interactive Multiobjective Optimization with NIMBUS in Chemical Process Design. *Journal of Multicriteria Decision Analysis*, 13:125–134, 2005.

[6] K. Hashimoto, S. Adachi, H. Noujima, and H. Maruyama. Models for the Separation of Glucose/Fructose Mixture Using a Simulated Moving Bed Adsorber. *Journal of Chemical Engineering of Japan*, 16(5):400–406, 1983.

[7] S. Karlsson. *Optimization of a Sequential-Simulated Moving-Bed Separation Process with Mathematical Programming Methods.* PhD thesis, Åbo Akademi University, Åbo, Finland, 2001.

[8] Y. Kawajiri and L.T. Biegler. A Nonlinear Programming Superstructure for Optimal Dynamic Operations of Simulated Moving Bed Processes. *Industrial & Engineering Chemistry Research.* To appear.

[9] Y. Kawajiri and L.T. Biegler. Optimization Strategies for Simulated Moving Bed and Powerfeed Processes. *AIChE J.*, 52(4):1343–1350, 2006.

[10] D. Ko and I. Moon. Multiobjective Optimization of Cyclic Adsorption Processes. *Industrial & Engineering Chemistry Research*, 41(1):93–104, 2002.

[11] K. Miettinen. *Nonlinear Multiobjective Optimization.* Kluwer Academic Publishers, Boston, 1999.

[12] K. Miettinen and M.M. Mäkelä. Interactive Bundle-Based Method for Nondifferentiable Multiobjective Optimization: NIMBUS. *Optimization*, 34:231–246, 1995.

[13] K. Miettinen and M.M. Mäkelä. On Scalarizing Functions in Multiobjective Optimization. *OR Spectrum*, 24:193–213, 2002.

[14] K. Miettinen and M.M. Mäkelä. Synchronous Approach in Interactive Multiobjective Optimization. *European Journal of Operational Research*, 170:909–922, 2006.

[15] H. Nakayama. Aspiration Level Approach to Interactive Multi-Objective Programming and Its Applications. In P.M. Pardalos, Y. Siskos, and C. Zopounidis, editors, *Advances in Multicriteria Analysis*, pages 147–174. Kluwer Academic Publishers, 1995.

[16] D.M. Ruthven and C.B. Ching. Counter-current and Simulated Counter-current Adsorption Separation Processes. *Chemical Engineering Science*, 44:1011–1038, 1989.

[17] H. Submarani, K. Hidajat, and A. Ray. Optimization of Reactive SMB and Varicol Systems. *Computers & Chemical Engineering*, 27:1883–1901, 2003.

[18] A. Toumi, F. Hanisch, and S. Engell. Optimal Operation of Continuous Chromatographic Processes: Mathematical Optimization of the VARICOL Process. *Industrial and Engineering Chemistry Research*, 41:4328–4337, 2002.

[19] A. Wächter and L.T. Biegler. On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming. *Mathematical Programming*, 106:25–57, 2006.

[20] A.P. Wierzbicki. A Mathematical Basis for Satisficing Decision Making. *Mathematical Modelling*, 3(25):391–405, 1982.

[21] A.P. Wierzbicki. Reference Point Approaches. In T. Gal, T.J. Stewart, and T. Hanne, editors, *Multicriteria Decision Making: Advances in MCDM Models, Algorithms, Theory, and Applications*, pages 9–1–9–39. Kluwer Academic Publishers, 1999.