# Monotonic Basin Hopping for the separation of acid gases from crude natural gas

Bernardetta Addis, Marjan Bozorg, Christophe Castel, Eric Favre, Veronica Picciali

# Monotonic Basin Hopping for the separation of acid gases from crude natural gas

B. Addis, M. Bozorg, C. Castel, E. Favre, V. Piccialli

## 1 Introduction

Membrane separation processes from being a novelty in separation methods have been increasingly important in the chemical process industries through the last decades [2, 3].

Among the numerous industrial applications of membrane separation processes, gas permeation is one of the most promising one. The development of an accurate and reliable design strategy is essential for optimizing the performances and the global costs of the process. Therefore, a rigorous mathematical model aimed at optimizing the process architecture and the operating conditions of the membrane system is necessary.

A number of recent publications focus on optimization techniques for designing multi-stage membrane system architectures with the aim of minimizing the total cost of system.

In [4], the cost minimization of the membrane separation configuration problem is modeled as a Mixed Integer Non-Linear Programming (MINLP) problem. The solution method is a combination of optimization and simulation, where a maximum of three stages is considered. The optimization strategy is a simulated annealing algorithm with an exploration of the neighborhood based on structure and stream moves. The structure moves change the structure of the network (membrane areas, number of stages, flow configurations), while the stream moves only change the flow distribution. Therefore, the exploration is both in terms of continuous and discrete variables. For each new generated point, once integer variables are fixed, the nonlinear equations describing the system behavior are imposed through a standard nonlinear equation solver. Then, this point is simulated, the cost function is computed, and the corresponding solution is accepted or rejected. Performance constraints are not directly inserted in the model as constraint, but enforced through a penalty term, therefore, solutions found by the optimization phase could be unfeasible. Different case studies are considered: air separation using vacuum pump with different percentages of $CO_2$, and $CH_4$ recovery from synthesis gas and refinery stream.

In [5], a multi-objective optimization model is formulated to simultaneously minimize energy consumption and membrane area of given single and two-stages membrane systems. The optimization model is solved by a genetic algorithm. Operating pressure, temperatures and intermediate composition have been considered as optimization variables.

In [6], a MINLP model is used to determine the minimum cost of multi-stage membrane configurations, and is solved by means of the BARON solver. Process layout, operating variables, and membrane material are included as decision variables, and the effect of feed conditions and membrane material on the process economics is also investigated. Unfortunately, the model description (both cost function and design description) is incomplete, so that we could not compare directly to their cost model.

Similarly, in [7], a MINLP model is proposed to optimize a two stage membrane system configuration for nitrogen removal from natural gas with different feed compositions, and again solved by BARON. Also, the trade off between selectivity and permeability in systems with different membranes was discussed. Here, the optimization variables are the membrane area, the flow distribution, and the pressure down.

In [8] a NLP (nonlinear programming problem) optimization model is proposed to design multi-stage membrane systems with minimum total cost. The process layout and operating variables are considered as decision variables, as well as the optimal number of stages. Indeed, they carefully model the equations on the maximum number of allowed stages, showing that they can get rid of the discrete variables describing the configuration. The NLP problem is solved by means of a local solver, using some heuristic to help generate a good starting solution with respect of feasibility. The impact of CO2 purity on optimal number of stages of a system is investigated.

In [9], a multi-objective optimization model is defined and solved by means of a genetic algorithm to minimize the compression energy and required membrane area. Authors consider as variables the membrane selectivity, the pressure up and the flow distribution, on four given design configurations.

Authors analyze the behavior of decision variables along the Pareto front.

In [1] a membrane system configuration to separate multicomponent gas mixtures is addressed, based on an approximate permeator model and MINLP model. Different multi-stage membrane configurations with discrete and continues variable area are presented. Process layout and area are the main optimization variables. The separation of acid gases ($CO_2$ and $H_2S$) from crude natural gas is also investigated as case study, which is exactly the case we consider in this paper. They have fixed pressures, and vary the membrane area and the flow distribution.

Although a range of mathematical models describing the performance of multistage membrane systems has been developed over recent years finding an optimal configuration requires considering different aspects of a multi-stage membrane system and it is still an open problem from the optimization point of view.

## 2 Comparing results of [1] and local optima of our model using COFE

Before starting using a global optimization method on our mathematical model, we want to validate the solutions presented in [1] using COFE. The idea is to try to validate the differences between the two models, to be sure that our improvement does not derive simply from differences in the optimization model.

Let us consider the configurations depicted in Figure2 (2 membranes) and Figure4 (3 membranes) of [1], we will refer to these case studies as Fig2 and Fig4. The configuration presented in the paper will be represented by the variables $x_p$, with corresponding objective value $C^p$ (as reported in the reference paper [1]). We made two different tests (see Figure 1 for a schematic representation):

- Insert the configuration represented by $x_p$ (membrane areas, flow splits, ...) in the COFE simulator, the corresponding equilibrium solution is called $x_{COFE}$. Then, we calculate the objective value corresponding to $x_{COFE}$, called $C_{COFE}$).
- Use $x_p$ as starting point for a **local** optimization step of our optimization model. The corresponding local optimum solution is represented

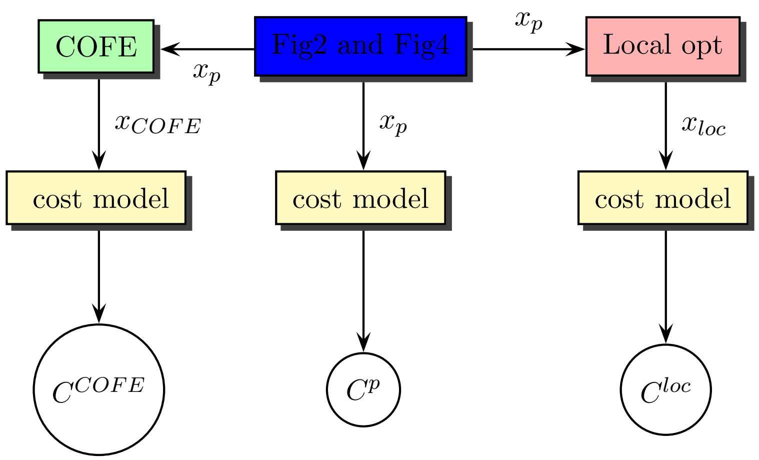

Figure 1: Procedure to validate the local model through simulation in COFE

by $x_{loc}$ and its objective value $C_{loc}$.

In this validation test, the design is completely fixed ( and corresponding to the one given as solution in [1]): membranes types, their connection (for splits also percentages are kept as given values) and feed (flow and composition), presence of compressors, values of pressures, etc.. Therefore, even if our local optimal solution and the one proposed in the reference paper are slightly different, a unique result using COFE is expected, in fact input variables for COFE (configuration, split values, membranes areas, ...) are parameters for the two models. The difference is due to the sight difference in the two optimization models. Experiments running a global optimization algorithm, namely MBH, support the hypothesis that a single local optimum exists also for our optimization model in this case.

In Table 1, the objective values $C^p$, $C^{COFE}$, $C^{loc}$ and the main elements of the vectors $x_p$, $x_{COFE}$, $x_{loc}$ for the case-study with 2-stages (and 3-stages) are reported. The differences in the flows and consequently in the objective values confirm that the two optimization models slightly differ, and that there is a difference between their results and COFE (even if our optimization models seems to obtain results closer to the ones found with the COFE simulation). For this reason, we will always consider the results of the simulation and not of the optimization itself for what concern the final quality of the solution. Nevertheless, a first comparison is made also

| Case Name |  | Cost | Flow |  | $CO_2$ |  |
|---|---|---|---|---|---|---|
|  |  |  | RET | PERM | RET | PERM |
| 2-stage | $x_p$ | 11.112 | 7.222 | 2.778 | 0.0200 | 0.6319 |
|  | $x_{COFE}$ | 11.660 | 7.023 | 2.977 | 0.0095 | 0.6159 |
|  | $x_{loc}$ | 11.500 | 7.124 | 2.876 | 0.0121 | 0.6307 |
| 3-stage | $x_p$ | 11.080 | 7.140 | 2.860 | 0.0200 | 0.6145 |
|  | $x_{COFE}$ | 11.807 | 7.026 | 2.974 | 0.0102 | 0.6147 |
|  | $x_{loc}$ | 11.300 | 7.252 | 2.748 | 0.0123 | 0.6591 |

Table 1: Objective value and main output variables for validation

on the result of the optimization procedure. Therefore, reference values for the cost function and for flows and gas percentages will be the ones obtained in the fixed case (membrane areas and flows) with our optimization model when we compare the solution of the optimization phase. On the contrary, to evaluate the "real" quality of the solution, we refer to the corresponding values returned by COFE using as initialization point the optimal solutions reported in [1]. This is the target we are trying to improve.

## 3 The optimization model

We define the following sets to generalize our model to a generic number of gas componenents and number of membranes:

- $\mathcal{S}$ is the set of states (membranes)
- $\mathcal{C}$ is the set of discretization cells in a membrane
- $C$ is the set of gas components

We define the following variables:

- Overall system
  - $Feed$ is the input flow to the system
  - $F^{Ret}$ (and $F^{Perm}$) system retentated (and permeated) flow
- Single membrane
  - $f_s$ is the input flow to the membrane $s$
  - $f_s^{ret}$ (and $f_s^{perm}$) membrane $s$ retentated (and permeated) flow
  - $x_{s,j}^{ret}$ fraction of retentated flow of membrane $s$ that is constituted by
  - $x_{s,j}^{perm}$ fraction of permeated flow of membrane $s$ that is constituted by
- Single cell
  - $g_{si}$ is the input flow to the cell $i$ of membrane $s$
  - $g_{si}^{ret}$ (and $g_{si}^{perm}$) cell $i$ of membrane $s$ retentated (and permeated) flow
  - $y_{s,j,i}$ fraction of the in flow of cell $i$ of membrane $s$ that is constituted by component $j$
  - $y_{s,j,i}^{perm}$ fraction of permeated flow of cell $i$ of membrane $s$ that is constituted by component $j$
  - $y_{s,j,i}^{ret}$ fraction of retentated flow of cell $i$ of membrane $s$ that is constituted by component $j$

$$Feed = F^{Ret} + F^{Perm} \tag{1}$$

$$f_s = (f_s^{ret} + f_s^{perm}) \qquad \forall s \in \mathcal{S} \tag{2}$$

$$g_{s,i} = (g_{s,i}^{ret} + g_{s,i}^{perm}) \qquad \forall s \in \mathcal{S}, i \in \mathcal{C} \tag{3}$$

$$g_{s,i} y_{s,j,i} = (g_{s,i}^{ret} y_{s,j,i}^{ret} + g_{s,i}^{perm} y_{s,j,i}^{perm}) \qquad \forall s \in \mathcal{S}, i \in \mathcal{C}, j \in C \tag{4}$$

$$g_{s,i+1} = g_{s,i}^{ret} \qquad \forall s \in \mathcal{S}, i \in \mathcal{C} \tag{5}$$

$$y_{s,j,i+1} = y_{s,j,i}^{ret} \qquad \forall s \in \mathcal{S}, i \in \mathcal{C}, j \in C \tag{6}$$

$$g_{s,i}^{perm} y_{s,j,i}^{perm} = Acell_s P_j (p^{up} y_{s,j,i}^{ret} - p^{down} y_{s,j,i}^{perm}) \qquad \forall s \in \mathcal{S}, i \in \mathcal{C}, j \in C \tag{7}$$

$$f_s = g_{s,1} \qquad \forall s \in \mathcal{S} \tag{8}$$

$$x_{s,j} = y_{s,j,1} \qquad \forall s \in S, j \in C \tag{9}$$

$$f_s^{ret} = g_{s,n_s}^{ret} \qquad \forall s \in S \tag{10}$$

$$x_{s,j}^{ret} = y_{s,j,n_s}^{ret} \qquad \forall s \in S, i \in C \tag{11}$$

$$f_s^{perm} = (\sum_{i \in 1..n_s} g_{s,i}^{perm}) \qquad \forall s \in \mathcal{S}, i \in \mathcal{C} \tag{12}$$

$$x_{s,j}^{perm} f_s^{perm} = (\sum_{i \in 1..n_s} y_{s,j,i}^{perm} g_{s,i}^{perm}) \qquad \forall s \in \mathcal{S}, j \in C \tag{13}$$

$$\sum_{j \in C} X_j^{Perm} = 1 \tag{14}$$

$$\sum_{j \in C} x_{s,j}^{ret} = 1 \qquad \forall s \in \mathcal{S} \tag{15}$$

$$\sum_{j \in C} x_{s,j}^{perm} = 1 \qquad \forall s \in \mathcal{S} \tag{16}$$

$$\sum_{j \in C} y_{s,j,i}^{ret} = 1 \qquad \forall s \in \mathcal{S}, i \in \mathcal{C} \tag{17}$$

$$\sum_{j \in C} y_{s,j,i}^{perm} = 1 \qquad \forall s \in \mathcal{S}, i \in \mathcal{C} \tag{18}$$

$$Feed = \sum_{s \in \mathcal{S}} g_s^{sFeed,flow} \quad \forall s \in S, i \in C \tag{19}$$

$$\begin{aligned} f_s^{ret} &= \\ \sum_{s_1 \in s+1..ns} f_{s,s1}^{forward,ret} + f_s^{out,ret} + \sum_{s1 \in 1..s} f_{s,s1}^{loop,ret} & \quad \forall s \in \mathcal{S} \end{aligned} \tag{20}$$

$$\begin{aligned} f_s^{perm} &= \\ \sum_{s_1 \in s+1..ns} f_{s,s1}^{perm,ret} + f_s^{out,perm} + \sum_{s1 \in 1..s} f_{s,s1}^{loop,perm} & \quad \forall s \in \mathcal{S} \end{aligned} \tag{21}$$

$$\begin{aligned} x_{s,j}^{ret} f_s^{ret} &= \\ \sum_{s1 \in s+1..ns} x_{s,j}^{ret} f_{s,s1}^{forward,ret} & \\ + x_{s,j}^{ret} f_s^{out,ret} & \\ + \sum_{s1 \in 1..s} x_{s,j}^{ret} f_{s,s1}^{loop,ret} & \quad \forall s \in \mathcal{S}, j \in C \end{aligned} \tag{22}$$

$$\begin{aligned} x_{s,j}^{perm} f_s^{perm} &= \\ \sum_{s1 \in s+1..ns} x_{s,j}^{perm} f_{s,s1}^{forward,perm} & \\ + x_{s,j}^{perm} f_s^{out,perm} & \\ + \sum_{s1 \in 1..s} x_{s,j}^{perm} f_{s,s1}^{loop,perm} & \quad \forall s \in \mathcal{S}, j \in C \end{aligned} \tag{23}$$

$$F^{Ret} = \sum_{s \in states} f_s^{out,ret} \quad \forall s \in \mathcal{S} \tag{24}$$

$$F^{Perm} = \sum_{s \in states} f_s^{out,perm} \quad \forall s \in \mathcal{S} \tag{25}$$

$$X_j^{Ret} F^{Ret} = \sum_{s \in \mathcal{S}, j \in C} x_{s,j}^{ret} f_s^{out,ret} \quad \forall s \in \mathcal{S}, j \in C \tag{26}$$

$$X_j^{Perm} F^{Perm} = \sum_{s \in \mathcal{S}, j \in C} x_{s,j}^{perm} f_s^{out,perm} \quad \forall s \in \mathcal{S}, j \in C \tag{27}$$

## 4 Global optimization strategy

We applied a continuous global optimization algorithm, that can be considered as the composition of two algorithms: Multistart and Monotonic Basin Hopping (MBH). Multistart and MBH can be considered as the basis of many other elaborated Global Optimization (GO) heuristics. Multistart is a modified version of the Monte-Carlo (MC) method: points are randomly sampled in the search space and then, differently from MC methods, a local optimization is performed on each point. This allows to combine a global search (random sampling) to the power of standard local optimization (gradient-based, quasi-newton, etc.) in determining local optima. This key idea of combining global and local search is used by almost all successful GO stochastic algorithms. MBH combines the idea of using a standard local search with the idea that *near* a good local solution it could be possible to find some better local solution (as long as this is not the global optimum). Therefore, instead of restarting every time from a new different point like Multistart, MBH restricts its search in a neighborhood of the current local solution.

The scheme of MBH is reported in Algorithm 1. Note that in the algorithm $\mathcal{L}(x)$ denotes the point resulting from performing a local search starting from the point $x$. MBH starts randomly generating a point and applying a local optimization, then this first local solution is considered the center of search in a reduced space (the neighborhood of the current solution). Therefore, the current local optimum is perturbed and locally optimized. If the new solution is better then the previous one, this step is considered an improvement step, and the new local solution substitutes the previous one, otherwise another perturbation is applied to the old local solution. The algorithm continues until no improvement is found for a certain number of consecutive perturbations[1]. The whole procedure is repeated starting from different initial points of the search space, therefore we can consider our GO method as the composition of the function Multistart with the function MBH. In Algorithm 2 a very schematic representation of the overall GO method is reported.

MBH (as Multistart) is a meta-heuristic, and therefore must be adapted at the problem we want to solve. Furthermore, both MBH and Mulstistart

[1] if an improvement is found, the number of unsuccessfully perturbation is re-set to zero

**Algorithm 1** `Monotonic Basin Hopping`

```
 1: procedure MONOTONIC BASIN HOPPING($\Delta$,$N$,$x_{start}$)
 2:     $n = 0, k = 0$
 3:     $x_0 = x^* = \mathcal{L}(x_{start})$
 4:     while $n < N$ do
 5:         $y_k$ = random uniform point in $\mathcal{B}(x_k, \Delta)$
 6:         if $C(\mathcal{L}(y_k)) < C(x^*)$ then
 7:             $n = 0$
 8:             $x^* = x_{k+1} = \mathcal{L}(y_k)$
 9:         else
10:             $n = n + 1$
11:         $k = k + 1$
        return $x^*, C(x^*)$
```

**Algorithm 2** `Global Optimization Algorithm`

```
 1: $t = 0$,$x^{min}$, $C^{min} = \infty$
 2: while $t < maxT$ do
 3:     $x_t^{start}$ = random uniform point in the feasible set
 4:     $x_t^*, C(x_t^*)$ =Monotonic Basin Hopping($\Delta$,$N$,$x_t^{start}$)
 5:     if $C(x_t^*) < C^{min}$ then
 6:         $C^{min} = C(x_t^*)$
 7:         $x^{min} = x_t^*$
 8:     $t = t + 1$
    return $x^{min}$,$C^{min}$
```

are originally conceived for unconstrained or box constrained optimization. When the problem is highly constrained, even finding a feasible point could be challenging. Therefore the method must be tailored to the problem at hand.

We have adapted the algorithm working mainly on the random generation, both for the Multistart (full point generation, Algorithm 2-line 3) and for the MBH random perturbation (Algorithm 1-line 5), to allow the main constraints to be satisfied in the generated/perturbed points. In fact, if this part is not carefully done, the local search procedure could be not able to find a feasible point, or find a point *very far* from the original one, making the search "completely random" and therefore loosing the capacity to search in the neighborhood typical of MBH. Just to give some insight of the tailoring procedure, we explain some of the modification we made.

- We impose that the flow balance constraints are satisfied at the beginning and at the end of each membrane, and that the sum of the percentage of all the components is one.
- Even if the number of variables is high, the independent variables are a limited number. To avoid generation of points that are completely infeasible for mismatch between independent and depend variables, only the independent ones are randomly generated or perturbed, and the others are obtained through a simplified step of local optimization.
- Furthermore, since we know that solution with small flows are not realizable and may cause problems in simulating the design in COFE, whenever we find a stationary point with small values of flows (less than $10^{-3}$) we fix the small flows to zero and re-optimize. Therefore, in our modified version of MBH, the function $\mathcal{L}(x)$ can be considered as an operator that maps the point $x$ to the "nearest" local minimum with no "small flows".

The best results produced for each configuration by our GO algorithm are then simulated in COFE in order to get the "real" values for each variable. Then, the objective function is calculated[2].

# 5 Global optimization of 2-stages and 3-stages cases keeping fixed pressures

We performed a set of different tests (see Figure 2), adding step by step more degrees of freedom. We recall that we have two case-studies: 2 stages, using as reference Fig2 and 3 stages, using as reference Fig4. For each of them, we have considered two different settings with respect to membrane areas:

- membrane areas are fixed to the values reported in [1]

[2] from a practical point of view, and to reduce possible errors due to different ways to evaluate the cost, the objective is calculated by plugging the resulting point of the COFE simulation in the optimization code and just calling a routine to evaluate the cost without performing any optimization.

- membrane areas are continuous variables, that is any area can be chosen in a given interval

and two different cases considering flows among membranes:

- the flow connections are kept fixed to the values reported in [1], meaning that we only allow nonzero flows on the connections reported in [1].
- flows are continuous variables, so we consider a general configuration where all mixers and splits can be used

We can observe, that the case with fixed areas and fixed flows is exactly the reference value, in fact in this case only one stationary point exists from the simulation point of view (and also from the optimization one). Therefore, at this step, we consider 3 global optimization problems for each case-study. In Figure 2, we report a schematic representation of the tests, from the reference configuration to the one with more degree of freedom.

$$\begin{array}{l} 2-\text{stages}\begin{cases}\text{Fixed Area}\begin{cases}\text{Fixed Flow}-\textbf{Reference}\\ \text{Free Flow}\end{cases}\\ \text{Free Area}\begin{cases}\text{Fixed Flow}\\ \text{Free Flow}\end{cases}\end{cases}\\ 3-\text{stages}\begin{cases}\text{Fixed Area}\begin{cases}\text{Fixed Flow}-\textbf{Reference}\\ \text{Free Flow}\end{cases}\\ \text{Free Area}\begin{cases}\text{Fixed Flow}\\ \text{Free Flow}\end{cases}\end{cases}\end{array}$$

Figure 2: **Summary of tests with fixed pressures**

We stress that in the first set of experiments, we keep the pressures for each membrane equal to the ones used in [1]. In analogy with the notation used for the validation of the model, we refer to the best solution found by our global optimization procedure as $x_{best}$ and its objective value as $C^{best}$. We always consider as reference for evaluating the quality of solutions the objective value and the point found with the COFE simulation (using as starting point the result of our optimization procedure).

In Table 2, the best results found by our GO algorithm, and then simulated by COFE are reported for all the considered case-studies and degrees of freedom.

In Table 3, some details on the obtained solution points are reported. We

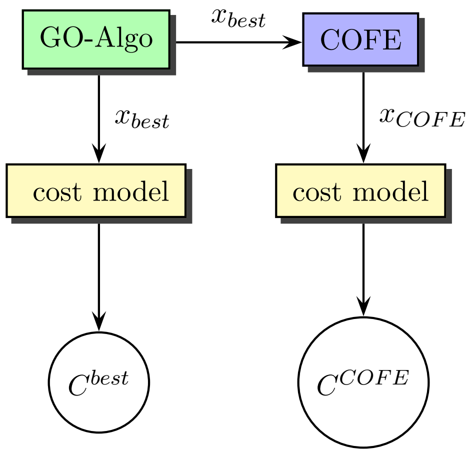

Figure 3: Procedure to validate the solutions found by the GO algorithm through simulation in COFE

| | Case Name | Reference | Flow configurations | | |
|---|---|---|---|---|---|
| | | | variable | fixed | variable |
| | | | Fixed area | Variable area | |
| GO | 2-stages | 11.094 | 11.121 | 9.502 | 9.182 |
| | 3-stages | 10.971 | 10.721 | 9.288 | 9.115 |
| COFE | 2-stages | 11.660 | 11.209 | 9.460 | 9.161 |
| | 3-stages | 11.807 | 10.681 | 9.259 | 9.095 |

Table 2: Objective values obtained by GO and simulating in COFE the results of GO

can observe that the percentage of $CO_2$ is always below the demanded value of 2%. In our optimization, we did not fix this value at 2%, but imposed that the values stays under this threshold, therefore different values can be possible. If the area is free to change, even considering the same configuration, better configurations (lower overall cost) can be found, and they corresponds to a reduced area.

In Figures 4-5 are reported the best configuration leaving all degree of freedom (except for pressure that is kept fixed at the values chosen in the reference paper) for the case with 2 and 3 stages, respectively. If we compare these configurations with the ones presented in the reference paper, we can

| | Variables | Reference | Flow configurations | | |
|---|---|---|---|---|---|
| | | | variable | fixed | variable |
| | | | Fixed area | Variable area | |
| 2-stages | Ret | 7.222 | 7.295 | 7.429 | 7.449 |
| | %$CO_2$ | 2.0 | 1.3 | 1.7 | 1.8 |
| | Area1 | 222.92 | 222.92 | 166.89 | 165.41 |
| | Area2 | 164.45 | 164.45 | 159.75 | 148.23 |
| | Total Area | 387.37 | 387.37 | 326.64 | 313.64 |
| 3-stages | Ret | 7.348 | 7.731 | 7.534 | 7.557 |
| | %$CO_2$ | 2.0 | 1.7 | 1.7 | 1.8 |
| | Area1 | 182.75 | 182.75 | 131.56 | 127.91 |
| | Area2 | 197.92 | 197.92 | 183.26 | 180.72 |
| | Area3 | 13.33 | 13.33 | 13.74 | 14.00 |
| | Total Area | 394.01 | 394.01 | 328.56 | 322.63 |

Table 3: Details on solution obtained simulating in COFE the results of GO

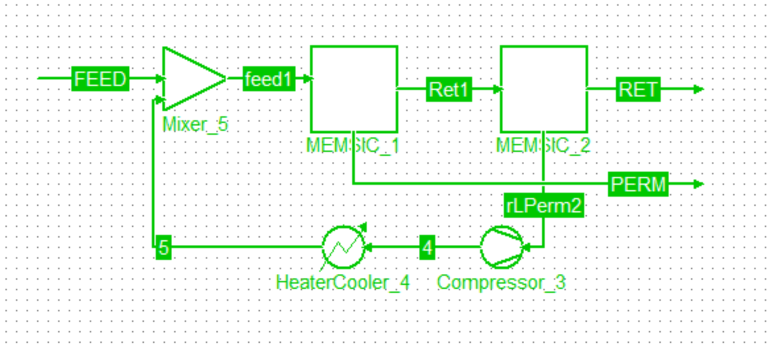

Figure 4: Best configuration for 2 stages - pressures are fixed (area and configuration are free)

observe that the main design is the same, the main difference is given by the membrane areas (total and repartition), and as consequence, in the relatives quantities of flows. Even if the same flows are present, also the choice of repartition of flows is different.

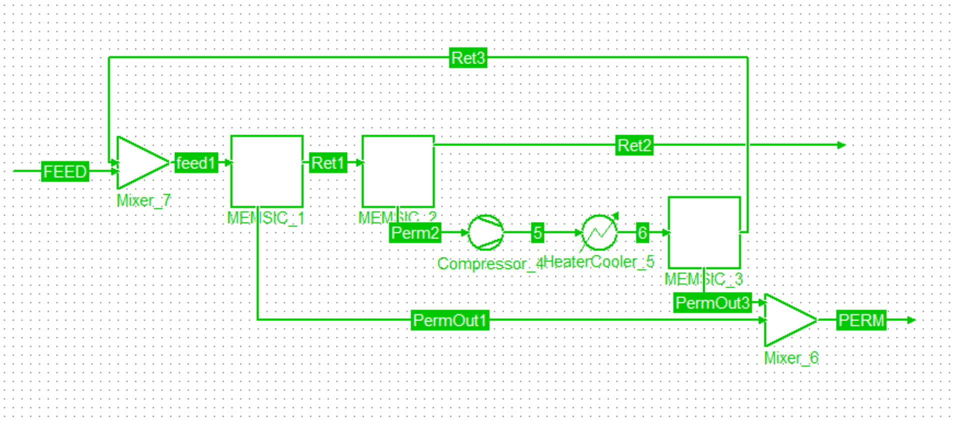

Figure 5: Best configuration for 3 stages - pressures are fixed (area and configuration are free)

# 6 Global optimization of 2-stages and 3-stages membrane cases allowing pressure to change

We performed a second set of experiments, where we consider again the two stages and three stages systems for the case proposed in [1], but with a further degree of freedom. Indeed, we allow the GO algorithm to determine the membrane areas, the flow distribution and also the flow pressures. In particular, we assume that each membrane has the same pression_up and pression_down (called $p_{\text{up}}$ and $p_{\text{down}}$), but these two values are considered variables and may vary in a bounded region. To be more precise, we impose the following constraints:

1. $p_{\text{up}} \in [l_p, u_p]$
2. $p_{\text{down}} \geq l_p$
3. $p_{\text{up}} \geq p_{\text{down}} + \epsilon$

We choose $l_p = 0.01$ bar, we set $\epsilon = 0.001$ and we try three different values for the upper bound on the pressure of the upstream flow ($u_{\text{up}} = 35, 50, ...$). It is important to notice that when $u_{\text{up}} > 35$, a compressor is added at the inlet flow (in fact the compound gas pressure is assumed to be 35 bar) and its cost is added to the overall objective function.

| Case Name | | Reference | Fixed | $u_p = 35$ | $u_p = 50$ |
|---|---|---|---|---|---|
| GO | 2-stages | 11.094 | 9.182 | 9.075 | 7.312 |
| | 3-stages | 10.971 | 9.115 | 9.044 | 7.271 |
| COFE | 2-stages | 11.660 | 9.161 | 9.057 | 7.307 |
| | 3-stages | 11.807 | 9.095 | 9.032 | 7.277 |

Table 4: Objective values obtained by GO and simulating in COFE the results of GO with free pressures

In Table 4, the cost values for the obtained configurations are shown, considering as usual both the point produced by the GO algorithm and the point given by the corresponding simulation in COFE. We report in the table a column for the reference value (from [1]) and also the value obtained with all the degree of freedom but the pressures. The results show that allowing the change of the pressures (even keeping the same pressures for all membranes) produces solutions with a significantly lower cost then keeping them fixed at the values considered in [1].

In Table 5, details on the best solution found by GO after validation with COFE are reported.

|  | Variables | Reference | Pressures | $u_p = 35$ | $u_p = 50$ |
|---|---|---|---|---|---|
|  |  |  | Fixed | $l_p = 0.01$ | $l_p = 35$ |
| 2-stages | Ret | 7.222 | 7.449 | 7.306 | 6.659 |
|  | %$CO_2$ | 2.0 | 1.8 | 1.7 | 1.7 |
|  | PressionUp | 35 | 35 | 35 | 50 |
|  | PressionDown | 1.05 | 1.05 | 1.31 | 2.36 |
|  | Area1 | 222.92 | 165.41 | 193.73 | 116.23 |
|  | Area2 | 164.45 | 148.23 | 136 | 110.19 |
| 3-stages | Ret | 7.348 | 7.557 | 7.306 | 6.659 |
|  | %$CO_2$ | 2.0 | 1.8 | 1.7 | 1.7 |
|  | PressionUp | 35 | 35 | 35 | 50 |
|  | PressionDown | 1.05 | 1.05 | 1.31 | 2.36 |
|  | Area1 | 182.75 | 127.91 | 106.21 | 86.60 |
|  | Area2 | 197.92 | 180.72 | 97.92 | 80.52 |
|  | Area3 | 13.33 | 14.00 | 123.18 | 60.22 |

Table 5: Details on solutions of best GO algorithm solutions validated by COFE for configurations with different pressures

# References

[1] R. Qi, M.A. Henson, Membrane system design for multicomponent gas mixtures via mixed-integer nonlinear programming, *Computers and Chemical Engineering*, vol. 24, pp 2719–2737 (2000).

[2] I. Marriott, Detailed mathematical modelling of membrane modules, *Computers and Chemical Engineering*, vol. 25, p. 693–700 (2001).

[3] J. A. Caballero, I. E. Grossmann, M. Keyvani and E.S. Lenz, Design of Hybrid Distillation-Vapor Membrane Separation Systems,*Industrial Engineering Chemistry*, vol. 48, p. 9151–9162 (2009).

[4] R. V. S. Uppaluri, P. Linke and A. C. Kokossis, Synthesis and Optimization of Gas Permeation Membrane Networks, *Industrial Engineering Chemistry*, p. 4305–4322 (2004).

[5] M. Yuan, K. Narakornpijit, R. Haghpanah, J. Wilcox, Consideration of a nitrogen-selective membrane for post combustion, *Membrane Science*, vol. 465, p. 177184 (2014).

[6] M. Scholz, M. Alders, T. Lohaus, M. Wessling, Structural optimization of membrane-based biogas,*Membrane Science*, vol. 474, p. 1–10 (2015).

[7] B. Ohs, J. Lohaus, M. Wessling, Optimization of membrane based nitrogen removal from natural gas, *Membrane Science*, vol. 498, p. 291-301 (2016).

[8] A. M. Arias, M. C. Mussati, P. L. Mores, N. J. Scenna, J. A. Caballero, S. F. Mussati, Optimization of multi-stage membrane systems for CO2 capture from flue gas, *International Journal of Greenhouse Gas Control*, vol. 53, p. 371-390 (2016).

[9] P. Gabrielli, M. Gazzani, M. Mazzotti, On the optimal design of membrane-based gas separation processes, *Membrane Science*, vol. 526, p. 118-130 (2017).