A Tabu Search-based algorithm for the integrated process and control system design
Proceedings of European Congress of Chemical Engineering (ECCE-6)
Copenhagen, 16-20 September 2007

A Tabu Search-based algorithm for the integrated process and control system design

O. Exler,[a] N. Faisca,[b] L. T. Antelo,[c] A. A. Alonso,[c] J. R. Banga,[c] E. N. Pistikopoulos[b]

[a]*Department of Computer Sience, University of Bayreuth, 95440 Bayreuth, Germany*
[b]*Department of Chemical Engineering, Imperial College London, Centre for Process Systems Engineering, Roderic Hill Building, South Kensington Campus, London SW7 2AZ, United Kongdom*
[c]*Process Engineering Group, IIM-CSIC, Eduardo Cabello 6, 36208 Vigo, Spain*

Abstract

In this contribution we focus on the area of integration of process and control system design. The interaction of process design and control is formulated as a mixed-integer dynamic optimization (MIDO) problem and by using the control parameterization approach we obtain as a result a finite dimensional mixed-integer non-linear programming (MINLP) problem. Because of the multimodal structure of these problems we use a global optimization strategy. The hybrid algorithm MITS is presented, which uses a combinatorial component, based on Tabu Search, to localize promising attraction basins, and a local solver, which is activated whenever such a promising area is located. Numerical results are presented for the optimization of the well-known Tennessee Eastman Process (TEP).

Keywords: Integrated Process and Control System Design, Mixed-Integer Nonlinear Programming, Metaheuristic, Tabu Search

1. Introduction

During the last decade, the importance of a simultaneous (integrated) process design approach, considering operability together with the economic issues, has been widely recognized (Morari and Perkins, 1994; Pistikopoulos and Ross, 1999; Bansal et al, 2000; Sakizlis et al, 2004). This paper focuses on the area of integration of process and control system design. The aim is to obtain profitable and operable process and control structures in a systematic way. Both, the process design characteristics, control strategies, control structure and controllers' tuning parameters will be optimally selected in order to minimize the total annualized cost of the system while satisfying a large number of feasibility constraints in the presence of time-varying disturbances.

The interaction of process design and control can be formulated as a mixed-integer dynamic optimization (MIDO) problem. There are different approaches to solve this MIDO problem, such as dynamic programming, control parameterization and complete discretization. This paper focuses on the control parameterization approach, obtaining as a result a finite dimensional mixed-integer non-linear programming (MINLP) problem. It should be noted that the optimization problems arising from these formulations are very challenging. The multimodal (non-convex) nature of these problems has been highlighted by e.g. Schweiger and Floudas (1997) and Bansal et al. (2000), among others. Because of the multimodal structure of these problems the use of global optimization (GO) strategies is mandatory.
The GO of nonlinear dynamic systems is receiving increased attention from engineers, mathematicians and computer scientists. In the domain of deterministic GO methods, Esposito and Floudas (2000) have presented approaches to solve nonlinear optimal control (dynamic optimization) and parameter estimation problems. This is indeed a very promising and powerful approach, but the objective function and the dynamics of the system must be twice continuously differentiable, and restrictions may also apply for the type of path constraints, which can be handled.
Regarding stochastic GO methods, a number of researches have shown that they can locate the vicinity of global solutions for nonlinear dynamic problems with relative efficiency (Banga et al, 2003), but the cost to pay is that global optimality cannot be guaranteed. However, in many practical situations these methods can be satisfactory if they provide us with a "good enough" (often, the best available) solution in modest computation times. Furthermore, they do not require transformation of the original problem, which can be treated as a black box. Thus, they can handle problems with complicated dynamics (e.g. discontinuities, non-smoothness, etc.).
In our case we use the metaheuristic Tabu Search (TS) which was originally developed by Glover (see Glover et al., 1997). Primary used for the optimization of combinatorial problems, in the last years TS also has been very successfully applied to the optimization of continuous problems. We present an adaptation of TS to solve MINLPs, called Mixed-Integer Tabu Search (MITS).
The paper is structured as follows. The next section will be used to formulate the simultaneous design and control problem. In the following section we present our algorithm in more detail. Since it includes two components (a global and a local component) we dedicate a separate section to each one of them. Finally we present some results we obtained by using our algorithm on the well-known Tennessee Eastman Process (TEP) by Downs and Vogel (1993), which has been widely used in the literature as a case study due to its challenging properties from a control engineering point of view.

2. The simultaneous design and control problem

In order to represent the interaction of process design and control the formulation of the problem has to combine components that express both design alternatives and the operability of the system. In general a superstructure is developed that contains possible alternatives that one wants to consider. A key element that contributes significantly in the calculation of optimal and sensible design solutions is the good definition of the design space and more specifically of the design superstructure.

Measuring the controllability of the process is achieved by introducing a system of differential and algebraic equations that simulates the behaviour of the process under dynamic operation.
Taking this into account the mathematical formulation is as follows:

$$\begin{aligned} \mathit{min} \quad & J(v, t_f) \\ \mathit{subject\ to} \quad & f(\dot{x}, x, p, v) = 0 \\ & x(t_0) = x_0 \\ & h(x, p, v) = 0 \\ & g(x, p, v) \geq 0 \\ & v^l \leq v \leq v^u \end{aligned} \tag{1}$$

where v is the vector of decision variables, J is the objective function (often representing the costs) to minimize, x is the state vector, f is the set of differential and algebraic equality constraints describing the system dynamics and h and g are possible equality and inequality path and/or point constraints which express additional requirements for the process performance. The lower and upper bounds for the decision variables are given with v^l and v^u.
A superstructure is developed that contains alternatives for the process design. The common way to express alternatives in the design is introducing binary variables. In this way the active state of a design alternative can be easily expressed. A value equal to one stands for an active alternative whereas in the case of zero the alternative is inactive, i.e. not used. Thus, some of the elements of the decision variable vector v can be restricted to integer values. As a result the problem formulated in (1) is a mixed-integer dynamic optimization (MIDO) problem.
Although the resulting problem is very difficult to solve it is worth investigating this kind of problems, since it is the only way to obtain both, a process that has low cost and a process that can be easily controlled. This approach is superior to all methods that only take one aspect into account. Reducing only the cost might result in an instable process, while on the other hand the best process from the controllability point of view can be very expensive. As mentioned before, there are different approaches to solve this MIDO problem. Dynamic programming suffers from the curse of dimensionality. Complete discretization, when applied to problems which contain a significant number of dynamic states (like the examples considered here) results in a large number of decision variables, complicating the application of global optimization methods. As an alternative, we think that the control parameterization approach, which results in a relatively small number of decision variables, facilitates the solution of the resulting MINLP using stochastic global optimization methods, like those based in Tabu search.
The control parameterization technique parameterizes only the control variables, so the decision vector will contain that discretization information plus other selected time invariant parameters. The optimization is carried out in the space of these particular decision variables only. In our work we have followed this technique, i.e. we discretize the control variables and obtain as a result a finite dimensional mixed-integer nonlinear programming (MINLP) problem with a dynamic system embedded, usually as a set of DAEs.

3. The optimization method

Many algorithms for global optimization use a local solver to identify a local minimum by starting from an initial point, and in order to reach the global minimum, a special strategy for deciding where to start the local solver is applied. For convex problems the local solver is able to find the global minimum. If the problem is not convex the global optimality of the solution of the local solver cannot be guaranteed. In this case one has to guide the local solver to the global optimum. This second component of the algorithm has to locate the attraction basin of the global minimum so that a run of the local solver started in this basin will find the global minimum. The approach developed here uses this strategy. As the global component, a procedure based on extensions of the Tabu Search (TS) algorithm is applied. Regarding the local solver, we have used a special adaptation of a sequential quadratic programming method for the mixed-integer case. The following subsection is dedicated to the description of the interaction of the two different components and the basic ideas behind the proposed methodology, whereas the local solver by Exler and Schittkowski (2006), called MISQP, is described in a separate subsection.

3.1. The Tabu Search Algorithm

TS is a metaheuristic originally developed by Glover (see Glover and Laguna, 1997). For the optimization of combinatorial problems, TS has proved to be a very successful strategy. During recent years, it has also been applied to the optimization of continuous problems. Here, an adaptation of TS to mixed-integer nonlinear optimization problems, called Mixed-Integer Tabu Search (MITS), is presented. This algorithm is an enhancement of an approach proposed by Battiti and Tecchiolli (1996). As Battiti and Tecchiolli, plus we also use a local solver to intensify the search if necessary. The aim is to profit from the fast convergence of the local method. In this sense, the local solver MISQP is integrated in the TS framework.

First, let us summarize the basic idea of a tabu search algorithm. The algorithm starts from an initial solution v_k. For this current solution v_k a set of neighbors is generated and the best among these neighbors is chosen to be the next iteration point v_{k+1}, even if the function value is worse than the one of the current iterate v_k. Allowing an increase in the objective function is necessary to escape from a local minimum. To avoid cycling and to guide the search into unexplored areas, some former visited points are set to be tabu and so prohibited for some time. This procedure is repeated by starting from the new current point, until some stopping condition is fulfilled.

The efficiency of the above mentioned procedure depends on the choice of some key parameters, i.e., the length of the period a point is set to be tabu, and this is especially important in the case of continuous variables. Since we do not know anything about the topology of the objective functions, choosing right values for these parameters is non trivial. Ideally, we would like to use a procedure that is independent from such choices. This was the main reason why we decided to use the basic concepts from Battiti and Tecchiolli (1996). These authors proposed a TS algorithm that is robust for any kind of functions and self-adjusting, so that no parameters have to be set a priori. The components will be described now in more detail.

Since there are several successful implementations of TS for the combinatorial case the idea is to transform the continuous problem into a combinatorial one. To achieve this goal we substitute the original variables by boxes, or in other words areas of the original search space. These boxes are identified by unique binary strings and on these strings the combinatorial TS is applied. The neighbors of a box are obtained by sequentially changing one of the bits of the string representing the current box.
The combinatorial search algorithm generates a search trajectory consisting of boxes. Since we are handling boxes we have to decide how to measure the quality of a single box and all the point located in this box. We represent the *box value* by the best objective function obtained by randomly generating points inside the box, because we do not have detailed information about the functions to be optimized. These box values will change during the time since we generate a new point with a uniform probability distribution every time a box is encountered.
In this way the combinatorial component will lead the search to local optimal boxes. When a box is locally optimal, the local solver MISQP will be started to precisely approximate the local optimum of this box. This is the basic procedure. We will now focus on the mentioned self-adjustment.
The initial search region is specified by the bounds on each variable and it is represented as a tree of boxes. The initial region is subdivided into 2^n equal sized boxes by dividing the range in half the initial range of each variable and the representing structure consists consequently of 2^n tree leaves. Since this division can be too crude for the function, the size of the boxes has to be adjusted. We achieve this by dividing the current box as soon as there are two minima detected in the box, i.e. if an additional run of the local solver finds a second local minimum in a box. This box is subdivided into 2^n smaller boxes and the tree is growing. In this way the search will react on the surface of the function. In some areas the search is intensified if there is the possibility that the minima are close to each other because of the multimodal structure of the function in this area. Only the leaves of the tree are considered during the combinatorial search. The leaves partition the initial search space, since the intersection of two leaves is empty and the union of all leaves will be the initial search space. Consequently the number of neighbors of a box can increase during the search.
The second important component that should be self-adjusting is the time a box is tabu. After visiting a box it is always prohibited for the next iteration. The number of iterations, we call this the tabu tenor, during which this box is tabu has to be flexible to ensure an efficient search. If the tabu tenor value is too big some promising areas can be prohibited wrongly and the search can fail the global optimum. Otherwise, if it is too small, the danger of cycling increases. We adjust the value of the tabu tenor according to the search history and we do not need predefine a value for the tabu tenor. Whenever a repetition of a previous encountered box occurs there will be an adjustment of the tabu tenor. The value for the tabu tenor will be increased by the factor *1.1*.
Increasing the tabu tenor will avoid cycling in some areas of the search space, but in other areas this value might be too high and the search can be slowed down unnecessarily. So a second mechanism reduces the tabu tenor to free some prohibited boxes. Every time a number of iterations passed from the last size change, the value is reduced. In this way the search will be efficient in all areas of the initial search space.

Finally, we want to mention when our procedure will stop. Defining a stopping criterion for GO algorithms is very difficult. If the algorithm stops too early, the global optimum can be missed. Otherwise, if it starts too late, computational effort will be wasted. Our procedure will stop if one of the following criterions is fulfilled:

- The search procedure will stop after a predefined maximum number of iterations.
- The procedure will stop if the best objective value is lower or equal to a predefined value.
- The program will terminate after the predefined maximum time has elapsed.

Since we have to handle computational expensive models, time will be the bottleneck. Hence in most cases we will stop our algorithm after a maximal time has elapsed.

3.2. The local solver MISQP

The purpose of this section is to give a short overview of the mathematical theory of the local solver that is activated by MITS. The solver called MISQP (Mixed-Integer Sequential Quadratic Programming) is a SQP Trust-Region method recently developed by Exler and Schittkowski (2006). We consider the general optimization problem to minimize an objective function *f* under nonlinear equality and inequality constraints,

$$
\begin{array}{rl}
 & \min f(x,y) \\
 & g_j(x,y)=0, j=1,\ldots,m_e, \\
x\in\mathbb{R}^{n_c}, y\in\mathbb{Z}^{n_i}: & g_j(x,y)\geq 0, j=m_e+1,\ldots,m, \\
 & x^l\leq x\leq x^u, \\
 & y^l\leq y\leq y^u
\end{array}
$$

where *x* denotes the vector of the continuous and *y* the vector of the integer variables. It is assumed that the problem functions $f(x,y)$ and $g_j(x,y)$, $j=1,\ldots,m$, are continuously differentiable subject to *x* on $\mathbb{R}^{n_c}$. The Hessian of the Lagrangian function - we call it *H* - is approximated by a quasi-Newton update formula subject to the continuous and integer variables. In our implementation the well-known BFGS-Formula is used. It is not assumed that the MINLP is relaxable, i.e., that $f(x,y)$ and $g_j(x,y)$, $j=1,\ldots,;m$, can be evaluated at any fractional parts of the integer variables. Thus, the first derivatives at $f(x,y)$ are approximated by the difference formula

$$
d_y f(x,y) := \frac{f(x,y_1,\ldots,y_j+1,\ldots,y_{n_i}) - f(x,y_1,\ldots,y_j-1,\ldots,y_{n_i})}{2}
$$

for $j = 1,\ldots,n_i$, at neighbored grid points. If either $y_j + 1$ or $y_j - 1$ violates a bound, we apply a non-symmetric difference formula. Similarly, we calculate the first derivatives at *of the constraints*. For the continuous variables the gradients are numerically approximated by a forward difference formula. Instead of a line search as usually applied in the continuous case, a trust region method is used to stabilize the

algorithm and to enforce convergence. We proceed from the continuous trust region method of Yuan (1995) with second order corrections.
The algorithm uses the exact penalty function

$$P_\sigma(x,y) := f(x,y) + \sigma \left\| g(x,y)^- \right\|_\infty ,$$

with a penalty parameter σ and where

$$g_j(x,y)^- := \begin{cases} g_j(x,y), & j \le m_e \\ \min(0, g_j(x,y)), & otherwise \end{cases}.$$

In every iteration the mixed integer quadratic problem (MIQP)

$$d \in \mathbb{R}^{n_c+n_i}: \quad \begin{array}{c} \min \frac{1}{2} d^T H_k d + \nabla f(x_k, y_k)^T d + \sigma_k \left\| \left(\nabla g(x_k, y_k)^T d + g(x_k, y_k) \right)^- \right\|_\infty \\ \|d\|_\infty \le \Delta_k \end{array}$$

has to be solved. One trusts the subproblem in the trust region Δ_k, i.e. it is expected that the MIQP approximates the original problem well inside this trust region. Since the generated subproblems are always convex, we apply a branch-and-bound algorithm implemented by Schick and Schittkowski (2004) to solve this problem. The solution of the quadratic subproblem d_k is used to compute the next iteration point

$$(x_{k+1}, y_{k+1}) = \begin{cases} (x_k, y_k) + d_k, \ if \ P_{\sigma_k}((x_k, y_k) + d_k) \le P_{\sigma_k}(x_k, y_k) \\ (x_k, y_k), \quad otherwise \end{cases}$$

A key role in the trust region algorithm plays the prediction of a new trust region radius Δ_{k+1} for the next iteration. The trust region radius is adjusted according to the quotient of the actual and the predicted improvement of the merit function:

$$r_k := \frac{P_k(x_k, y_k) - P_k((x_k, y_k) + d_k)}{\phi_k(0) - \phi_k(d_k)},$$

with

$$\phi_k(d) := \frac{1}{2} d^T B_k d + \nabla f(x_k, y_k)^T d + \sigma_k \left\| \left(\nabla g_j(x_k, y_k)^T d + g_j(x_k, y_k) \right)^- \right\|_\infty$$

is used to estimate the predicted improvement, i.e., the objective function of subproblem (MIQP).
The trust region radius Δ_k has to be updated in order to enforce convergence. More formally, we use the same constants proposed by Yuan (1995), and set

$$\Delta_{k+1} = \begin{cases} \max[2\Delta_k, 4\|d_k\|_\infty], & if \quad r_k > 0.9 \\ \Delta_k, & if \quad 0.1 \le r_k \le 0.9 \\ \min[\Delta_k/4, \|d_k\|_\infty/2], & if \quad 0 < r_k < 0.1 \end{cases}$$

The algorithm continues until a specified stopping criterion is satisfied.

4. Numerical results - The Tennessee Eastman Benchmark Problem

Since the publication of the Tennessee Eastman Process (TEP) example by Downs and Vogel (1993), it has been widely used in the literature as a case study due to its challenging properties from a control engineering point of view: it is highly nonlinear, open-loop unstable and it presents a large number of measured and manipulated variables which offer a wide set of candidates for possible control strategies. The flowsheet for the TEP is depicted in Figure 1. Two products (*G* and *H*) are produced from four reactants (*A*, *C*, *D* and *E*). A further inert trace component (*B*) and one byproduct (*F*) are present. The process units consist of a continuous stirred tank reactor, a condenser, a flash drum and a stripper. The gaseous reactants are fed to the reactor where they are transformed into liquid products. The following reactions take place in gas phase:

$$A_{(g)} + C_{(g)} + D_{(g)} \rightarrow G_{(l)}$$
$$A_{(g)} + C_{(g)} + E_{(g)} \rightarrow H_{(l)}$$
$$A_{(g)} + E_{(g)} \rightarrow H_{(l)}$$
$$3D_{(g)} \rightarrow 2F_{(l)}$$

These reactions are irreversible and exothermic with rates that depend on temperature through Arrhenius expressions and on the reactor gas phase concentration of the reactants. The reaction heat is removed from the reactor by a cooling bundle. The products and the unreacted feeds pass through a cooler and, once condensed, they enter a vapour-liquid separator. The noncondensed components recycle back to the reactor feed and the condensed ones go to a product stripper in order to remove the remaining reactants by stripping with feed stream. Products *G* and *H* are obtained in bottoms. The inert (*B*) and the byproduct (*F*) are mainly purged from the system as a vapour from the vapour-liquid separator.

Recently, Antelo et al. (2006) applied their systematic approach to a plant-wide control design developed in a previous work (Antelo et al., 2005) to derive robust decentralized controllers for the Tennessee Eastman Process. In this framework, the TEP is represented as a process network. Then, conceptual mass and energy inventory control loops for each node are designed first to guarantee that the states of the plant will remain on a convex invariant region, where the system will be passive and therefore input-output stability can be stated (Antelo et al., 2005). The next step is to realize the proposed conceptual inventory control loops using the physical inputs-outputs of the process. Some extra control loops are needed to achieve the convergence of the intensive variables since the inventory control by itself does not ensure the convergence of these variables to a desired operation point. In some cases, the available degrees of freedom are not enough to implement the complete control

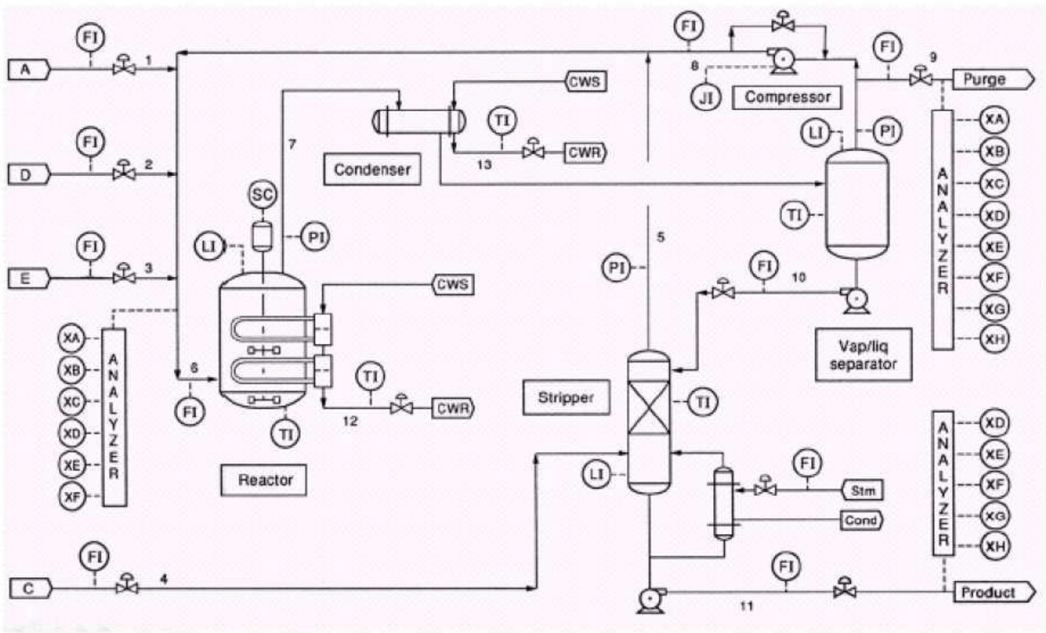

Figure 1 – The Tennessee Eastman Process Flowsheet

structure that ensures both extensive and intensive variables convergence to the reference values. As a consequence, the setpoints of the inventory controllers can be used as new manipulated variables to complete the decentralized control design.

We explain the alternatives we introduced to extend the original hierarchical control design proposed by Antelo et al. (2006). Concerning the reactor level control loop in the original design, its set point modifies the reference for the flow controller acting over *E feed*. As an alternative for closing this loop, *D Feed* is proposed as manipulated variable.

For the reactor pressure case, the original proposal by Antelo et al.(2006) considers the condenser cooling water flow as the manipulated variable. By using this, the reactor pressure can be varied since the separator pressure and, as a consequence, the recycle rate can be modified. It is at this point where the control over the vapor mass inventory in the separator by using the purge rate is established in order to ensure that all inventories in the TEP will remain bounded and, therefore, input-output stability is guaranteed (Antelo et al., 2006).

As an alternative, we are considering here a manipulated variable widely used in the literature for the reactor pressure control loop: the purge flow. By modifying this, it is possible to regulate the separator pressure as well as the recycle flow, and therefore the reactor pressure. When this alternative is considered, an extra loop controlling the separator temperature (energy inventory) by acting over the condenser coolant flow is defined. Note that we are not considering other alternatives to control the reactor pressure as the *A* Feed, that is a disturbance in the model, or *C* Feed.

In order to determine the best control alternative among the proposed ones, a new binary vector b is added to our system dynamics. These 0-1 variables express which of the four control strategies is being used, and they are defined as follows:

$b_1 \in \{0,1\}$ (E feed)

$b_2 \in \{0,1\}$ (D feed)

$b_3 \in \{0,1\}$ (Condenser coolant)

$b_4 \in \{0,1\}$ (Purge flow)

Therefore, the original control design proposal by Antelo et al. (2006) will be characterized by the vector $b = (1, 0, 1, 0)^T$ since it uses *E* Feed to control the reactor level and the condenser coolant flow to control the reactor pressure.
The resulting MINLP can be formulated as follows:

$$\begin{aligned}
min_{v,b} \quad & J(x,v,b) \\
subject\ to \quad & f(\dot{x},x,p,v,b,t)=0 \\
& h(x,p,v,b)=0 \\
& g(x,p,v,b)\geq 0 \\
& b_1+b_2=1 \\
& b_3+b_4-1\geq 0 \\
& v^l \leq v \leq v^u \\
& b^l \leq b \leq b^u
\end{aligned} \qquad (2)$$

where $b \in \{0,1\}^4$ is the vector of binary variables (0-1 variables) and $v \in \mathbb{R}^{36}$ are the continuous variables (the controller parameters). The lower and upper bounds for the binary variables will be of the form $b^l = (0,0,0,0)^T$ and $b^u = (1,1,1,1)^T$. It must be pointed out that we are considering that only one of the two alternatives for each loop can be active at one time, being necessary to introduce the additional linear constraint $b_1 + b_2 = 1$. The linear constraint $b_3 + b_4 - 1 \geq 0$ ensures that at least one of the alternatives b_3 or b_4 is active. The rest of the decision variables are connected to the tuning of the PI controllers. The dynamics (DAEs) of the system are expressed by *f* in equation (2).
The MINLP is also made up of the following constraints which are related with the reactor pressure, temperature and volume, and with the separator and the stripper volumes.

$$\begin{aligned}
& P_{reactor} \leq 3000kPa \\
& 2m^3 \leq Vol_{reactor} \leq 24m^3 \\
& T_{reactor} \leq 175^\circ C \\
& 1m^3 \leq Vol_{separator} \leq 12m^3 \\
& 1m^3 \leq Vol_{stripper} \leq 6m^3
\end{aligned}$$

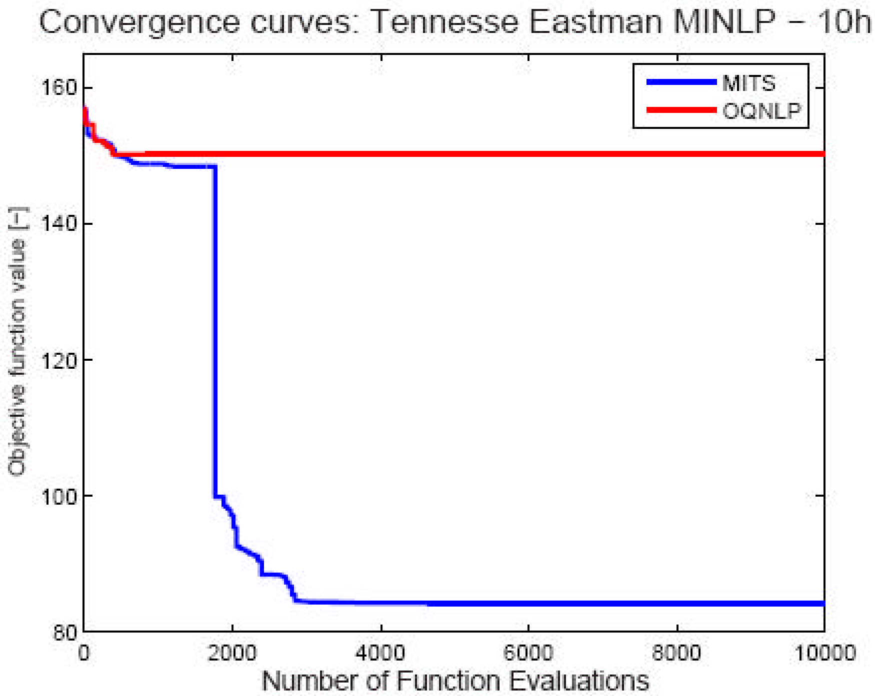

Figure 2 – The Tennessee Eastman Process – Convergence curve

The objective function proposed by Downs and Vogel (1993) in the TEP definitions is based on the operating costs and can be defined as follows:

$$\textit{Total operating costs at base case} = (\textit{Purge costs})(\textit{Purge rate}) + (\textit{Product Stream Costs})(\textit{Product rate}) + (\textit{Compressor costs})(\textit{Compressor work}) + (\textit{Steam costs})(\textit{Steam rate})$$

$$\textit{Total operating costs at base case} = 7.5973\frac{\$}{kgmol}(\textit{Purge rate}) + 0.1434\frac{\$}{kmol}(\textit{Product rate}) + 0.0536\frac{\$}{kWh}(\textit{Compressor work}) + 0.0318\frac{\$}{kg}(\textit{Steam rate})$$

Operating costs for this process are primarily determined by the loss of raw materials (in the purge, in the product stream and by means of the two side reactions). Economic costs for the process are determined by summing the costs of the raw materials and the products leaving in the purge stream and in the product stream, and by using an assigned cost to the amount of *F* formed. The costs concerning the compressor work and the steam to the stripper are also included. Note that the objective function used in the MINLP formulation will be the mean of these operating costs along the whole simulation time horizon. For this work, this simulation time horizon was set to $t = 10\ h$. This is enough time for stabilization of the TEP.

In order to solve the problem we used the solvers MITS and OQNLP. OQNLP, based on the Scatter Search metaheuristic, has been recently reported as one of the best global optimization solvers for black box problems (Neumaier et al., 2005). The dynamic model for the proposed thermodynamic-based control design has been also implemented as a SIMULINK code by the Process Engineering Group at IIM-CSIC. Both solvers were started from the same initial point – with an objective function value of 156.84 $. As stopping criterion, we set the maximum of function evaluations equal to 10000.

The solution obtained by MITS for the binary vector is $b = (0,1,0,1)^T$ and the new objective function value is 84.29 $. This vector defines the new realization of the control loops for the pressure and the level in the reactor by acting over the purge and the *D* Feed, respectively. Figure 2 shows the convergence curves for both solvers. MITS clearly outperformed OQNLP, both regarding the final objective function value and the convergence speed.

5. Conclusion

We have developed a hybrid algorithm for the optimization of the integrated process and control system design problem. We have considered a mixed integer nonlinear program (MINLP) formulation for this class of problems. Our novel hybrid strategy, MITS, uses a combinatorial component, based on Tabu Search, to guide the search into promising areas, and a local solver, MISQP, which is activated to precisely approximate local minima. A Matlab implementation of this technique was developed, and its performance and robustness was tested on the challenging benchmark problems: the Tennessee Eastman Process. MITS presented very good performance, and it was able to obtain remarkable results, clearly outperforming a selected modern MINLP solver.

Acknowledgement

We acknowledge the support of the EU and the Marie-Curie Actions program. We thank the project PRISM – project number MRTN-CT-2004-512233 – 'Towards Knowledge-based Processing Systems' for the support.

References

Antelo, L.T., Banga, J.R. and Alonso, A.A., (2006) *Hierarchical Design of Decentralized Control Structures for the Tennessee Eastman Process, Submitted to Industrial and Chemistry Engineering Research.*

Antelo, L.T., Otero-Muras, I., Banga, J.R. and Alonso, A.A., (2005) *A systematic approach to plant-wide control based on thermodynamics. Submitted to Computers & and Chemical Engineering*. Presented in the 15th European Symposium on Computer Aided Chemical Engineering (Proceedings of ESCAPE-15, 20(B), 1105-1110.

Banga, J.R., Moles, C.G. and Alonso, A.A., (2003) Global optimization of bioprocesses using stochastic and hybrid methods. In "Frontiers In Global Optimization ", C.A. Floudas and P. M. Pardalos, (Eds.), Nonconvex Optimization and Its Applications, vol. 74, pp. 45-70, Kluwer Academic Publishers, ISBN 1-4020-7699-1.

Bansal, V., Perkins, J.D., and Pistikopoulos, E. N., (2000). *Simultaneous design and control optimisation under uncertainty.* Computers and Chemical Engineering 4(2/7), 261.

Battiti, R. and Tecchiolli, G., (1996) *The continuous tabu search: Blending combinatorial optimization and stochastic search for global optimization*, Annals of Operations Research, 63, 153-188.

Downs, J. J. and Vogel, E. F., (1993) *A plant-wide industrial process control problem*, Computers & Chemical Engineering, 17, 245-255.

Esposito W. R. and Floudas C.A. (2000). *Deterministic Global Optimization in Nonlinear Optimal Control Problems.* J. Global Optim. 17, 97-126.

Exler, O. and Schittkowski, K., (2006) *A trust region SQP algorithm for mixed-integer nonlinear programming*, Optimization Letters, DOI 10.1007/s11590-006-0026-1.

Glover, F. and Laguna, M., (1997) *Tabu search*, Boston, Kluwer Academic Publishers.

Morari, M. and Perkins, J.D., (1994) *Design for Operations*, In Foundations of Computer-Aided Process Design (FOCAPD).

Neumaier, A., Shcherbina, O., Huyer, W. and Vinko, T., (2005) *A comparison of complete global optimization solvers.* Math. Program., Ser. B 103, 335-356.

Pistikopoulos, E.N., Ross R. et al., (1999) Towards *the Integration of Process Design, Process Control and Process Operability. Current Status and Future Trends*, Foundations of Computer-Aided Process Design (FOCAPD). Snowmass, Colorado.

Sakizlis, V., Perkins, J.D. and Pistikopoulos, E.N., (2004) *Recent advances in optimization-based simultaneous process and control design.* Computers and Chemical Engineering 28, 2069-2086.

Schweiger, C.A. and Floudas, C.A., (1997) *Interaction of Design and Control: Optimization with Dynamic Models.* In: Optimal Control: Theory, Algorithms, and Applications. Hager W.W. and Pardalos P.M. (Eds.), Kluwer Academic Publishers, 1997.

Schick, K. and Schittkowski, K., (2004*) MIQL: A Fortran Code for Convex Mixed-Integer Quadratic Prbogramming – User´s Guide.* Report, Department of Computer Science, University of Bayreuth.

Yuan, Y.-X., (1995) *On the convergence of a new trust region algorithm,* Numerische Mathematik 70, 515-539.