OPTIMAL CONTROL APPLICATIONS AND METHODS
Optim. Control Appl. Meth., 2003; **24**:103–120 (DOI: 10.1002/oca.724)

Dynamic 'back-off' analysis: use of piecewise linear approximations

José L. Figueroa*,†,‡ and Alfredo C. Desages✠

Departamento de Ingeniería Eléctrica, Universidad Nacional del Sur, Av. Alem 1253 - (8000) Bahía Blanca, Argentina

SUMMARY

The operating point of a process is usually computed by optimizing an objective function, e.g. the profit, subject to some plant characteristics. Typically, the resulting point lies on the boundary of the operating region. At this point, the presence of disturbances can easily cause constraint violations and make the process move to the unfeasible region. Then, it is necessary to move the operating point away into the feasible region by considering the effect that the expected disturbances will have on the operation of the plant. The purpose of this paper is to present an efficient algorithm to modify the operating point in order to keep feasibility (both in steady-state and along transitory) in the process operation against the disturbances. Copyright © 2003 John Wiley & Sons, Ltd.

KEY WORDS: optimizing control; canonical piecewise linear approximations; dynamic optimization; back-off; coal-fired steam generator control

1. INTRODUCTION

The operating point of a chemical process is usually designed to maximize (or minimize) an objective function, e.g. the profit, subject to constraints like the ones inferred from the characteristics of the plant, operating conditions, product specifications and others. These constraints define a feasibility set for the possible operating points, and in most cases, the optimal operating point lies in the boundary of the set. In a second stage, a controller is designed to regulate the behaviour of the plant around the designed steady-state value. The underlying idea is that the controller provides '*perfect control*', so that the plant remains at, or at least close to, its nominal operating point against disturbances, parameter variations and uncertainties on the plant characteristics.

The effect of the disturbances at such regulation level will perturb the operating point from the previously designed one. Thus, this point will be surrounded by a region within which the

*Correspondence to: Dr. J.L. Figueroa, Departamento de Ingeniería Eléctrica, Universidad Nacional del Sur, Av. Alem 1253, 8000 Bahía Blanca, Argentina
† E-mail: figueroa@uns.edu.ar
‡ Also in Planta Piloto de Ing. Química, UNS-CONICET, Camino la Carrindanga Km. 7, 8000-Bahía Blanca.
✠ Deceased

Received 16 August 2000
Revised 4 November 2002
Accepted 29 January 2003

plant will actually operate. Under these perturbed conditions, the plant operation may become unfeasible (in steady-state and/or along transitory). A possible solution is to take a safety margin by strengthening the constraints, i.e. by reducing the feasibility region, thereby designing the operating point away from the actual plant constraints in order to compensate for the unknown process characteristics. In the absence of information as to how disturbances or plant uncertainties affect the steady-state point, such overdesign is hard to justify on economical grounds.

The problem of ensuring feasible regions of operation in spite of plant uncertainty or parameter variations had been addressed for Grossman *et al.* [1]. Their main objective was to come up with 'feasible' chemical processes, i.e. plants that can operate in spite of significant uncertainty in the values of some process parameters, under the assumption of perfect control described above. This concept was extended to dynamic systems for Dimitriadis and Pistikopoulos [2]. The driving idea was to provide the plant designer with tools to evaluate a given design in terms of its flexibility and to provide a quantitative mean of studying design trade-off, in the from of an optimization problem.

In this paper we will follow a different although related approach, suggested for different authors (Bandoni *et al.* [3]; Figueroa *et al.* [4]; Perkins and Walsh [6]) to include operating conditions (such as considerations of disturbances and model uncertainties) at the design stage of the operating point.

The main idea of this strategy is to move the operating point away from the boundary of the feasibility region by considering the effect that the expected disturbances could have on the operation of the plant. This is called *back-off*, and it was originally calculated from the desire for evaluating and comparing control strategies on the economical basis.

Bandoni *et al.* [3] proposed a method to compute the steady-state back-off by converting the problem to a semi-infinite programming. The disturbances are assumed to lie in a given bounded set.

In another context, Narraway and Perkins [6] propose a dynamic optimization problem to the selection of process control structure based on economics, but this approach was limited to a single disturbance. Simultaneously, Figueroa *et al.* [4] extended Bandoni's approach to the dynamics problem, but the results requires considerably large computational resources.

Lately, Figueroa and Desages [7] addressed the steady-state problem, using a canonical piecewise linear approximation of the model, and developing an efficient algorithm for steady-state back-off computation. In this paper, we study the dynamic back-off calculation for non-linear systems using a similar approach.

2. PROBLEM FORMULATION

Consider the following system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{w}) \tag{1}$$

$$\mathbf{y} = \mathbf{h}(\mathbf{x}, \mathbf{u}, \mathbf{w}) \tag{2}$$

where the functions $\mathbf{f}$ and $\mathbf{h}$ are continuously differentiable with respect to their arguments, $\mathbf{x} \in \Re^{n_x}$ is the system state vector, $\mathbf{y} \in \Re^{n_y}$ is the system output vector, $\mathbf{u} \in \Re^{n_l}$ is the vector of the optimization variables (for example the reference values for controllers or other free variables) that are considered as constant and $\mathbf{w} \in \Re^{n_m}$ is a vector of exogenous disturbances. In this formulation we consider that the controller model is included in the model (i.e. in the functions **f** and **h**). The exogenous inputs are assumed to be in the set of step functions (see Figure 1),

$$W = \left\{ w_i(t);\ i = 1, .., m;\ w_i(t) = \left\{ \begin{array}{llll} \tilde{w}_i & \text{if} & t < 0 & \text{with} \quad \underline{w}_i \leqslant \tilde{w}_i \leqslant \bar{w}_i \\ w_i & \text{if} & t \geqslant 0 & \text{with} \quad \underline{w}_i \leqslant w_i \leqslant \bar{w}_i \end{array} \right. \right\}$$

where $\tilde{w}_i$ is the nominal value for the disturbance and $\underline{w}_i$ and $\bar{w}_i$ are the lower and the upper bounds over the disturbances.

In order to complete the description of our system, consider now a set of constraints, that should be satisfied at any time:

$$\mathbf{z}_c = \mathbf{p}(\mathbf{x}, \mathbf{u}, \mathbf{w}) \leqslant 0 \tag{3}$$

where the function **p** is continuously differentiable with respect to its arguments and $\mathbf{z}_c \in \Re^{n_c}$.

This set of inequalities normally follows from the physical analysis of the process (for example, in a chemical process some temperatures, pressures or flowrates must not be exceeded), but they could also be of a more general kind (e.g. product quality control, safety and environmental regulations, etc.).

Let us suppose the uniqueness of the steady state in the region of operation of the process, i.e. given vectors **u** and **w**, there is only one vector **x** satisfying the steady state condition. This means that the equation

$$0 = \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{w}) \tag{4}$$

has a unique solution in **x**. This is a classical assumption in optimization and it is included to ensure that in the region of interest it is possible obtain a numerical solution for the steady state model.

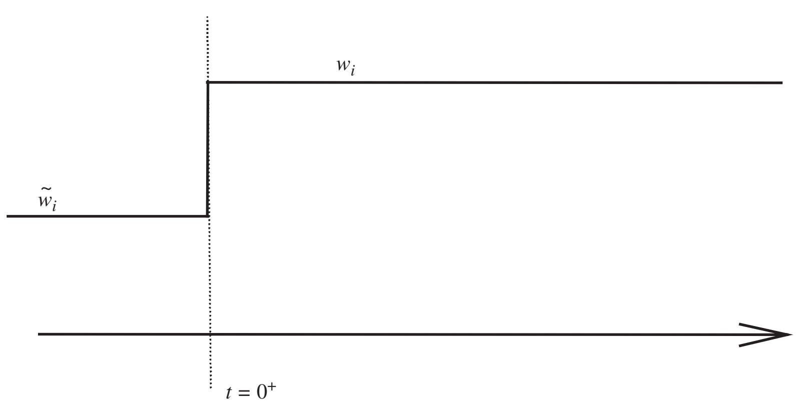

Figure 1. Disturbance $w(t)$ applied to the system.

Given this system, the operating point it is usually computed by solving the following problem.

Problem 1 (Steady-state optimization)

Given a constant fixed vector of disturbance inputs ($\tilde{\mathbf{w}}$) compute the set of free variables ($\mathbf{u}$) so that the following function will be optimized:

$$\begin{aligned} &\min_{\mathrm{u}} \quad z_o(\mathbf{u}, \mathbf{x}) \\ &\text{s.t.} \\ &\qquad \mathbf{f}(\mathbf{u}, \mathbf{x}, \tilde{\mathbf{w}}) = 0 \\ &\qquad \mathbf{z}_{\mathrm{c}} = \mathbf{p}(\mathbf{u}, \mathbf{x}, \tilde{\mathbf{w}}) \leqslant 0 \end{aligned} \tag{5}$$

where $z_0(\mathbf{x}, \mathbf{u})$ is an objective function with some economical meaning and the disturbances are considered at their nominal value (i.e. $\mathbf{w} = \tilde{\mathbf{w}}$).

When optimization (5) is solved, the operating condition is fixed by using the solution vector $\mathbf{u}$. We will consider now the case in which the vector of exogenous inputs ($\mathbf{w}$) is in the set W. The effect of the disturbances at this point is to move the plant away from this desired operating point. Under this perturbed condition, the process operation may become unfeasible at any time during the transient. This effect may require a displacement in the operating point away from the one determined in the optimization (5) to maintain feasible operation. To compute the magnitude of this movement, we will formulate the following optimization,

Problem 2 (Dynamic back-off)

We shall compute the set of control inputs ($\mathbf{u}$) so that the following function will be optimized:

$$\begin{aligned} &\min_{\mathrm{u}} \quad z_{\mathrm{o}}(\mathbf{x}(0), \mathbf{u}) \\ &\text{s.t.} \\ &\qquad \left.\begin{aligned} \dot{\mathbf{x}} - \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{w}(t)) &= 0 \\ \mathbf{z}_{\mathrm{c}} = \mathbf{p}(\mathbf{x}, \mathbf{u}, \mathbf{w}(t)) &\leqslant 0 \end{aligned}\right\} \forall \mathbf{w}(t) \in W \end{aligned} \tag{6}$$

where the expression $\mathbf{x}(0)$ means the value of $\mathbf{x}(\mathrm{t})$ at the time $t = 0$ and in conditions of nominal disturbances ($\mathbf{w} = \tilde{\mathbf{w}}$). Note that this optimization involves an infinite number of constraints (one for each disturbance in the set W). An algorithm for the solution of this optimization problem has been presented by Figueroa *et al.* [4]. This problem may not have a solution. In practice, this means that it is not possible to operate the plant under some requirements of performance for the disturbance magnitude defined by $\mathbf{w}(t) \in W$.

Note the difference between problems 1 and 2. In the former, the constraints need to be verified for a specific $\mathbf{w}(t) = \tilde{\mathbf{w}}$, in the latter, the constraints need to be satisfied for all $\mathbf{w}(t)$ in the set W. It is important to point out that the dynamic system of equation (1) could include the controller dynamics for almost all control structures.

3. PROBLEM FORMULATION AS CPWL

Our strategy to find an appropriate solution for these problems, due to the computational complexity present in the solution of the non-linear case, consists in finding a Canonical Piecewise Linear Approximation (CPWL) of the system and constraints under consideration. A description of these functions can be found in Appendix A.

3.1. Problem representation as CPWL

Let the sets $X \subset \Re^{n_x}$, $U \subset \Re^{l}$, $W \subset \Re^{m}$ be the domains of the $\mathbf{x}$, $\mathbf{u}$ and $\mathbf{w}$ variables respectively, and consider the set

$$\aleph = \left\{ \left[\mathbf{x}^{\mathrm{T}}, \mathbf{u}^{\mathrm{T}}, \mathbf{z}^{\mathrm{T}}\right]^{\mathrm{T}} : \mathbf{x} \in X, \mathbf{u} \in U, \mathbf{w} \in W \right\}$$

on which we want to approximate the non-linear system. Consider also the following partition in the set $\aleph$ such that

$$\aleph = \bigcup_{j=1}^{\sigma} \aleph^{j}$$

where $\aleph^j$ is called the '*j*th *partition*' or '*j*th *region*' of the set $\aleph$.

In each of these regions, (for example, if the system is constrained to the kth region (i.e. $(\mathbf{x}, \mathbf{u}, \mathbf{w}) \in \aleph^k$)) the non-linear representation of Equations (1) and (3), is approximated using CPWL functions in the form

$$\dot{\mathbf{x}} = \boldsymbol{\xi}_{\mathrm{xx}}^{k} \mathbf{x}, \boldsymbol{\xi}_{\mathrm{xu}}^{k} \mathbf{u} + \boldsymbol{\xi}_{\mathrm{xw}}^{k} \mathbf{w} + \boldsymbol{\eta}_{\mathrm{x}}^{k} \tag{7}$$

$$\mathbf{z}_{\mathrm{c}} = \boldsymbol{\xi}_{\mathrm{cx}}^{k} \mathbf{x}, \boldsymbol{\xi}_{\mathrm{cu}}^{k} \mathbf{u} + \boldsymbol{\xi}_{\mathrm{cw}}^{k} \mathbf{w} + \boldsymbol{\eta}_{\mathrm{c}}^{k} \tag{8}$$

and the objective function as

$$\mathbf{z}_{\mathrm{o}} = \boldsymbol{\xi}_{\mathrm{ox}}^{k} \mathbf{x}, \boldsymbol{\xi}_{\mathrm{ou}}^{k} \mathbf{u} + \eta_{\mathrm{o}}^{k} \tag{9}$$

That can be written in a unique expression for the complete domain as

$$\dot{\mathbf{x}} = \mathbf{a}_{\mathrm{x}} + \mathbf{B}_{\mathrm{xx}} \mathbf{x} + \mathbf{B}_{\mathrm{xu}} \mathbf{u} + \mathbf{B}_{\mathrm{xw}} \mathbf{w} + \sum_{i=1}^{\sigma} \mathbf{c}_{\mathrm{x}i} |\boldsymbol{\alpha}_{\mathrm{x}i} \mathbf{x} + \boldsymbol{\alpha}_{\mathrm{u}i} \mathbf{u} + \boldsymbol{\alpha}_{\mathrm{w}i} \mathbf{w} - \beta_i| \tag{10}$$

$$\mathbf{z}_{\mathrm{c}} = \mathbf{a}_{\mathrm{c}} + \mathbf{B}_{\mathrm{cx}} \mathbf{x} + \mathbf{B}_{\mathrm{cu}} \mathbf{u} + \mathbf{B}_{\mathrm{cw}} \mathbf{w} + \sum_{i=1}^{\sigma} \mathbf{c}_{\mathrm{c}i} |\boldsymbol{\alpha}_{\mathrm{x}i} \mathbf{x} + \boldsymbol{\alpha}_{\mathrm{u}i} \mathbf{u} + \boldsymbol{\alpha}_{\mathrm{w}i} \mathbf{w} - \beta_i| \tag{11}$$

$$\mathbf{z}_{\mathrm{o}} = \mathbf{a}_{\mathrm{o}} + \mathbf{B}_{\mathrm{ox}} \mathbf{x} + \mathbf{B}_{\mathrm{ou}} \mathbf{u} + \sum_{i=1}^{\sigma} \mathbf{c}_{\mathrm{o}i} |\boldsymbol{\alpha}_{\mathrm{x}i} \mathbf{x} + \boldsymbol{\alpha}_{\mathrm{u}i} \mathbf{u} + \boldsymbol{\alpha}_{\mathrm{w}i} \mathbf{w} - \beta_i| \tag{12}$$

where all the matrices and vectors have appropriate dimensions with elements in the real field. We should make a clear distinction between the original description of the real system

(Equations (1) and (3)) in a generic non-linear representation and the CPWL representation (Equations (10)–(11)), that is an approximation to the real system.

It is also important to note that the approximation of the constraint functions has been made within the feasible region, i.e. the CPWL approximation ($\mathbf{z}_{c_{pw}}$ of Equation (11)) of the function ($\mathbf{Z}_c$ of Equation (3)) satisfies $\mathbf{z}_{c_{pw}} - \mathbf{z}_c \leqslant \mathbf{0}$ for all possible set of variables.

The coefficients of Equations (7)–(9) are related with the ones of Equations (10)–(12) are related to the following inequalities:

$$\xi^k_{xx} = \mathbf{B}_{xx} + \sum_{i=1}^{\sigma} \mathbf{c}_{xi} \cdot \boldsymbol{\alpha}_{xi} \cdot \gamma^k_i, \quad \xi^k_{xu} = \mathbf{B}_{xu} + \sum_{i=1}^{\sigma} \mathbf{c}_{xi} \cdot \boldsymbol{\alpha}_{ui} \cdot \gamma^k_i$$

$$\xi^k_{xw} = \mathbf{B}_{xw} + \sum_{i=1}^{\sigma} \mathbf{c}_{xi} \cdot \boldsymbol{\alpha}_{wi} \cdot \gamma^k_i, \quad \boldsymbol{\eta}^k_x = \mathbf{a}_x + \sum_{i=1}^{\sigma} \mathbf{c}_{xi}.\boldsymbol{\beta}_i.\gamma^k_i, \quad \xi^k_{cx} = \mathbf{B}_{cx} + \sum_{i=1}^{\sigma} \mathbf{c}_{ci} \cdot \boldsymbol{\alpha}_{xi} \cdot \gamma^k_i$$

$$\xi^k_{cu} = \mathbf{B}_{cu} + \sum_{i=1}^{\sigma} \mathbf{c}_{ci} \cdot \boldsymbol{\alpha}_{ui} \cdot \gamma^k_i, \quad \xi^k_{cw} = \mathbf{B}_{cw} + \sum_{i=1}^{\sigma} \mathbf{c}_{ci} \cdot \boldsymbol{\alpha}_{wi} \cdot \gamma^k_i, \quad \boldsymbol{\eta}^k_c = \mathbf{a}_c + \sum_{i=1}^{\sigma} \mathbf{c}_{ci}.\boldsymbol{\beta}_i.\gamma^k_i$$

$$\xi^k_{ox} = \mathbf{B}_{ox} + \sum_{i=1}^{\sigma} \mathbf{c}_{oi}.\boldsymbol{\alpha}_{xi} \cdot \gamma^k_i, \quad \xi^k_{ou} = \mathbf{B}_{ou} + \sum_{i=1}^{\sigma} \mathbf{c}_{oi} \cdot \boldsymbol{\alpha}_{ui} \cdot \gamma^k_i, \quad \boldsymbol{\eta}^k_o = \mathbf{a}_o + \sum_{i=1}^{\sigma} \mathbf{c}_{oi} \cdot \beta_i \cdot \gamma^k_i$$

with $\gamma^k_i = \text{sign}(\boldsymbol{\alpha}_{xi}\mathbf{x} + \boldsymbol{\alpha}_{ui}\mathbf{u} + \boldsymbol{\alpha}_{wi}\mathbf{w} - \beta_i)$. From comparison of (9) and (12) it is clear that $\sum_{i=1}^{\sigma} \mathbf{c}_{oi}.\boldsymbol{\alpha}_{wi}.\gamma^k_i = 0$ for the complete domain $\aleph$.

Note that the *sign* function in the last expression determines the *Sector belonging condition*, i.e. the sign vector $\boldsymbol{\gamma}^k$ defined as $\boldsymbol{\gamma}^k = [\gamma^k_1, \gamma^k_2, \ldots, \gamma^k_\sigma]$ is uniquely related to the kth partition (Figueroa and Desages [7]). Consequently, a point $(\mathbf{x}^j, \mathbf{u}^j, \mathbf{w}^j)$ will lie in $\aleph^k$ if and only if it satisfies the inequality

$$\mathbf{z}^k_\aleph = \xi^k_{\aleph x}.\mathbf{x}^j + \xi^k_{\aleph u}.\mathbf{u}^j + \xi^k_{\aleph w}.\mathbf{w}^j + \boldsymbol{\eta}^k_\aleph \leqslant \mathbf{0} \tag{13}$$

where $[\xi^k_{\aleph x}]_i = -\gamma^k_i \boldsymbol{\alpha}_{xi}$, $[\xi^k_{\aleph u}]_i = -\gamma^k_i \boldsymbol{\alpha}_{ui}$, $[\xi^k_{\aleph w}]_i = -\gamma^k_i \boldsymbol{\alpha}_{wi}$, $[\boldsymbol{\eta}^k_\aleph]_i = \gamma^k_i \beta_i$, and $[.]_i$ means the ith row in the matrix [.].

3.2. Steady-state solution for a CPWL model

Let us consider an algorithm to compute the steady-state solution for the system. Let us assume that the inverse of each matrix ξ^k_{xx} exists. This implies that (in each sector) the equation

$$0 = \xi^k_{xx}\mathbf{x} + \xi^k_{xu}\mathbf{u} + \xi^k_{xw}\mathbf{w} + \boldsymbol{\eta}^k_x$$

has a unique solution in $\mathbf{x}$ (Note that the same condition was globally imposed for the non-linear system). Then, it is possible to use the following algorithm to compute the steady-state point (Figueroa and Desages [7]):

Algorithm 1: Steady-State Computation

Data: A set of external variables $(\tilde{\mathbf{u}}, \tilde{\mathbf{w}})$ and an initial guess $\mathbf{x}^0$. Set $k = 0$.

Step 1: Compute the positive λ_i^k that makes the *i*th entry of the following vector zero:

$$\mathbf{z}_{\aleph}^k = \boldsymbol{\xi}_{\aleph\mathrm{x}}^k(\mathbf{x}^k + \lambda\,(\boldsymbol{\xi}_{\mathrm{xx}}^k)^{-1}\mathbf{z}_{\mathrm{x}}^k) + \boldsymbol{\xi}_{\aleph\mathrm{u}}^k\tilde{\mathbf{u}} + \boldsymbol{\xi}_{\aleph\mathrm{w}}^k\tilde{\mathbf{w}} + \boldsymbol{\eta}_{\aleph}^k$$

for $i = 1, \ldots, \sigma$, where $\mathbf{z}_{\mathrm{x}}^k = \boldsymbol{\xi}_{\mathrm{xx}}^k\mathbf{x}^k + \boldsymbol{\xi}_{\mathrm{xu}}^k\tilde{\mathbf{u}} + \boldsymbol{\xi}_{\mathrm{xw}}^k\tilde{\mathbf{w}} + \boldsymbol{\eta}_{\mathrm{x}}^k$.

Step 2: Compute $\lambda_{\mathrm{c}} = \min_i\ \lambda_i^k$ and $\mathbf{x}^{k+1} = \mathbf{x}^k - \lambda_{\mathrm{c}}\,(\boldsymbol{\xi}_{\mathrm{xx}}^k)^{-1}\mathbf{z}_{\mathrm{x}}^k$ with $\mathbf{z}_{\mathrm{x}}^k = \boldsymbol{\xi}_{\mathrm{xx}}^k\mathbf{x}^k + \boldsymbol{\xi}_{\mathrm{xu}}^k\tilde{\mathbf{u}} + \boldsymbol{\xi}_{\mathrm{xw}}^k\tilde{\mathbf{w}} + \boldsymbol{\eta}_{\mathrm{x}}^k$.

Step 3: If λ_{c} is smaller than one, set $k = k + 1$ and *return to Step 1*. Otherwise, $\mathbf{x}^{k+1}$ is the steady-state value. *Stop*.

This is a version of the Katzenelson algorithm modified to our particular problem. Some considerations about the existence and uniqueness of solutions for this algorithm could be found in Figueroa and Desages [7]. Efficient versions of this algorithm can be found in the literature [8, 9].

3.3. *Dynamic simulation for a CPWL model*

In this section we will analyse a way to obtain an expression for $\mathbf{x}(t)$ when a disturbance is applied. Consider the system at the steady state point $(\mathbf{x}^0, \mathbf{u}, \tilde{\mathbf{w}}) \in \aleph^0$, (i.e. it satisfies $\mathbf{x}^0 = -(\boldsymbol{\xi}_{\mathrm{xx}}^0)^{-1}.(\boldsymbol{\xi}_{\mathrm{xu}}^0.\mathbf{u} + \boldsymbol{\xi}_{\mathrm{xw}}^0.\tilde{\mathbf{w}} + \boldsymbol{\eta}_{\mathrm{x}}^0)$. Then, a disturbance $\mathbf{w}(t) \in W$ is applied to the system. While the system is in sector $\aleph^0$, it is easy to see that the state vector will be

$$\mathbf{x}(t) = \mathrm{e}^{\xi_{\mathrm{xx}}^0 t}\mathbf{x}^0 - \left(\mathbf{I} - \mathrm{e}^{\xi_{\mathrm{xx}}^0 t}\right)(\boldsymbol{\xi}_{\mathrm{xx}}^0)^{-1}.(\boldsymbol{\xi}_{\mathrm{xu}}^0.\mathbf{u} + \boldsymbol{\xi}_{\mathrm{xw}}^0.\mathbf{w} + \boldsymbol{\eta}_{\mathrm{x}}^0) \tag{14}$$

This expression will be valid till the moment in which the system reaches the next sector (called $\aleph^1$). Suppose that this occurs at time $\mathbf{t}^0$, when the value of the state is

$$\begin{aligned}\mathbf{x}^1 &= \mathbf{x}(t^0) = \mathrm{e}^{\xi_{\mathrm{xx}}^0 t^0}\mathbf{x}^0 - \left(\mathbf{I} - \mathrm{e}^{\xi_{\mathrm{xx}}^0 t^0}\right)(\boldsymbol{\xi}_{\mathrm{xx}}^0)^{-1}.(\boldsymbol{\xi}_{\mathrm{xu}}^0.\mathbf{u} + \boldsymbol{\xi}_{\mathrm{xw}}^0.\mathbf{w} + \eta_{\mathrm{x}}^0)\\
&= -\mathrm{e}^{\xi_{\mathrm{xx}}^0 t^0}.(\boldsymbol{\xi}_{\mathrm{xx}}^0)^{-1}.(\boldsymbol{\xi}_{\mathrm{xu}}^0.\mathbf{u} + \boldsymbol{\xi}_{\mathrm{xw}}^0.\tilde{\mathbf{w}} + \eta_{\mathrm{x}}^0) - \left(\mathbf{I} - \mathrm{e}^{\xi_{\mathrm{xx}}^0 t^0}\right).(\boldsymbol{\xi}_{\mathrm{xx}}^0)^{-1}.(\boldsymbol{\xi}_{\mathrm{xu}}^0.\mathbf{u} + \boldsymbol{\xi}_{\mathrm{xw}}^0.\mathbf{w} + \eta_{\mathrm{x}}^0)\\
&= -(\boldsymbol{\xi}_{\mathrm{xx}}^0)^{-1}\boldsymbol{\xi}_{\mathrm{xu}}^0.\mathbf{u} + \left(\mathbf{I} - \mathrm{e}^{\xi_{\mathrm{xx}}^0 t^0}\right)(\boldsymbol{\xi}_{\mathrm{xx}}^0)^{-1}\boldsymbol{\xi}_{\mathrm{xw}}^0\mathbf{w} - \mathrm{e}^{\xi_{\mathrm{xx}}^0 t^0}(\boldsymbol{\xi}_{\mathrm{xx}}^0)^{-1}\boldsymbol{\xi}_{\mathrm{xw}}^0\tilde{\mathbf{w}} - (\boldsymbol{\xi}_{\mathrm{xx}}^0)^{-1}\eta_{\mathrm{x}}^0\end{aligned}$$

and using this state as an initial condition for sector $\aleph^1$ it is possible to compute

$$\mathbf{x}(t) = \mathrm{e}^{\xi_{\mathrm{xx}}^1 t}\mathbf{x}^1 - \left(\mathbf{I} - \mathrm{e}^{\xi_{\mathrm{xx}}^1 t}\right)(\boldsymbol{\xi}_{\mathrm{xx}}^1)^{-1}.(\boldsymbol{\xi}_{\mathrm{xu}}^1.\mathbf{u} + \boldsymbol{\xi}_{\mathrm{xw}}^1.\mathbf{w} + \boldsymbol{\eta}_{\mathrm{x}}^1)$$

expression that will be valid till the time in which the system reaches sector $\aleph^2$. Then, it is possible to obtain an algorithm to perform the dynamic simulation.

Algorithm 2: Dynamic Simulation

Data: A set of external variables $(\tilde{\mathbf{u}}, \tilde{\mathbf{w}})$, the magnitude of the step in the disturbance $\mathbf{w}$ and the horizon $(T^{\max})$ to perform the simulation.

Step 0: Compute the steady-state solution for the model (i.e. $\mathbf{x}^0$) using Algorithm 1. Set $k = 0$.

Step 1: Determine in which sector, $\aleph^k$, the point $(\mathbf{x}^k, \mathbf{u}, \mathbf{w})$ lies; and compute the linear model valid in this sector.

Step 2: Using this linear model, perform the time simulation as

$$\mathbf{x}(t) = \mathrm{e}^{\xi_{\mathrm{xx}}^k t}\mathbf{x}^k - \left(\mathbf{I} - \mathrm{e}^{\xi_{\mathrm{xx}}^k t}\right)\left(\xi_{\mathrm{xx}}^k\right)^{-1}.\left(\xi_{\mathrm{xu}}^k.\mathbf{u} + \xi_{\mathrm{xw}}^k.\mathbf{w} + \boldsymbol{\eta}_{\mathrm{x}}^k\right)$$

until the time t^{k+1} for which the first entry of the vector

$$\mathbf{z}_{\aleph}^k = \xi_{\aleph\mathrm{x}}^k\mathbf{x}(t) + \xi_{\aleph\mathrm{u}}^k\mathbf{u} + \xi_{\aleph\mathrm{w}}^k\mathbf{w} + \boldsymbol{\eta}_{\aleph}^k$$

is zero (this means that the linear model does not longer represent the real process). The state at this time is

$$\mathbf{x}(t^{k+1}) = \mathrm{e}^{\xi_{\mathrm{xx}}^k t^{k+1}}\mathbf{x}^k - \left(\mathbf{I} - \mathrm{e}^{\xi_{\mathrm{xx}}^k t^{k+1}}\right)\left(\xi_{\mathrm{xx}}^k\right)^{-1}.\left(\xi_{\mathrm{xu}}^k.\mathbf{u} + \xi_{\mathrm{xw}}^k.\mathbf{w} + \boldsymbol{\eta}_{\mathrm{x}}^k\right)$$

Step 3: If t^{K+1} is smaller that $T^{\max}$, set $\mathbf{x}^{k+1} = \mathbf{x}\left(t^{k+1}\right)$, make $k = k+1$ and return to Step 1 to proceed similarly in the next sector. Otherwise, *Stop*.

Using the results of this algorithm, a generic expression for $\mathbf{x}(t)$ when the system goes through sectors $[\aleph^0, \aleph^1, \aleph^2, \ldots, \aleph^h]$ could be written as

$$\mathbf{x}(t) = \boldsymbol{\Phi}_{\mathrm{xu}}(t)\mathbf{u} + \boldsymbol{\Phi}_{\mathrm{xw}}(t)\mathbf{w} + \Phi_{\mathrm{x}}(t, \tilde{\mathbf{w}}) \tag{15}$$

where

$$\boldsymbol{\Phi}_{\mathrm{xu}}(t) = -\,\mathrm{e}^{\xi_{\mathrm{xx}}^h t}\left(\left(\prod_{j=1}^{h-1}\mathrm{e}^{\xi_{\mathrm{xx}}^j t^j}\right)\left(\xi_{\mathrm{xx}}^0\right)^{-1}\xi_{\mathrm{xu}}^0 + \sum_{i=1}^{h-1}\left(\prod_{j=i+1}^{h-1}\mathrm{e}^{\xi_{\mathrm{xx}}^j t^j}\right)\left(\mathbf{I} - \mathrm{e}^{\xi_{\mathrm{xx}}^i t^i}\right)\left(\xi_{\mathrm{xx}}^i\right)^{-1}\xi_{\mathrm{xu}}^i\right) - \cdots$$
$$\times\left(\mathbf{I} - \mathrm{e}^{\xi_{\mathrm{xx}}^h t}\right)\left(\xi_{\mathrm{xx}}^h\right)^{-1}\xi_{\mathrm{xu}}^h$$

$$\boldsymbol{\Phi}_{\mathrm{xw}}(t) = -\mathrm{e}^{\xi_{\mathrm{xx}}^h t}\left(\sum_{i=0}^{h-1}\left(\prod_{j=i+1}^{h-1}e^{\xi_{\mathrm{xx}}^j t^j}\right)\left(\mathbf{I} - e^{\xi_{\mathrm{xx}}^i t^i}\right)\left(\xi_{\mathrm{xx}}^i\right)^{-1}\xi_{\mathrm{xw}}^i\right) - \left(\mathbf{I} - \mathrm{e}^{\xi_{\mathrm{xx}}^h t}\right)\left(\xi_{\mathrm{xx}}^h\right)^{-1}\xi_{\mathrm{xw}}^h$$

$$\boldsymbol{\Phi}_{\mathrm{x}}(t, \tilde{\mathbf{w}}) = -\,\mathrm{e}^{\xi_{\mathrm{xx}}^h t}\left(-\left(\prod_{j=0}^{h-1}\mathrm{e}^{\xi_{\mathrm{xx}}^j t^j}\right)\left(\xi_{\mathrm{xx}}^0\right)^{-1}\xi_{\mathrm{xu}}^0\tilde{\mathbf{w}} + \left(\prod_{j=1}^{h-1}\mathrm{e}^{\xi_{\mathrm{xx}}^j t^j}\right)\left(\xi_{\mathrm{xx}}^0\right)^{-1}\boldsymbol{\eta}_{\mathrm{x}}^0 + \cdots\right.$$
$$\left.\times\sum_{i=1}^{h-1}\left(\prod_{j=i+1}^{h-1}\mathrm{e}^{\xi_{\mathrm{xx}}^j t^j}\right)\left(\mathbf{I} - \mathrm{e}^{\xi_{\mathrm{xx}}^i t^i}\right)\left(\xi_{\mathrm{xx}}^i\right)^{-1}\boldsymbol{\eta}_{\mathrm{x}}^i\right) - \left(\mathbf{I} - \mathrm{e}^{\xi_{\mathrm{xx}}^h t}\right)\left(\xi_{\mathrm{xx}}^h\right)^{-1}\boldsymbol{\eta}_{\mathrm{x}}^h$$

where t^i is the time at which the system leaves sector $\aleph^i$. Note the dependence of the matrices $\boldsymbol{\Phi}_{\mathrm{xu}}(t)$, $\boldsymbol{\Phi}_{\mathrm{xw}}(t)$, and $\boldsymbol{\Phi}_{\mathrm{x}}(t, \tilde{\mathbf{w}})$ on the sectors $[\aleph^0, \aleph^1, \aleph^2, \ldots, \aleph^h]$ and on the times $[t^0, t^1, t^2, \ldots, t^{h-1}]$. This means that, in general, expression (15) is not longer valid if any change occurs in the disturbance $\mathbf{w}$ or a new optimization variable $\mathbf{u}$ is applied. In this case, the simulation should be repeated.

Now, we can compute an expression for the constraints for a generic sector $\aleph^h$,

$$\begin{aligned}
\mathbf{z}_c &= \boldsymbol{\xi}^h_{cx}.\mathbf{x}(t) + \boldsymbol{\xi}^h_{cu}.\mathbf{u} + \boldsymbol{\xi}^h_{cw}.\mathbf{w} + \boldsymbol{\eta}^h_c \\
&= \boldsymbol{\xi}^h_{cx}.\left(\boldsymbol{\Phi}^h_{xu}.\mathbf{u} + \boldsymbol{\Phi}^h_{xw}.\mathbf{w} + \boldsymbol{\Phi}^h_{x}\right) + \boldsymbol{\xi}^h_{cu}.\mathbf{u} + \boldsymbol{\xi}^h_{cw}.\mathbf{w} + \boldsymbol{\eta}^h_c \\
&= \left(\boldsymbol{\xi}^h_{cx}.\boldsymbol{\Phi}^h_{xu} + \boldsymbol{\xi}^h_{cu}\right).\mathbf{u} + \left(\boldsymbol{\xi}^h_{cx}.\boldsymbol{\Phi}^h_{xw} + \boldsymbol{\xi}^h_{cw}\right).\mathbf{w} + \left(\boldsymbol{\xi}^h_{cx}.\boldsymbol{\Phi}^h_{x} + \boldsymbol{\eta}^h_c\right) \\
&= \boldsymbol{\Phi}^h_{cu}(t).\mathbf{u} + \boldsymbol{\Phi}^h_{cw}(t).\mathbf{w} + \boldsymbol{\Phi}^h_{c}(t, \tilde{\mathbf{w}})
\end{aligned} \quad (16)$$

4. SOLUTION TO THE DYNAMIC BACK-OFF PROBLEM

First, let us consider the constraints specified in Problem 2:

$$\left.\begin{aligned}
\dot{\mathbf{x}} - \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{w}) &= 0 \\
\mathbf{z}_c = \mathbf{p}(\mathbf{x}, \mathbf{u}, \mathbf{w}) &\leqslant 0
\end{aligned}\right\} \forall \mathbf{w} \in W$$

Our strategy is to determine the 'worst case perturbation' $\mathbf{w} \in W$ in the sense of producing the largest value of the entries of $\mathbf{z}_c$. To do so, for the jth entry of vector $\mathbf{z}_c$, a function $\lambda_j(\mathbf{u})$ is defined as

$$\begin{aligned}
&\lambda_j(\mathbf{u}) = \max_{\mathrm{w} \in W} \max_{t \in [o, \infty)} [\mathbf{z}_c]_j \\
&\text{s.t.} \\
&\qquad \mathbf{x} - \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{w}) = \mathbf{0}
\end{aligned} \quad (17)$$

or equivalently

$$\lambda_j(\mathbf{u}) = \max_{\mathrm{w} \in W} \max_{t \in [o, \infty)} [\mathbf{z}_c(\mathbf{x}(t), \mathbf{u}, \mathbf{w})]_j \quad (18)$$

where $\mathbf{x}(t)$ is the solution to $\dot{\mathbf{x}} - \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{w}) = \mathbf{0}$, computed by using algorithm 2; and the subscript j means the jth row of the vector (or matrix). If we consider that the maximum of $\max_{t \in [o, \infty)} [\mathbf{z}_c(\mathbf{x}(t), \mathbf{u}, \mathbf{w})]_j$ is in sector $\aleph^h$ at time $t = t^{\max}$, then it is possible to write,

$$\max_{t \in [0, \infty)} [\mathbf{z}_c(\mathbf{x}(t), \mathbf{u}, \mathbf{w})]_j = \left[\boldsymbol{\Phi}^h_{cw}(t^{\max}).\mathbf{w} + \boldsymbol{\Phi}^h_{cu}(t^{\max}).\mathbf{u} + \boldsymbol{\Phi}^h_{c}(t^{\max})\right]_j \quad (19)$$

Now, given a fixed control input vector $\mathbf{u}$, and assuming that the argument $t^{\max}$, the sectors $[\aleph^0, \aleph^1, \aleph^2, \ldots, \aleph^h]$ and the times $[t^0, t^1, t^2, \ldots, t^{h-1}]$ are not depending on $\mathbf{w}$, the solution to the problem (18) is for the disturbance

$$\mathbf{w}^{\max} = \left\{ w_j = \left\{ \begin{array}{lll} \bar{w}_j & \text{if} & \left[\Phi^{\mathrm{h}}_{cw}(t^{\max})\right]_j \geqslant 0 \\ \underline{w}_j & \text{if} & \left[\Phi^{\mathrm{h}}_{cw}(t^{\max})\right]_j < 0 \end{array} \right\} \right. \quad (20)$$

However, this situation is unrealistic because when $\mathbf{w}$ changes, the values of $t^0, t^1, t^2, \ldots, t^{h-1}$, $t^{\max}$, $\aleph^0, \aleph^1, \aleph^2, \ldots, \aleph^{h-1}$ and $\aleph^h$ will also change. Then, in order to compute $\mathbf{w}^{\max}$ we propose the following algorithm:

Algorithm 3: Worst Disturbance Determination

Data: A set of external variables $(\mathbf{u}, \tilde{\mathbf{w}})$, an initial disturbance $\mathbf{w}^0$ and the set W of possible disturbances. Set $k = 1$,

Step 1: Perform the simulation using algorithm 2, determine the time for which the jth entry of vector $\mathbf{z}_c$ (i.e. $[\mathbf{z}_c]_j$) is at maximum and compute the matrices $\mathbf{\Phi}^h_{cw}(t^{\max})$, $\mathbf{\Phi}^h_{cu}(t^{\max})$ and $\mathbf{\Phi}^h_c$ $(t^{\max})$. Compute

$$\lambda_j^{k-1} = \left[\mathbf{\Phi}^h_{cw}(t^{\max}).\mathbf{w}^{k-1} + \mathbf{\Phi}^h_{cu}(t^{\max}).\mathbf{u} + \mathbf{\Phi}^h_c(t^{\max})\right]_j$$

Step 2: Compute the argument, $\mathbf{w}^k = \mathbf{w}^{\max}$, that maximizes the expression

$$\max_{\mathbf{w} \in W} \max_{t \in [0,\infty)} [\mathbf{z}_c(\mathbf{x}(t), \mathbf{u}, \mathbf{w})]_j$$

using equation (20), set $\lambda_j^k = \left[\mathbf{\Phi}^h_{cw}(t^{\max}).\mathbf{w}^k + \mathbf{\Phi}^h_{cu}(t^{\max}).\mathbf{u} + \mathbf{\Phi}^h_c(t^{\max})\right]_j$, and adapt the limits for the disturbance set as

$$\bar{\mathbf{w}} = \left\{ \bar{w}_j = \begin{cases} \bar{w}_j & \text{if} \quad \left[\mathbf{\Phi}^h_{cw}(t^{\max})\right]_j \geqslant 0 \\ w_j^{k-1} & \text{if} \quad \left[\mathbf{\Phi}^h_{cw}(t^{\max})\right]_j < 0 \end{cases} \right\}$$

and

$$\underline{\mathbf{w}} = \left\{ \underline{w}_j = \begin{cases} w_j^{k-1} & \text{if} \quad \left[\mathbf{\Phi}^h_{cw}(t^{\max})\right]_j \geqslant 0 \\ \underline{w}_j & \text{if} \quad \left[\mathbf{\Phi}^h_{cw}(t^{\max})\right]_j < 0 \end{cases} \right\}$$

Step 3: If $\lambda_j^{k-1} \neq \lambda_j^k$, make $k = k + 1$ and return to Step 1. Otherwise, $\lambda_j = \left[\mathbf{\Phi}^h_{cw}(t^{\max}).\mathbf{w}^k + \mathbf{\Phi}^h_{cu}(t^{\max}).\mathbf{u} + \mathbf{\Phi}^h_c(t^{\max})\right]_j$ and $\hat{\mathbf{w}}_j = \mathbf{w}^k$ is the argument for which it happened; Stop the algorithm.

Note that the iterations in this algorithm are necessary due to the non-linear nature of the original problem. This is due to the fact that in the solution proposes on (20); we cannot ensure that the resulting point is in the same sector, or neither the system states are in a path on another sectors than the computed in Step 1. This problem is equivalent to the solution of a global non-linear optimization, where not convergence can be ensured. If the problem were linear, the convergence of this algorithm will be guaranteed in one iteration.

To simplify the notation in the following, let us consider the following equality:

$$\left[\mathbf{\Phi}^h_{cw}(t^{\max}).\hat{\mathbf{w}}_j + \mathbf{\Phi}^h_{cu}(t^{\max}).\mathbf{u} + \mathbf{\Phi}^h_c(t^{\max})\right]_j = [\mathbf{\Phi}_{cu}]_j.\mathbf{u} + [\mathbf{\Phi}_{cw}]_j.\hat{\mathbf{w}}_j + [\mathbf{\Phi}_c]_j$$

And then, it is possible to group these equations for $j = 1, 2, \ldots, n_c$, (where n_c is the number of constraints) as

$$\lambda = \mathbf{D}.\mathbf{u} + \mathbf{E} \tag{21}$$

$$\text{where} \quad \mathbf{D} = \begin{bmatrix} [\mathbf{\Phi}_{cu}]_1 \\ [\mathbf{\Phi}_{cu}]_2 \\ \vdots \\ [\mathbf{\Phi}_{cu}]_{n_c} \end{bmatrix} \quad \text{and} \quad \mathbf{E} = \begin{bmatrix} [\mathbf{\Phi}_{cw}]_1.\hat{\mathbf{w}}^1 + [\mathbf{\Phi}_c]_1 \\ [\mathbf{\Phi}_{cw}]_2.\hat{\mathbf{w}}^2 + [\mathbf{\Phi}_c]_2 \\ \vdots \\ [\mathbf{\Phi}_{cw}]_{n_c}.\hat{\mathbf{w}}^{n_c} + [\mathbf{\Phi}_c]_{n_c} \end{bmatrix}$$

In this way, problem 2 is equivalent to

$$\begin{aligned} &\min_{u} \quad z_0(\mathbf{x}(0), \mathbf{u}) \\ &\text{s.t.} \\ &\qquad \lambda_j(\mathbf{u}) \leqslant \mathbf{0} \quad j = 1, \ldots, n_c \end{aligned} \tag{22}$$

Now, let us analyse the objective function $z_0(\mathbf{x}(0), \mathbf{u})$. If this problem is constrained to the kth sector, the steady-state condition is $\boldsymbol{\xi}^k_{xx}.\mathbf{x} + \boldsymbol{\xi}^k_{xu}.\mathbf{u} + \boldsymbol{\xi}^k_{xw}.\tilde{\mathbf{w}} + \boldsymbol{\eta}^k_x = 0$, or equivalently

$$\mathbf{x} = -\left(\boldsymbol{\xi}^k_{xx}\right)^{-1}.\left(\boldsymbol{\xi}^k_{xu}.\mathbf{u} + \boldsymbol{\xi}^k_{xw}.\tilde{\mathbf{w}} + \boldsymbol{\eta}^k_x\right) \tag{23}$$

Introducing this expression in (12), the objective function constrained to the kth sector could be written as

$$\mathbf{x} = \mathbf{A}^k.\mathbf{u} + \mathbf{B}^k \tag{24}$$

where $\mathbf{A}^k = \left(\boldsymbol{\xi}^k_{ou} - \boldsymbol{\xi}^k_{ox}.\left(\boldsymbol{\xi}^k_{xx}\right)^{-1}\boldsymbol{\xi}^k_{xu}\right)$, $\mathbf{B}^k = \left(-\boldsymbol{\xi}^k_{ox}.\left(\boldsymbol{\xi}^k_{xx}\right)^{-1}\boldsymbol{\xi}^k_{xw}\right).\tilde{\mathbf{w}} + \left(\eta^k_o - \boldsymbol{\xi}^k_{ox}.\left(\boldsymbol{\xi}^k_{xx}\right)^{-1}\boldsymbol{\eta}^k_x\right)$ and $\tilde{\mathbf{w}}$ is the disturbance considered as nominal.

Note that this expression of the objective function will be valid only for the steady-states solution lying in the kth *sector*. Considering conditions (13) and (23), both constraints could be written as a unique set of inequalities as,

$$\begin{aligned} \mathbf{z}^k_{\aleph} &= \boldsymbol{\xi}^k_{\aleph x}.\left(-\left(\boldsymbol{\xi}^k_{xx}\right)^{-1}\left(\boldsymbol{\xi}^k_{xu}.\mathbf{u} + \boldsymbol{\xi}^k_{xw}.\tilde{\mathbf{w}} + \boldsymbol{\eta}^k_x\right)\right) + \boldsymbol{\xi}^k_{\aleph u}.\mathbf{u} + \boldsymbol{\xi}^k_{\aleph w}.\tilde{\mathbf{w}} + \boldsymbol{\eta}^k_{\aleph} \leqslant \mathbf{0} \\ &= \mathbf{D}^k_c.\mathbf{u} + \mathbf{E}^k_c \leqslant \mathbf{0} \end{aligned} \tag{25}$$

where $\mathbf{D}^k_c = \left(\boldsymbol{\xi}^k_{\aleph u} - \boldsymbol{\xi}^k_{\aleph x}.\left(\boldsymbol{\xi}^k_{xx}\right)^{-1}.\boldsymbol{\xi}^k_{xu}\right)$ and $\mathbf{E}^k_c = \left(\boldsymbol{\xi}^k_{\aleph w} - \boldsymbol{\xi}^k_{\aleph x}.\left(\boldsymbol{\xi}^k_{xx}\right)^{-1}.\boldsymbol{\xi}^k_{xw}\right)\tilde{\mathbf{w}}$. This expression is considered as the Sector Belonging Condition for the steady-state solution.

Then, problem (22) constrained to the kth sector, could be expressed as

$$\begin{aligned} \min_{u} \quad & \mathbf{A}^k\mathbf{u} + \mathbf{B}^k \\ \text{s.t.} \quad & \\ & \mathbf{D}.\mathbf{u} + \mathbf{E} \leqslant 0 \\ & \mathbf{D}_c^k.\mathbf{u} + \mathbf{E}_c^k \leqslant 0 \end{aligned} \tag{26}$$

Note that in (26) the first set of constraints comes from the operative constraints in the original system, while the second set comes from the specific sector belonging condition (i.e. $(\mathbf{x}, \mathbf{u}, \tilde{\mathbf{w}}) \in \aleph^k$). In this situation, the optimum will be either on the boundary of the sector $\aleph^k$ or in the ones defined by the operative constraints. If the solution to the problem (22) is on the boundary of sector $\aleph^k$, at least one entry in the vector $\mathbf{D}_c^k\mathbf{u} + \mathbf{E}_c^k$ will be zero. If this occurs, we must change the sign of this entry (the rest remaining invariable) and, in this way, we go on with the optimization in the next sector $\aleph^{k+1}$. If no entry of $\mathbf{D}_c^k\mathbf{u} + \mathbf{E}_c^k$ is zero, we are in the border fixed by the operative constraints.

Also note that in the set of inequalities $\mathbf{D}.\mathbf{u} + \mathbf{E} \leqslant \mathbf{0}$ the matrices $\mathbf{D}$ and $\mathbf{E}$ are dependent on the vector of optimization variables $\mathbf{u}$; then, each time that the optimization software changes vector $\mathbf{u}$, we should check if the matrices $\mathbf{D}$ and $\mathbf{E}$ still valid. If not, they should be adapted. However, this involves only a new simulation and our computational experience shows that it is performed in less than three iterations.

Then we have solved the problem. In summary, we have the following algorithm:

Algorithm 4 (Dynamic Back-off Computation):

Data: An initial guess for $\mathbf{u}$ ($\mathbf{u}^0$) and a nominal disturbance $\tilde{\mathbf{w}}$.

Step 0: Compute a steady-state point $(\mathbf{x}^0, \mathbf{u}^0, \tilde{\mathbf{w}}) \in \aleph^0$ and the vector $\boldsymbol{\gamma}^0$, which identifies this sector, using Algorithm 1. Set $k = 0$.

Step 1: Compute the matrices $\mathbf{D}$ and $\mathbf{E}$, using Algorithm 3 and Equation (21).

Step 2: In the sector $\aleph^k$ compute the optimization variable $\mathbf{u}^k$ that solves the following minimization problem

$$\begin{aligned} \min_{u} \quad & \mathbf{A}^k\mathbf{u} + \mathbf{B}^k \\ \text{s.t.} \quad & \\ & \mathbf{D}\mathbf{u} + \mathbf{E} \leqslant \mathbf{0} \\ & \mathbf{D}_c^k\mathbf{u} + \mathbf{E}_c^k \leqslant \mathbf{0} \end{aligned}$$

Step 3: Check if the model ($\mathbf{D}$, $\mathbf{E}$) still valid for this new $\mathbf{u}^k$, if not return to Step 2.

Step 4: If the solution is at the boundary of sector $\aleph^k$ (i.e., if any entry in the vector $\mathbf{D}_c^k\mathbf{u} + \mathbf{F}_c^k$ is zero), identify the next sector $\aleph^{k+1}$. Set $k = k + 1$ and return to Step 1. Otherwise (i.e. if no entry of $\mathbf{D}_c^k\mathbf{u} + \mathbf{F}_c^k$ is zero), continue.

Mathematically, if any entry in the vector $\mathbf{D}_c^k\mathbf{u} + \mathbf{F}_c^k$ is zero (e.g. jth-entry) change the sign of the correspondent entry in the vector γ^k to obtain the new vector γ^{k+1} (i.e. vector $\gamma^{k+1}(i) = \gamma^k(i)$ for $i \neq j$ and $\gamma^{k+1}(j) = -\gamma^k(j)$).

Step 5: Compute the worst disturbance (using Algorithm 3). If any entry in the vector $\boldsymbol{\lambda}$ is larger than zero return to Step 1. Otherwise, *Stop*.

In this algorithm, the loop between Steps 3 and 2 is the most time demanding. However, usually a solution is found in few iterations and demands only few seconds of processing. In Step 4 there might be a problem of determining which region (close to $\aleph^k$) could be chosen as $\aleph^{k+1}$. Suppose that the optimum in region $\aleph^k$ is at the intersection of l hyperplanes. This means that l entries of $\mathbf{D}_c^k\mathbf{u} + \mathbf{F}_c^k$ are zero. This makes it difficult to continue with the optimization algorithm in the sector $\aleph^{k+1}$, because this new sector may be obtained by changing the sign of either of these entries of γ^k, or any combination of them. This gives $(2^l - 1)$ possibilities. It is obvious that the convergence of the algorithm depends on our choice. Figueroa and Desages [7] present three possible search methods to avoid this problem.

As is typical in non-linear optimization, the convergence of this algorithm towards the global optimum cannot always be guaranteed. However, our computational experience shows that the algorithm usually converges to the same optimal point that the one obtained using non-linear optimization algorithms. The use of the CPWL approximation does not introduce extra limitations and makes the solution of the back-off problem more efficient.

5. EXAMPLE

In this section, the algorithms presented above are analyzed by means of an example. Our attention has been focused to determine the feasibility of operation while ensuring no constraint violations, for a given set of process disturbances.

Power and steam systems, in which the boiler is a fundamental part, are a good application example due to their large operating cost and their need of satisfying specific energy demands. Despite these facts, utility systems have not received the same degree of attention as other process units when dealing with disturbance effects. One reason for this situation has been the uneasy availability of simple reliable mathematical models for boilers in the open literature. The steam-generating unit studied in this paper consists of five pulverizers supplying fuel to a 200 MW drum type boiler. Ray and Majumder [10] have developed the model for this unit.

5.1. Pulverizer model

Primary air required for this unit is supplied by two P.A. fans and then bifurcated into hot air and cold air flows for pulverizer units. A non-linear model for a single pulverizer has been developed having as inputs the feeder speed, the hot air damper opening, the cold air damper opening and the P.A. fans speed. The states for each unit are the fuel output of the pulverizer, the hot air flow and the cold air flow. The output variable is the fuel outlet from pulverizers, supplied to the boiler. Simulation of a single pulverizer unit can be performed using the following set of equations:

$$\frac{\mathrm{d}F^i}{\mathrm{d}t} = c_1^i\, u_2^i\, H^i + c_2^i\, u_3^i\, C^i + c_3^i\, F^i + c_4^i(H^i + C^i)u_7^i$$

$$\frac{dH^i}{dt} = c_5^i\, u_1^i + c_6^i\, u_6^i - H^i u_2^i - C^i u_3^i$$

$$\frac{dC^i}{dt} = c_7^i\, u_1^i + c_8^i\, u_6^i + c_9^i\, H^i u_2^i + c_{10}^i C^i u_3^i$$

where F is the fuel output of the pulverizer, H is the hot air flow, C is the cold air flow, u_1 and u_6 are the P.A. fan speed (nominally 24.7252 rad/s), u_2 is the hot air damper opening (0.8), u_3 is the cold air damper opening (0.2) and u_7 is the feeder speed (3 r.p.m.). The parameters are included in Table I.

Rest of the units will be having similar kind of dynamics. In this analysis (and for the only proposes of the algorithm demonstration) we will consider the feeder speed in each pulverizer as a disturbance, because it depends on the coal characteristics.

The manipulated variables are the hot air damper opening and the speed of the two P.A. fan units. Also, we consider that the hot and the cold air damper openings are normalized, then they should verify the following relation $u_3 = 1 - u_2$.

5.2. Boiler model

The states of the non-linear drum type coal-fired boiler model are the drum pressure (P), the steam flow to the H.P. turbine (S) and the drum level (L). It has two optimization variables, there are the fuel input from the pulverizer (F) and the feed water input (w_c, nominally 193 Kg/s). The disturbance inputs are the feed water temperature (T_e, nominally 288°C) and the control valve setting (c_v, nominally 0.8). The model is as follows

$$\frac{dP}{dt} = -0.00193SP^{1/8} + 0.014524F - 0.000736w_c + 0.00121L + 0.000176T_e$$

$$\frac{dS}{dt} = 10c_v P^{1/2} - 0.785716S$$

$$\frac{dL}{dt} = 0.00863w_c + 0.002F + 0.463c_v - 6 \times 10^{-6}P^2 - 0.00914L - 8.2 \times 10^{-5}L^2 - 0.007328S$$

Table I. Pulverizer parameters.

i	1	2	3	4	5
c_1^i	10	9	11	10	11
c_2^i	2	1.8	2.2	2.1	1.9
c_3^i	−0.073591	−0.07	−0.075	−0.071	−0.072
c_4^i	0.057306	0.06	0.063	0.059	0.055
c_5^i	0.001413	0.0013	0.0015	0.0014	0.00125
c_6^i	0.001413	0.0013	0.0015	0.0014	0.00125
c_7^i	0.003	0.0027	0.0033	0.0031	0.0028
c_8^i	0.003	0.0027	0.0033	0.0031	0.0028
c_9^i	−1.903016	−1.88	−1.93	−1.9	−1.89
c_{10}^i	−3.0	−2.9	−3.1	−3.1	−3.0

The purpose of this model is to describe the gross behaviour of the plant. The control variables for the boiler are the fuel input (from the pulverizers) and the feed water input. The disturbances are the feed water temperature and control valve setting. The last one represents the variation on the steam demand to the service units.

5.3. Control scheme

The control scheme for this system is composed of two control structures:

A control system for the boiler control. It involves two SISO loops, controlling the pressure and the drum level by using the fuel and the water feed inputs, respectively, as manipulated variables. These loops are closed for PI controllers with parameters as in Table II.

A control system for the pulverizers. In this case, the manipulated variable is the hot air damper opening (u_2) and the controlled variables are the fuel outputs of the pulverizer (F). It is important to remark that the value of the reference of these loops are computed using a divisor, and each of these controllers are a slave controller which master is the pressure loop of the boiler (i.e. $F^{\mathrm{spi}} = 0.2F^{\mathrm{sp}}, \mathrm{i} = 1, \ldots, 5$. These controllers are proportional ones with parameter $K_{\mathrm{p}} =$ 100.00.

In order to study the complete system operation, we consider the five pulverizers supplying fuel to the boiler plus the controllers, so we have a total of 20 non-linear differential equations. There are 16 freed variables considered, these are the hot air damper and the two P.A. fan speeds for each pulverizer and the feed water input at the boiler. The disturbances considered are the feeder speed for each pulverizer, the control valve displacement and the feed water temperature.

The operative constraints came from physical limitations of the process:

Minimal steam flow,	$S \leqslant 110$
Bounds on the drum pressure,	$140 \leqslant P \leqslant 200$
Bounds on the drum level,	$45 \leqslant L \leqslant 66$
Bounds on the fuel output on each pulverizer,	$5 \leqslant F_i \leqslant 9$

The objective function is to minimize the operating cost (fuel and water). In this point we will assume that all pulverizers have the same operative cost.

$$z_{\mathrm{obj}} = 0.25\, w_{\mathrm{c}} + \sum_{i=1}^{5} 10F_i$$

Table II. PI controllers.

	Loop 1 (*P*-*F*)	Loop 2 (*L*-w_{c})
P	0.75	0.05
I	50000	10

The set of free variables are allowed to move in the following ranges:

$$0.8 \leqslant u_2 \leqslant 0.95 \text{ (for each pulverizer)}$$

$$22 \leqslant u_1 \leqslant 25.5 \text{ (for each pulverizer)}$$

$$22 \leqslant u_6 \leqslant 25.5 \text{ (for each pulverizer)}$$

$$175.0 \leqslant w_c \leqslant 210.0$$

and also we allow free value for disturbances between the following limits (with the nominal value between parenthesis):

$$280.0 \leqslant T_e \; (288^\circ\text{C}) \leqslant 292.0$$

$$2.9 \leqslant u_7 \; (3 \text{ r.p.m}) \leqslant 3 \text{ (for each pulverizer)}$$

$$0.8 \leqslant c_v \; (0.8) \leqslant 0.85$$

To solve this non-linear optimization a CPWL approximation for the problem is used. To perform this CPWL model a direct unconstrained optimization algorithm was used on each individual non linearity of the model, and then, they are joined in a complete CPWL model. The variable domain results divided in 140 regions. This number of sectors had been chosen to obtain a CPWL model with enough accuracy respect to the non-linear expressions.

The optimal objective function is $Z_{\text{obj}}=$ \$358.1295 (for nominal disturbance) and it is obtained for optimized variables $u_2^i = 0.8$ for $i = 1, \ldots, 5$; $u_1^i = 22.00$ for $i = 1, \ldots, 5$; $u_6^i = 22.00$ for $i = 1, 3, 4, 5$; $u_6^2 = 24.41$ and $w_c = 175.0$.

When we allow free values for disturbances between the bounds, the optimum operating point becomes not feasible, due for example, to a increase in the drum level. We used Algorithm 3 to identify the worst disturbance for each constraint and compute matrices **D** and **E** (*Step 1 of Algorithm 4*). Using these disturbances the optimization problem of (22) is solved (*Step 2 of Algorithm 4*). This optimization is performed in five sectors before converges to the optimum back-off objective function $Z_{\text{obj}}=$ \$356.39. The values of the manipulated variables are $u_2^i = 0.8$ for $i = 1, \ldots, 5$; $u_1^i = 22.$ for $i = 1, 2, 3, 5$; $u_1^4 = 22.56$; $u_6^i = 22.$ for $i = 1, 2, 4, 5$; $u_6^3 = 22.25$ and $w_c = 175.0$. It is important to remark, as a conclusion of this example that the convergence is obtained in a few minutes, using a Pentium processor.

6. CONCLUSIONS

There is a notable trend towards the use of non-linear dynamic optimization as a tool for design/control integration. In this paper, efficient methods to solve the problem of dynamic simulation and optimization are presented in the context of back-off calculation. The solution is carrying out by using an approximation of a generic non-linear model for the system. The algorithms studied are generic and could be used in a wide number of applications.

APPENDIX A: CANONICAL PIECEWISE LINEAR FUNCTIONS

The general formulation of piecewise lineal functions allows us to write a non-linear function as several linear expressions, each of them valid in a certain region. In mathematical terms, this can be described as follows:

Let $f(x) : \Re^n \to \Re^m$ be a nonlinear function with $x \in \aleph \subset \Re^n$. The domain $\aleph$ is partitioned in σ non-empty regions, $\aleph^i$, such that $\aleph = \bigcup_{i=1}^{\sigma} \aleph^i$. In each of these regions, $\aleph^i$, the function $f(x)$ is approximated by a linear representation in the form

$$g(x) = J^{(i)}x + j^{(i)} \tag{A1}$$

where $J^{(i)} \in \Re^{mxn}, j^{(i)} \in \Re^m$ and $\|f(x) - g(x)\|_{x \in \aleph^i} < \varepsilon$.

Using this procedure the non-linear function can be rewritten in an alternative form.

If we impose additional conditions regarding to the continuity of the linear approximation through the boundary of the region $\aleph^i$ (as the ones used by Chua and Ying [11]; Chua and Deng [8]), a compact expression can be obtained for all the regions as

$$g_j(x) = a + B.x + \sum_{i=1}^{\sigma} c_i \left| \langle \alpha_i, x \rangle - \beta_i \right|$$

where $g_j(x)$ is the jth entry in the vector $g(x)$; a, c_i and β_i are scalars, and B, and α_i are vectors of appropriate dimensions. An expression like this is called a Canonical Piecewise Linear (CPWL) function.

Note that

1. the domain $\aleph$ is divided into a finite number of polyhedral regions bounded by a set of hyperplanes of the type $\langle \alpha_i, x \rangle - \beta_i$ with dimensions not lower than $\sigma - 1$;
2. the CPWL function is continuous in any boundary of neighboring regions, namely, the piecewise linear function is continuous on the planes $\langle \alpha_i, x \rangle - \beta_i$.

Note also that it is easy to determine a region containing a given point x, using the sign of the functions $\langle \alpha_i, x \rangle - \beta_i$. Thus, the vector $\text{sign}(\langle \alpha_i, x \rangle - \beta_i)$ establishes a one-to-one mapping with the defined partitions.

It has been shown (see Chua and Ying [11], Chua and Deng [8]) that the number of parameters involved in this canonical representation is far lower than the one used in the classical linear representation. The problems associated with the determination of these regions and the CPWL approximations have been extensively studied in the literature. For example, Lin and Unbehahuen [12] proved that CPWL models can approximate as much as it is desired a continuous non-linear function on a compact set of variables. Lin and Unbehahuen [13] proposes an algorithm for adjust a CPWL model of one variable, this method could be applied in conjunction with the algorithm due to Yamamura [14] that allows the representation of a general non-linear function as a superposition of non-linear terms of one variable. Julián [15] presents an algorithm for the CPWL approximation of smooth functions. The CPWL representation uses in this paper presents some lack of flexibility that higher order models [15, 16] achieve. However, this flexibility involves a more complex description for non-linearities.

REFERENCES

1. Grossmann IE, Halamane KP, Swaney RE. Optimization strategies for flexible chemical processes. *Computers and Chemical Engineering* 1983; **7**:439–462.
2. Dimitriadis VD, Pistikopoulos EN. Flexibility analysis of dynamic systems. *Industrial and Engineering Chemistry Research* 1995; **34**:4451–4462.
3. Bandoni JA, Romagnoli JA, Barton GW. On optimising control and the effect of disturbances: calculations of the open-loop back-offs. *Computers and Chemical Engineering* 1994; **18/S**:505–509.
4. Figueroa JL, Bahri PA, Romagnoli JA. Economic impact of disturbances in process control-a dynamical analysis. In *Interactions Between Design and Process Control*, Zafirou J (ed.). Pergamon Press: Baltimore, 1994; 141–146.
5. Perkins JD, Walsh SPK. Optimization as a tool for design/control integration. In *Interactions Between Design and Process Control*, Zafirou J (ed.). Pergamon Press: Baltimore, 1994; 1–10.
6. Narraway L, Perkins J. Selection of process control structure based in economics. *Computers and Chemical Engineering* 1994; **18/S**:511–515.
7. Figueroa JL, Desages AC. Use of piecewise linear approximations for steady-state back-off analysis. *Optimal Control: Applications and Methods* 1998; **19**:93–110.
8. Chua LO, Deng AC. Canonical piecewise-linear modeling. *IEEE Transactions on Circuits and Systems* 1986; **33**: 511–525.
9. Chua LO, Deng AC. Canonical piecewise-linear analysis: generalized breakpoint hopping algorithm. *International Journal of Circuit Theory* 1986; **14**:35–52.
10. Ray KS, Dutta Majumder YD. Simulation of a nonlinear steam generating unit. *Proceedings of the International Conference on Systems, Man and Cybernetics*. Bombay and New Delhi, India, 1983; 705–708.
11. Chua LO, Ying LP. Canonical piecewise-linear analysis. *IEEE Transactions Circuits and Systems* 1983; **30**:125–140.
12. Lin JN, Unbehauen R. Adaptive nonlinear digital filter with canonical piecewise-linear function structure. *Transactions on Circuits and Systems* 1990; **37**:347–353.
13. Lin JN, Unbehauen R. Canonical piecewise-linear approximations. *IEEE Transactions on Circuits and Systems* 1992; **39**:697–699.
14. Yamamura K. An algorithm for representing functions of many variables by superpositions of functions of one variable and addition. *IEEE Transactions on Circuits and Systems* 1996; **43**:338–340.
15. Julián P, Jordán M, Desages A. Canonical piecewise linear approximation of smooth functions. *IEEE Transactions on Circuits and Systems* 1998; **45**:567–571.
16. Kahlert C, Chua LO. The complete canonical piecewise-linear representation. Part I: The geometry of the domain space. *IEEE Transactions on Circuits and Systems I* 1992; **39**:222–236.

