Neural Comput & Applic (2006) 15: 339–349
DOI 10.1007/s00521-006-0035-0

ORIGINAL ARTICLE

K. Najim · E. Ikonen · P. Del Moral

Open-loop regulation and tracking control based on a genealogical decision tree

Received: 25 February 2005 / Accepted: 4 January 2006 / Published online: 31 March 2006

Abstract The goal of this paper is to design a new control algorithm for open-loop control of complex systems. This control approach is based on a genealogical decision tree for both regulation and tracking control problems. The idea behind this control strategy consists of associating Gaussian distributions to both the norms of the control actions and the tracking errors. This stochastic search model can be interpreted as a simple genetic particle evolution model with a natural birth and death interpretation. It converges on probability. A numerical example dealing with the control of a fluidized bed combustion power plant illustrates the feasibility and the performance of this control algorithm.

Keywords Monte Carlo method · Optimal control · Optimization problems · Particle filtering · Population based search · Power plants

1 Introduction

Control of complex systems has attracted the interest of many researchers. Lately, several workshops, research cluster/networks projects, and summer schools have been dedicated to dynamics and control of complex systems. There is no commonly accepted definition for such systems. A complex system can be characterized by the lack of information (time varying or stochastic systems [1], etc.), nonlinear behavior, or by the fact that it is impossible to associate one of available control approaches to the considered system. Complex systems are difficult to control. The linear control theory, which is well stated, does not permit the derivation of control strategies for these kind of systems. Indeed, the state of nonlinear systems analysis is not nearly complete. The control and modeling of such systems requires a deep understanding and experience of a large spectrum of modeling and control techniques.

Complex systems can be classified into two main categories: continuous and batch systems. In the fine chemistry, many processes are operating in batch mode, are repetitive in nature and operate over a fixed time interval. The iterative learning control approach [2] incorporates past control information (such as tracking errors and control input signals) from previous runs into the construction of the present control action.

The complexity of a system does not correlate with its scale. Indeed, it is, for example, more easy to derive a control policy for an industrial phosphate drying furnace of 40 m long, than for a rapid thermal system used in semiconductor wafer fabrication process [3]. Note also that the complexity can derive from multiple simple dynamic components that interact in varying and complex ways. For different reasons (improved conversion and selectivity, heat integration benefits and avoidance of azeotropes, etc.), chemical engineers are now concerned with process intensification [4], which generally leads to very simple systems. For example, the manufacturing of methyl acetate is usually done in a plant consisting of a chemical reactor and nine distillations columns. This manufacturing can be done in a single reactive distillation. The resulting reactive distillation process is very simple and more economical.

K. Najim was partially supported by UK EPSRC Research cluster project, grant no. GR/S63779/. E. Ikonen was supported by the Academy of Finland, projects nos. 48545 and 203231.

K. Najim
Process Control Laboratory, E.N.S.I.A.C.E.T., 118, route de Narbonne, 31077 Toulouse Cedex 4, France
E-mail: Kaddour.Najim@ensiacet.fr

E. Ikonen (✉)
Systems Engineering Laboratory, Department of Process and Environmental Engineering, University of Oulu, P.O.Box 4300, 90014 Oulu, Finland
E-mail: Enso.Ikonen@oulu.fi
Tel.: +358-8-5531011
Fax: +358-8-5532439

P. Del Moral
Laboratoire J.-A. Dieudonné, Université de Nice—Sophia Antipolis, Parc Valrose, 06108 Nice Cedex 02, France
E-mail: delmoral@math.unice.fr

Compared to controlling a set consisting of a reactor, a number distillation columns, pumps, and heat exchangers, it is relatively easy to control a single distillation column.

The development of high performance digital computers and later evolutionary algorithms [5] has introduced new tools for optimizing and controlling complex systems. Genealogical decision trees possess several properties that make them particularly attractive for applications to modeling and control of complex systems. Among these properties are their flexibility, robustness and applicability to both linear and non linear systems, and for both continuous and batch processes. On the other hand, their asymptotic behavior with respect to the time horizon or the particle system size is nowadays well understood. These recent results allow us to quantify with precision their ability to solve any optimal control problem which can be interpreted in terms of a nonlinear filtering problem. The first heuristic schemes of these control particle models appeared in [6]; some recent results were reported in [7]. One objective of the present article is to connect these tree based models with the recently developed asymptotic analysis [8]. This control algorithm falls within the areas of mathematical population genetics and interacting particle systems.

The control approach suggested in this paper is suitable for processes for which there exist no on-line sensors for the controlled outputs (semiconductors manufacturing, chemistry, biotechnology, etc.), or on-line analyzers (concentration measurement, etc.) are very expensive and need high maintenance costs. Soft sensors and the inferential control approach [9] have been dedicated to solve these process control problems where the outputs measurements are available only off-line. Soft sensors are based on models relating the desired controlled variable to some easily measured variables. The inferential control approach is mainly based on the prediction of the process outputs over the interval separating two successive measurements, or on the on-line measured variables which are correlated to the controlled variables. Notice that the interval between two measurements is usually not constant. Indeed, some measurements are carried out in laboratory. As a consequence, the efficiency of soft sensors and inferential control approaches are limited by the adopted model for the sensor, and the characteristics and the behavior of the predictor of the outputs. Open-loop control is employed extensively for the execution of fast movements where feedback propagates too slowly, for example, to affect the motor response in human voluntary movement [10]. To our knowledge, there exist no algorithms for solving general open-loop tracking control problems.

The remainder of this paper is organized as follows: the control problem is formulated in the next section. The control algorithm under consideration is described in Sect. 3, and its analysis is given in Sect. 4. Section 5 presents a numerical example. Some concluding remarks ends this paper.

2 Problem formulation

Let us consider a complex system described by the following time-varying state representation:

$$X_n = F_n(X_{n-1}, U_n), \quad \text{with } X_n \in \mathscr{R}^S,\ n = \overline{1, T},\ X_0, \tag{1}$$

where X_n is a column vector. The control sequence is assumed to be a P-dimensional column vector and it is denoted by

$$U_n \in \mathscr{R}^P.$$

The output sequence is given by a Q-dimensional equation of the form

$$Y_n = h_n(X_n), \quad Y_n \in \mathscr{R}^Q. \tag{2}$$

In the above displayed formulae, the index n and X_0 represent the time and the initial states, respectively. Let A_n and B_n be symmetric and semi-definite positive covariance matrices. The control objective (finite horizon T) is given by

$$J_T(U_1, \ldots, U_T) = \sum_{n=1}^{T} \|U_n\|^2_{A_n} + \sum_{n=1}^{T} \left\|Y_n - Y_n^{\text{ref}}\right\|^2_{B_n} \tag{3}$$

where $Y_n^{\text{ref}} \in \mathscr{R}^Q$ represents the reference (desired) trajectories, and $||U||_A = U^{\mathrm{T}}\ A^{-1}U$.

Our objective is to find the sequence of controls actions that minimizes this cost function in open-loop control.

3 Control algorithm

Let X_0 be the fixed initial states. In what follows, we shall present the different steps related to a decision tree based algorithm. This particle search technique can be interpreted as the historical process associated with a genetic type evolution model.

The idea behind this control algorithm consists of associating Gaussian distributions to both the norms of the control actions and the tracking errors, i.e.,

$$\begin{aligned} \|U_n\|^2 &\longrightarrow \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{\beta}{2}\|U_n\|^2_{A_n}\right) \\ \left\|Y_n - Y_n^{\text{ref}}\right\|^2 &\longrightarrow \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{\beta}{2}\left\|Y_n - Y_n^{\text{ref}}\right\|^2_{B_n}\right). \end{aligned} \tag{4}$$

The parameter β is similar to the inverse of the temperature in simulated annealing optimization algorithms. Indeed, for large value of β, the probability distribution will have the form of a hair pin.

This duality between cost/performance functions and probability measures can obviously be extended to more general situations. For instance, for bang/bang control problems the cost function is rather given by an expression of the form

$$J_T(U_1,\ldots,U_T)=\sum_{n=1}^{T}\alpha_n U_n+\sum_{n=1}^{T}\left\|Y_n-Y_n^{\mathrm{ref}}\right\|_{B_n}^2$$

with control sequences $U_1,\ldots,U_T$ taking values in {0,1}, and for some strictly positive sequence of parameter α_n. In this context the duality is given as above by replacing the Gaussian distribution (4) by the distribution of a random variable U_n which takes the value 0 with probability

$$p_n=\frac{1}{1+\exp(\alpha_n)}$$

and 1 with probability $1-p_n$. In other words, the Gaussian distribution (4) is replaced by $\Pr(U_n=0)=1-\Pr(U_n=1)=p_n$.

Step 1 The first transition can be decomposed into two traditional mutation/selection genetic type mechanisms. The mutation procedure consists of sampling randomly a sequence of decisions, and the selection stage consists of selecting randomly the best fitted decisions. These two steps can also be interpreted as local prediction and updating search mechanisms.

Step 1.0 (Local prediction of the optimal control) Compute N independent and identically distributed normal variables $\mathcal{N}(0,A_1)$

$$U_1^1,U_1^2,\ldots,U_1^N.$$

These control values lead to a series of N outputs

$$Y_1^1=h_1\left(X_1^1\right),Y_1^2=h_1\left(X_1^2\right),\ldots,Y_1^N=h_1\left(X_1^N\right),$$

with

$$X_1^i=F_1\left(X_0,U_1^i\right),\ i=\overline{1,N}.$$

Step 1.1 (Local selection of the control toward the tracking trajectory) In order to simplify the notations, let us introduce the following terms

$$p_1^i=\frac{\exp\left(-\frac{\beta}{2}\left\|Y_1^{\mathrm{ref}}-Y_1^i\right\|_{B_1}^2\right)}{\sum_{j=1}^N\exp\left(-\frac{\beta}{2}\left\|Y_1^{\mathrm{ref}}-Y_1^j\right\|_{B_1}^2\right)}. \tag{5}$$

for $i=1,\ldots,N$ where $\sum_{i=1}^N p_1^i=1$.

Generate N independent and identically distributed random variables $\hat{U}_1^1,\hat{U}_1^2,\ldots,\hat{U}_1^N$ according to the following discrete distribution (see Appendix A)

$$p_1=\sum_{i=1}^N p_1^i\delta_{U_1^i},$$

where δ_u is the Dirac measure at the control value $u\in\mathscr{R}^P$. In other words, for each $k=\overline{1,N}$, each random control $\hat{U}_1^k$ takes the value U_1^i with probability equal to p_1^i.

The implementation of these control actions leads to

$$\hat{X}_1^1=F_1\left(\hat{X}_0,\hat{U}_1^1\right)\Longrightarrow\hat{Y}_1^1=h_1\left(\hat{X}_1^1\right),\ \hat{X}_0=X_0$$
$$\hat{X}_1^2=F_1\left(\hat{X}_0,\hat{U}_1^2\right)\Longrightarrow\hat{Y}_1^2=h_1\left(\hat{X}_1^2\right)$$
$$\vdots$$
$$\hat{X}_1^N=F_1\left(\hat{X}_0,\hat{U}_1^N\right)\Longrightarrow\hat{Y}_1^N=h_1\left(\hat{X}_1^N\right).$$

Step 2 *Step 2.0* (Local prediction of the optimal control at the second step) As in the first step, we generate a series of independent and identically normally distributed ($\mathcal{N}(0,A_2)$) random variables

$$U_2^1,U_2^2,\ldots,U_2^N.$$

These control values lead to a series of N outputs

$$X_2^1=F_2\left(\hat{X}_1^1,U_2^1\right)\Longrightarrow Y_2^1=h\left(X_2^1\right)$$
$$\vdots$$
$$X_2^N=F_2\left(\hat{X}_1^N,U_2^N\right)\Longrightarrow Y_2^N=h\left(X_2^N\right).$$

Step 2.1 (Local selection of the control toward the tracking trajectory) Generate

$$\hat{U}_2^1,\hat{U}_2^2,\ldots,\hat{U}_2^N$$

according to

$$\sum_{i=1}^N p_2^i\delta_{U_2^i},\quad\text{with}\quad p_2^i=\frac{\exp\left(-\frac{\beta}{2}\left\|Y_2^{\mathrm{ref}}-Y_2^i\right\|_{B_2}^2\right)}{\sum_{j=1}^N\exp\left(-\frac{\beta}{2}\left\|Y_2^{\mathrm{ref}}-Y_2^j\right\|_{B_2}^2\right)}.$$

Now suppose that the selected control sequence is given by

$$\left(\hat{U}_2^1,\hat{U}_2^2,\ldots,\hat{U}_2^N\right)=\left(U_2^{i_1},U_2^{i_2},\ldots,U_2^{i_N}\right)$$

for some index sequence $i_1,\ldots,i_N$ in [1, N]. These control actions lead to

$$\hat{X}_2^1=F_2\left(\hat{X}_1^{i_1},U_2^{i_1}\right)\Longrightarrow\hat{Y}_2^1=h\left(\hat{X}_2^1\right)$$
$$\vdots$$
$$\hat{X}_2^N=F_2\left(\hat{X}_1^{i_N},U_2^{i_N}\right)\Longrightarrow\hat{Y}_2^N=h\left(\hat{X}_2^N\right).$$

Step 3 Repeat Step 2 for $n=3$, 4, ..., T. Appendix B gives a pseudo-code mechanization of the algorithm.

3.1 A genealogical decision tree model

This stochastic search model can be interpreted as a simple genetic particle evolution model. These evolutionary algorithms have a natural birth and death interpretation. More precisely, each controlled state $\hat{X}_n^i$ results from the selection of a random control action, say $\hat{U}_n^i=U_n^j$, for some j. That is, we have that

$$\hat{X}_n^i = F_n\left(\hat{X}_{n-1}^j, U_n^j\right).$$

In this case, we can interpret the state $\hat{X}_{n-1}^j$ as the parent of the individual $\hat{X}_n^i$ at level $(n-1)$. In the same way, the parent individual $\hat{X}_{n-1}^j$ results from the selection of a random control action, say $\hat{U}_{n-1}^j = U_{n-1}^k$, for some k, that is we have that

$$\hat{X}_{n-1}^j = F_{n-1}\left(\hat{X}_{n-2}^k, U_{n-1}^k\right).$$

Arguing as above, we can interpret the state $\hat{X}_{n-2}^k$ as the parent of the individual $\hat{X}_{n-1}^j$, and therefore, as the ancestor of the individual $\hat{X}_n^i$ at level $(n-2)$. Running back in time, we can trace the complete ancestral line of the current individual

$$\hat{X}_{1,n}^i \longleftarrow \cdots \longleftarrow \hat{X}_{n-2,n}^i (= \hat{X}_{n-2}^k) \longleftarrow \hat{X}_{n-1,n}^i (= \hat{X}_{n-1}^j) \longleftarrow \hat{X}_{n,n}^i = \hat{X}_n^i.$$

We define in the same way, the ancestral decision line of the corresponding control actions

$$\hat{U}_{1,n}^i \longleftarrow \cdots \longleftarrow \hat{U}_{n-2,n}^i (= \hat{U}_{n-2}^k) \longleftarrow \hat{U}_{n-1,n}^i (= \hat{U}_{n-1}^j) \longleftarrow \hat{U}_{n,n}^i = \hat{U}_n^i.$$

At the final horizon time T, we obtain the approximating optimal open-loop control actions

$$\hat{U}_{1,T}^I \longleftarrow \cdots \longleftarrow \hat{U}_{T-2,T}^I \longleftarrow \hat{U}_{T-1,T}^I \longleftarrow \hat{U}_{T,T}^I = \hat{U}_T^I,$$

where the index label I is chosen so that

$$\inf_{i=\overline{1,N}} J_n\left(\hat{U}_{1,n}^i, \hat{U}_{2,n}^i, \ldots, \hat{U}_{n,n}^i\right) = J_n\left(\hat{U}_{1,n}^I, \hat{U}_{2,n}^I, \ldots, \hat{U}_{n,n}^I\right).$$

4 Asymptotic analysis

There exist many results on the asymptotic analysis of the genetic evolutionary models presented in this article. For instance, in advanced signal processing, these interacting particle algorithms, and their genealogical tree models, provide a powerful stochastic and adaptive grid approximation for solving nonlinear filtering and smoothing problems. In our optimal control/tracking context and in some sense, we can prove that, for any time horizon n, we have for any bounded and measurable function φ_n on $\mathscr{R}^{P\times n}$

$$\frac{1}{N}\sum_{i=1}^{N} \varphi_n\left(\hat{U}_{1,n}^i, \hat{U}_{2,n}^i, \ldots, \hat{U}_{n,n}^i\right) \underset{N\to\infty}{\to} E\Big[\varphi_n(W_1, W_2, \ldots, W_n) \,|\, Y_1 = Y_1^{\text{ref}}, \ldots, Y_n = Y_n^{\text{ref}}\Big]. \tag{6}$$

The conditional expectation in the above display corresponds to the conditional distribution of the signal disturbance filtering problem given by the equations

$$\begin{aligned} X_n^w &= F_n\left(X_{n-1}^w, W_n\right) \\ Y_n &= h_n\left(X_n^w\right) + V_n \end{aligned}, \tag{7}$$

where W_n and V_n are independent centered Gaussian random vectors, with respective covariance matrices A_n/β and B_n/β.

To find the control actions which minimize the considered control objective is equivalent to look for the most likely actions W_n. To understand the duality between this filtering model and the control problem discussed in this work, we observe that

$$\begin{aligned} &\Pr((W_1, W_2, \ldots, W_n) \in d(w_1, w_2, \ldots, w_n) \\ &|Y_1 = Y_1^{\text{ref}}, \ldots, Y_n = Y_n^{\text{ref}}) \\ &= \frac{1}{Z_n}\exp\left(-\frac{\beta}{2}\left(\sum_{k=1}^{n}\|w_k\|_{A_k}^2 + \sum_{k=1}^{n}\left\|Y_k^{\text{ref}} - h_k\left(X_k^w\right)\right\|_{B_k}^2\right)\right) \\ &\times dw_1 \cdots dw_n \\ &= \frac{1}{Z_n}\exp\left(-\frac{\beta}{2}(J_n(w_1, \ldots, w_n))\right) dw_1 \cdots dw_n, \end{aligned} \tag{8}$$

where $d(w_1, w_2, \ldots, w_n)$ stands for an infinitesimal neighborhood of the perturbation sequence $(w_1, w_2, \ldots, w_n) \in \mathscr{R}^{P\times n}$, Z_n is a normalizing constant, and dw_k represents the Lebesgue measure on $\mathscr{R}^P$. Loosely speaking, the above expression indicates that the conditional probability mass is concentrated around the optimal control sequence. The approximation result (6) can be used to estimate the desired control sequence.

To get one step further in our discussion, let us denote by μ_n the conditional distribution defined in (8). Using Theorem 8.3.3 in [8], if we let $\mu_n^{\otimes q}$ be the q-tensor product of the measure μ_n, then we know that

$$\left\|\text{Law}\left(\left(\hat{U}_{1,n}^i, \hat{U}_{2,n}^i, \ldots, \hat{U}_{n,n}^i\right)_{i=\overline{1,q}}\right) - \mu_n^{\otimes q}\right\|_{tv} \leq \frac{q^2}{N}c(n)$$

for some finite constant $c(n)$, and for any $q < N$, and any time horizon n. In the above display, $\|\mu - \nu\|_{tv}$ stands for the total variation distance between two probability measures μ and ν defined by

$$\|\mu - \nu\|_{tv} = \sup\left\{\left|\int(\mu(\mathrm{d}x) - \nu(\mathrm{d}x))f(x)\right|, f : \|f\| < 1\right\}.$$

The rationale behind this result is that the first q decision control lines,

$$\left(\hat{U}_{1,n}^i, \hat{U}_{2,n}^i, \ldots, \hat{U}_{n,n}^i\right), \quad i = 1, \ldots, q,$$

are asymptotically independent with the same distribution (8).

It is out of the scope of this paper to provide a detailed analysis of these genealogical tree simulation models. For a more thorough discussion we refer the reader to [8]. These models can be interpreted as a mean field particle interpretation of the flow of conditional distributions introduced in (8). This probabilistic interpretation also provides a stochastic and adaptive grid approximation of the desired conditional distributions. This random grid is more refined on control regions with high probability mass. These regions correspond to low and minimal energy control actions. In this sense, the genealogical tree models provide a powerful technique to track the optimal decision sequence. In this connection, we observe that the genealogical particle decision tree consists of approximate samples according to the conditional distribution (8). As a consequence, the probability to have high energy controls is exponentially small. In this sense, the former particle search algorithm avoids the generation of high energy decisions.

Thanks to previous propagation of chaos estimate, one can prove easily the following convergence in probability

$$\lim_{N\to\infty} \underline{\inf}_{i=1,q(N)} J_n\left(\hat{U}^i_{1,n}, \hat{U}^i_{2,n}, \ldots, \hat{U}^i_{n,n}\right) = \underline{\inf}_{U_1,\ldots,U_n} J_n(U_1,\ldots,U_n)$$

for any increasing sequence of block sizes $q(N) = o\left(\sqrt{N}\right)$, where inf J_n stands for the essential infinimum of the control objective function J_n with respect to the Lebesgue measure on $\mathscr{R}^{P\times n}$. To prove this claim, we note that for any $\delta > 0$

$$\begin{aligned} \Pr\Bigg(&\underline{\inf}_{i=1,q(N)} J_n\left(\hat{U}^i_{1,n}, \hat{U}^i_{2,n}, \ldots, \hat{U}^i_{n,n}\right) \\ &> \underline{\inf}_{U_0,\ldots,U_n} J_n(U_1,\ldots,U_n) + \delta\Bigg) \\ &\leq \frac{q^2(N)}{N} c(n) \\ &\quad + \left(1 - \mu_n\left(J_n < \underline{\inf}_{U_1,\ldots,U_n} J_n(U_1,\ldots,U_n) + \delta\right)\right)^{q(N)}. \end{aligned}$$

5 Numerical example

Let us illustrate the behavior of the genealogical decision tree approach in the design of an optimal control sequence for a fluidized bed combustion (FBC) power plant. The simulations were based on a model of a sawdust fired FBC power plant.

5.1 FBC process

A schematic drawing of a FBC power plant is shown in Fig. 1. The combustion chamber contains a large quantity of finely divided particles (called bed material) such as sand. The primary combustion air lifts these particles until they form a turbulent bed which behaves like a boiling fluid. The fuel is added to the bed and the mixture of sand and fuel is kept in constant movement by the primary air. The fuel ignites almost immediately, and the heat released as the material burns maintains the bed temperature; the turbulence keeps the temperature uniform through the bed.

The heat released in combustion is captured by heat exchangers and used for the generation of electricity, steam, or both. The steam generation system of a boiler consists of a drum system (drum with water tubes) and a superheater system (superheaters with attemperators). The drum is fed by the feed water generated by the water supply system. The water is evaporated by the heat released in combustion and the fresh steam out of the drum goes through the superheaters.

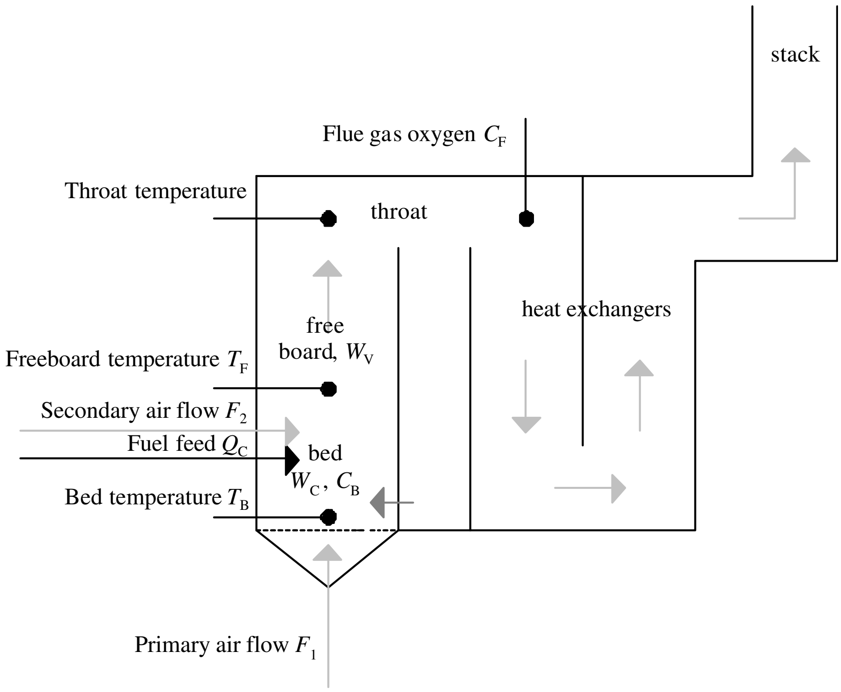
In general, FBC plants are distinguished by low average combustion temperatures (~850°C), high combustion efficiency ~95%, flexibility to different fuels and changes in fuel quality, low NO_x emissions, easy SO_2 reduction, high excess air levels (30%), intermediate particle sizes (1–3 mm), long residence times of fuel particles (several minutes), and vigorous particle motion (which dominates heat transfer and reaction processes). The fluidized bed boiler is particularly suitable when it is intended to use widely differing fuels with varying heat values. This is the case when using fuels with low heat values (together with coal as a support fuel), coal with high sulphur content, fuels which contain a high proportion of fines, etc. The lighter fractions of the fuel can be returned to the reactor through separators. Multi-fuel capability is also due to the ability to alter the heat transfer coefficient by shifting bed inventory from the lower to the upper furnace. The plant units range from small units of less than 3 MW up to over 550 MW for biomass fuels and 1,000 MW using coal.

A process model has been developed for the plant, consisting of seven differential equations describing the fuel inventories, temperatures, and oxygen concentrations in the bed and freeboard of the furnace, as well as the generated superheated steam power. The model is given in Appendix C (see also [11]). The parameters of the model were fine-tuned according to measurements from a 25 MW district heating plant using saw-dust as main fuel. Earlier studies include modeling and control of fuel inventory [12] and modeling and optimization of flue gas emissions (see, e.g., [13]), among many others (see [14] and references from there).

5.2 Control of FBC

The product of the process is the superheated steam, often further converted into district heat and/or electrical power. Typically, the power level (steam pressure) is feedback controlled using the fuel feed rate. The fuel feed rate gives a set point for the primary air, in order to maintain safe stoichiometric and fluidization conditions in the bed. The oxygen level is kept at a constant level

Fig. 1 Fluidized-bed combustion (FBC) power plant



using the O_2-trim, which manipulates the secondary air flow.

The flue gas O_2 measurement can also be used to give a fast indication of the combustion conditions, and further in calculation of the combustion power. The fuel inventory models the amount of unburned char in the bed, which, depending on fuel and furnace conditions, is more or less instantly available for combustion. Therefore it can potentially serve as a control element during transients, such as load and fuel changes. The formation of fuel inventory depends heavily on the fuel (char, wood, peat, etc.) and its properties (such as the fuel particle size distribution). The power and O_2 are typically measured on-line, whereas the fuel inventory cannot be measured during normal operation (see [12]). A model based control approach is therefore required.

Multivariable control of the FBC plant can be considered starting from linear static descriptions (using the relative gain array method) or linear(ized) dynamic models using the multivariable state space generalized predictive control (GPC). These approaches for FBC were considered in [11]. For a restricted class of nonlinear systems, an inverse-model based approach can be considered. For a general nonlinear control design, approaches such as cell-to-cell mapping [15] or controlled Markov chains [16] can be applied, leading to dynamic programming problems. These approaches rely on discretization of state-space, however, and are impractical when the dimensions of the system increase. The genealogical decision tree attempts to solve the optimization problem using a randomized model-based approach.

5.3 Simulations

In the simulations using the genealogical decision tree approach, a three-input three-output multiple-input multiple-output (MIMO) open-loop control problem was considered. The control inputs were chosen to be the fuel feed rate [kg/s], the primary air flow, and the secondary air flow [Nm^3/s], which are typically used in the control of this type of plant. The fuel inventory [kg], flue gas oxygen [Nm^3/Nm^3], and the superheated steam power [MW] were the controlled variables.

The control objective consisted of two phases: (a) An increase in fuel inventory from initial steady state 150–300 kg following a 10 min ramp trajectory; (b) An increase of the load from 21 to 26 MW following a 10 min ramp trajectory. During the load change, the flue gas oxygen was to be kept at a constant 3.1%.

An incremental control was used in the optimization, i.e., the control action applied to the plant was given by

$$V_n = U_n + V_{n-1},$$

where V_n was the plant input vector [fuel feed, primary air, secondary air], and the increment, U_n, was optimized using the genealogical decision tree scheme.

In general, the following parameters need to be set: A_n and B_n (cost function weight matrices), N (number of particles, i.e., decisions lines in parallel), and β (inverse temperature).

- Selection of A_n and B_n: In the simulations, A_n and B_n were taken to be diagonal (no prior knowledge on the relation between inputs, output costs were fully decoupled). The distribution of the i.i.d. random vectors

U_n^i will always be zero mean, the matrix A_n effects the covariance of the distribution. A large A_n distributes the realizations wider across the search space $\mathscr{R}^P$, but will also increase the costs in the control objective J_T. Setting the diagonal elements of B_n equal gives an equal and time-invariant emphasis on the error at the outputs. A reasonable initial guess in tuning the B_n can be obtained by selecting the diagonal elements in B_n so that a suitable importance in the cost is obtained (balance between the output deviation and control terms in the cost function).

- Parameter N: With small N, only a rough control sequence will be obtained, while with large N the randomness in the optimization disappears. A large N will result in unfeasible computational burden, however. For the model and the implementation considered, N up to several thousands was calculated with little effort on a standard office PC. Most of the computational load was due to numerical integration of the plant differential equations.
- The parameter β can be seen as a scaling parameter. For large values of β, the distribution for $\hat{U}_n^i$ will be sharp; small positive values of β result in more variability in the different ancestral decision lines. For an initial guess, $\beta=1$ should give reasonable values, provided that the A_n and B_n were tuned as described above. Clearly, the selection of β is inversely related to the scale of B_n. The selection of β is also linked with that of N, in that for large values of β a large N may be needed in order to have sufficient variability in the search space.

In the FBC simulations, the algorithm parameters were set to $A_n=(1/20)\,\mathrm{diag}\,[4^2, 4^2, 12^2]$, $B_n=\mathrm{diag}\,[50^2, 0.002^2, 1^2]$, $\beta=1$, based on the scales of the variables, the population size was set to $N=10{,}000$. Setting the control interval to 15 s results in trajectories of length $T=130$.

Figures 2, 3, and 4 show the plant inputs and outputs in a typical simulation. Figure 2 shows the fuel feed rate control sequence, Q_C, and the primary and secondary air flows, F_1 and F_2. The main characteristics of the fuel feed rate follow the trajectory for power demand (dotted line in Fig. 4). The requirement for a constant flue gas oxygen (dotted line in Fig. 3) results in that the air flow levels increase simultaneously with the fuel feed rate. The control sequences for Q_C, F_1, and F_2 appear realizable, even if some filtering could be useful so as to reduce jittering.

Despite the large changes in the process, the flue gas oxygen is kept well in the vicinity of the given set point, and the bed and freeboard temperatures remain between allowable limits (Fig. 3). The bed oxygen concentration, C_B, is an idealized quantity describing average bed conditions and cannot be measured on-line.

The two desired ramp trajectories are shown in Fig. 4. The superheated steam power, P, follows closely the given trajectory. The plant model also gives an increased bed fuel inventory as desired (bottom right plot, W_C). The fuel inventory in the freeboard, W_V, is negligible as it consists of volatiles.

We conclude that the genealogical decision tree algorithm was able to find a feasible solution to the 3×3 tracking problem. The solution given by the approach is an open-loop control sequence. In real applications, feed-back control is required to diminish the effect of disturbances acting on the process. A straightforward way to use the genealogical decision tree optimization is to add simple PI-controllers to compensate for disturbances (see [14]).

It is the belief of the authors that these simulations are indicative of results that can be obtained in real situations, and demonstrate the efficacy of this open-loop regulation and tracking control algorithm.

6 Conclusions

A new control algorithm for open-loop control of complex systems is suggested. The approach is based on a genealogical decision tree for tracking control problems. The idea behind this control strategy consists of associating Gaussian distributions to both the norms of the control actions and the tracking errors. This stochastic search technique can be interpreted as a simple genetic particle evolution model with a natural birth and death interpretation. It converges in probability. A numerical example, drawn from FBC power plant control, illustrated the operation and good performance of this control algorithm.

7 Appendix A: Generation of random variables distributed according to a discrete distribution

Let us describe a way for generating the set of control actions $\hat{U}_1^1, \hat{U}_1^2, \ldots, \hat{U}_1^N$ according to the discrete distribution p_1. As an aside, the forthcoming simulation technique is well known for simulating uniform order statistics. Loosely speaking, this simulation method consists of sampling a uniform and ordered sequence of random variables

$$0<D^1<D^2<\cdots<D^N<1,$$

then duplicating the best fitted decisions U_1^i in accordance with the number of D^j in the interval

$$\left[\sum_{j=1}^{i-1} p_1^j, \sum_{j=1}^{i} p_1^j\right].$$

The uniform ordered sequence is sampled using renormalized exponential times. This "global" strategy avoids the use of N separate tests to detect the interval where a given uniform random variable falls. Notice that the latter simulation method requires N^2 operations to perform the selection/updating stage, while the one based

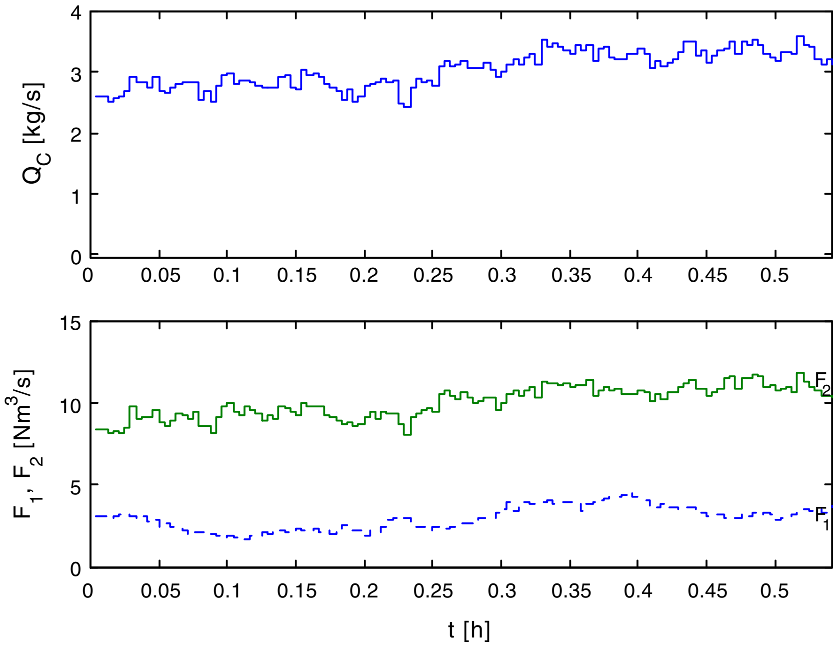


Fig. 2 FBC control signals. *Upper plot* shows the fuel feed rate. The *lower plot* shows the primary and secondary air flows (*dashed* and *solid lines*, respectively)

on ordered statistics only requires N elementary simulations.

To describe more precisely this technique, we start by generating $(N+1)$ random variables uniformly distributed on the interval [0,1], and consider their opposite logarithm

$$-\log(\text{uniform on } [0,1])$$

which gives

$$R^1, R^2, \ldots, R^{N+1}$$

Let us now consider their cumulative sums, i.e.,

$$\overline{R}^1 = R^1, \quad \overline{R}^2 = R^1 + R^2, \ldots, \quad \overline{R}^{N+1} = R^1 + \cdots + R^{N+1},$$

$$D^i = \frac{\overline{R}^i}{\overline{R}^{N+1}}.$$

In order to be able to associate these numbers to probability, let us scale them as follows

$$D^i = \frac{\overline{R}^i}{\overline{R}^{N+1}} \in [0,1].$$

Finally, the selection of the control actions is selected according to the schematic diagram given in Fig. 5. As

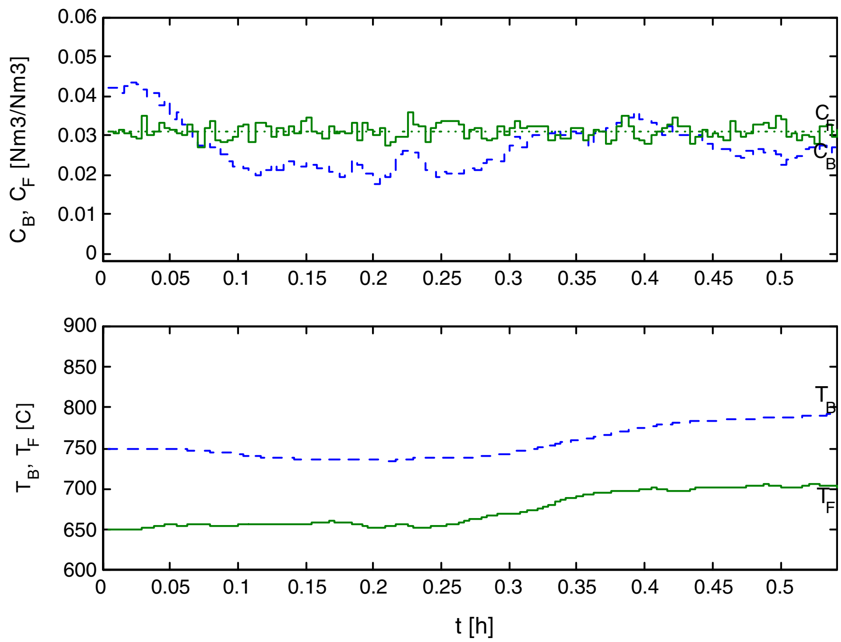


Fig. 3 FBC O_2 and temperature signals. The *upper plot* shows the bed and flue gas O_2 (*dashed* and *solid lines*, respectively). The *lower plot* shows the bed and freeboard temperatures (*dashed* and *solid lines*)

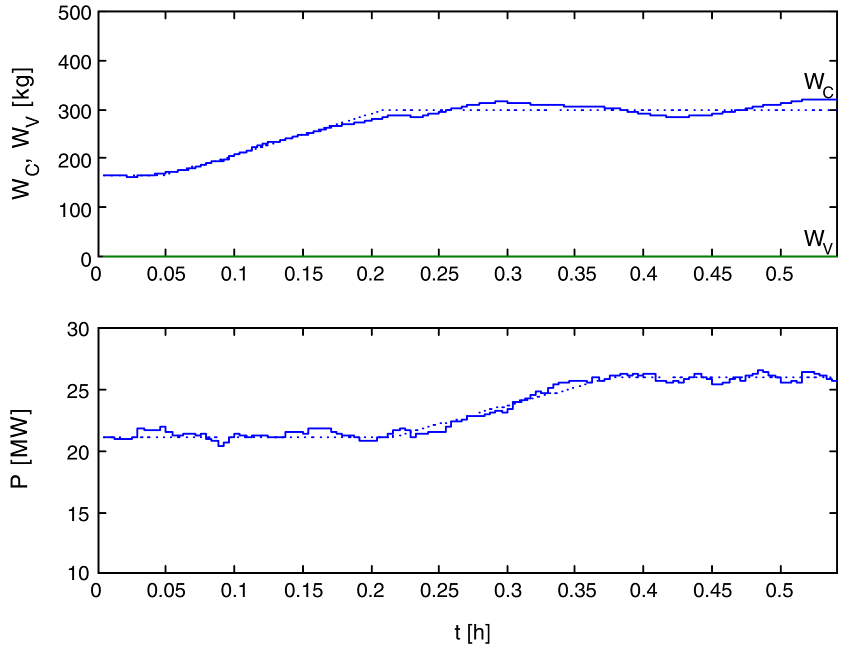


Fig. 4 FBC target trajectories. The *upper plot* shows the bed fuel inventory target trajectory (*dotted line*) and the simulated trajectory (*solid line*). The freeboard inventory is negligible. The *lower plot* shows the desired and simulated superheated steam power (*dotted* and *solid lines*, respectively)

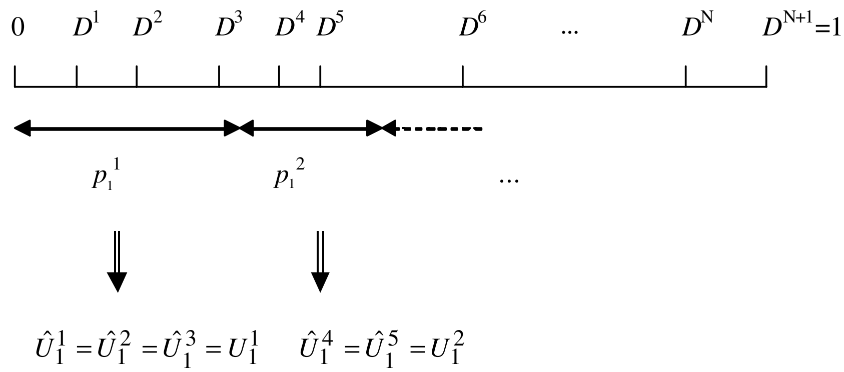


Fig. 5 Selection of the control actions

shown in the figure, we then count the number N_1^j of D^i in each interval of length p_1^j and we affect N_1^j times the value of the control U_1^j.

This selection procedure is based on the facts that if Z is a random variable distributed uniformly on the interval [0,1], then the random variable $-(1/\lambda)\log(1-Z)$ is distributed according to an exponential law with parameter $\lambda > 0$. As $(1-Z)$ is also uniformly distributed on the interval [0, 1], the random variable $-(1/\lambda)\log(Z)$ is also distributed according to the exponential law with parameter λ [1]. If S^1, S^2, ..., S^{N+1} are $(N+1)$ independent random variables distributed according to an exponential law with parameter λ, then the random variable

$$\overline{R}^{N+1} = S^1 + S^2 + \cdots + S^{N+1}, \quad \overline{R}^N - \overline{R}^{N-1} = S^N$$

is distributed according to the Gamma law with parameter $(N+1)$ the form of which is very close to the Poisson law, and if $\{S^N; N \geq 1\}$ are independent random variables distributed according to an exponential law with parameter $\lambda = 1$, then for any $\gamma > 0$ the integer random variable

$$M = \inf\{N \geq 1; S^1 + S^2 + \cdots + S^{N+1} > \gamma\}$$

is distributed according a Poisson law with parameter γ [1]. Notice also that for any $N \geq 1$ the random variable $\overline{R}^{N+1}$ and the normalized vector

$$(D^1, \ldots, D^N) = \left(\frac{\overline{R}^1}{\overline{R}^{N+1}}, \ldots, \frac{\overline{R}^N}{\overline{R}^{N+1}}\right)$$

are independent, and this vector is distributed according to an uniform statistic on the interval [0,1].

8 Appendix B: Mechanization of the algorithm

Let us assume that the plant model is given by

$$\mathbf{x}_{n+1} = f(\mathbf{x}_n, \mathbf{u}_n), \quad \mathbf{y}_n = g(\mathbf{x}_n).$$

The following pseudo-code implements the algorithm:

for $n = 1 : T$

for $i = 1 : N$

if $n == 1$, Initialize $\mathbf{x}^i = \mathbf{x}_0$, $\mathbf{y}^i = g(\mathbf{x}_0)$. Evaluate $J_Y = \left\|\mathbf{y}_n^{\mathrm{ref}} - \mathbf{y}^i\right\|_{\mathbf{B}_n}^2$ and set $J_T^i = 0$; end

Generate random $\mathbf{u}^i \sim N(\mathbf{0}, \mathbf{A}_n)$.

Store action to a list $\mathbf{v}_n^i = \mathbf{u}^i$.

Evaluate $J_U = \|\mathbf{u}^i\|_{\mathbf{A}_n}^2$ and $J_T^i = J_T^i + J_Y + J_U$.

Set weight $p_{\mathrm{un}}^i = \exp\left(\frac{-\beta}{2} J_Y\right)$.

end

For all $i = 1 : N$: Compute resampling probabilities: $p^i = p_{\mathrm{un}}^i / \sum_{i=1}^N p_{\mathrm{un}}^i$.

Resample: For all $i = 1 : N$: Select $\widehat{I}_i = k$ such that $\Pr(k = j) = p^j$. (see Appendix A)

For all $i \in \widehat{I}$: Compute model for next n: $\mathbf{x}^i = f(\mathbf{x}^i, \mathbf{u}^i)$; $\mathbf{y}^i = g(\mathbf{x}^i)$. Evaluate $J_Y = \left\|\mathbf{y}_n^{\mathrm{ref}} - \mathbf{y}^i\right\|_{\mathbf{B}_n}^2$.

Death and birth. For all $j = 1 : N$: Replace $\mathbf{x}^j$, $\mathbf{y}^j$, $\mathbf{v}_n^j$, J_Y^j and J_T^j by $\mathbf{x}^{\widehat{I}_j}$, $\mathbf{y}^{\widehat{I}_j}$, $\mathbf{v}_n^{\widehat{I}_j}$ $J_Y^{\widehat{I}_j}$ and $J_T^{\widehat{I}_j}$.

end

Find $i^* = \arg\min_i \mathrm{J}_T^i$. The solution for the optimal control sequence is $\mathbf{v}^{i^*}$.

9 Appendix C: Model for a fluidized bed combustor

A simple model for FBC boiler furnace can be formulated based on mass and energy balances. The model divides the furnace into two parts: the bed and the freeboard. The control inputs of the system are the fuel feed Q_C [kg/s], and the primary and secondary air flows F_1 and F_2 [Nm^3/s]. Measurable system outputs are the flue gas O_2 content C_F [Nm^3/Nm^3], the bed and the freeboard temperatures T_B and T_F [K], and the power output P [MW].

The model is described by the following differential equations. For the fuel inventory W_C [kg], oxygen concentration C_B [Nm^3/Nm^3], and temperature T_B [K] in bed:

$$\begin{aligned}
\frac{\mathrm{d}W_{\mathrm{C}}(t)}{\mathrm{d}t} &= (1 - V)Q_{\mathrm{C}}(t) - Q_{\mathrm{B}}(t) \\
\frac{\mathrm{d}C_{\mathrm{B}}(t)}{\mathrm{d}t} &= \frac{1}{V_{\mathrm{B}}}[C_1F_1(t) - X_{\mathrm{C}}Q_{\mathrm{B}}(t) - C_{\mathrm{B}}(t)F_1(t)] \\
\frac{\mathrm{d}T_{\mathrm{B}}(t)}{\mathrm{d}t} &= \frac{1}{c_{\mathrm{I}}W_{\mathrm{I}}}\{H_{\mathrm{C}}Q_{\mathrm{B}}(t) - a_{\mathrm{Bt}}A_{\mathrm{Bt}}[T_{\mathrm{B}}(t) - T_{\mathrm{Bt}}] \\
&\quad + c_1F_1(t)T_1 - c_{\mathrm{F}}F_1(t)T_{\mathrm{B}}(t)\}
\end{aligned}$$

Similarly, the freeboard dynamics are given by:

$$\begin{aligned}
\frac{\mathrm{d}W_{\mathrm{V}}(t)}{\mathrm{d}t} &= VQ_{\mathrm{C}}(t) - Q_{\mathrm{F}}(t) - Q_{\mathrm{T}}(t) \\
\frac{\mathrm{d}C_{\mathrm{F}}(t)}{\mathrm{d}t} &= \frac{1}{V_{\mathrm{F}}}\{C_{\mathrm{B}}(t)F_1(t) + C_2F_2(t) \\
&\quad - X_{\mathrm{V}}Q_{\mathrm{F}}(t) - C_{\mathrm{F}}(t)[F_1(t) + F_2(t)]\} \\
\frac{\mathrm{d}T_{\mathrm{F}}(t)}{\mathrm{d}t} &= \frac{1}{c_{\mathrm{F}}V_{\mathrm{F}}}\{H_{\mathrm{V}}Q_{\mathrm{F}}(t) - a_{\mathrm{Ft}}A_{\mathrm{Ft}}[T_{\mathrm{F}}(t) - T_{\mathrm{Ft}}] \\
&\quad + c_{\mathrm{F}}F_1(t)T_{\mathrm{B}}(t) + c_2F_2(t)T_2(t) \\
&\quad - c_1[F_1(t) + F_2(t)]T_{\mathrm{F}}(t)\}
\end{aligned}$$

The plant superheated steam power dynamics can be approximated by

$$\frac{\mathrm{d}P(t)}{\mathrm{d}t} = \frac{1}{\tau_{\mathrm{mix}}}[P_{\mathrm{T}}(t) - P(t)].$$

The combustion rates can be approximated by $Q_{\mathrm{B}}(t) = \frac{W_{\mathrm{C}}(t)}{t_{\mathrm{C}}}\frac{C_{\mathrm{B}}(t)}{C_1}$, $Q_{\mathrm{F}}(t) = \frac{W_V(t)}{t_{\mathrm{V}}}\frac{C_{\mathrm{F}}(t)}{C_2}$, where t_{C} and t_{V} refer to the mean particle combustion time; $Q_{\mathrm{T}}(t) = \frac{W_V(t)}{t_{\mathrm{T}}(t)}$, $t_{\mathrm{T}}(t) = \frac{V_{\mathrm{F}}}{F_1(t)+F_2(t)}$. The heat transfer is given by $P_{\mathrm{T}}(t) = a_{\mathrm{Bt}} A_{\mathrm{Bt}} [T_{\mathrm{B}}(t) - T_{\mathrm{Bt}}] + a_{\mathrm{Ft}} A_{\mathrm{Ft}} [T_{\mathrm{F}}(t) - T_{\mathrm{Ft}}] + c_{\mathrm{F}}(F_1 + F_2)(T_{\mathrm{F}} - T_{\mathrm{stack}})$.

For most of the details on the considered model, the construction of balances and tuning of the model, we refer to [11, 12], including parameters for a 25 MW semi-circulated district heating plant. Note that the above model for combustion in the freeboard and transfer into superheated steam power is slightly more refined, however.

References

1. Najim K, Ikonen E, Ait-Kadi D (2004) Stochastic processes, estimation, optimisation, and analysis. Kogan Page, London
2. Moore KL, Dahleh M, Bhattacharyya SP (1992) Iterative learning control: a survey and new results. J Robotic Syst 9(5):563–594
3. Najim K (2006) Control of continuous linear systems: theory and solved problems. Hermes Science Publisher, London
4. Ramshaw C (1995) In: Ramshaw C (ed) Proceedings of the 1st international conference on process intensification for the chemical industry, BHR Group Publication 18, London
5. Zalzala AMS, Fleming PJ (eds) (1997) Genetic algorithms in engineering. IEE Control Engineer Series, London
6. Del Moral P (1997) Maslov optimisation theory: optimality versus randomness. In: Kolokoltsov VN, Maslov VP (eds) I-dempotency analysis and its applications, vol 401, Translation of Idempotency analysis and its applications in optimal control (Russan), Nauka, Moscow 1994, Kluwer, Boston, Mathematics and its Applications, pp 243–302
7. Ikonen E, Del Moral P, Najim K (2004) A genealogical decision tree solution to optimal control problems. IFAC Workshop AFNC'04, Oulu, Finland
8. Del Moral P (2004) Feynman-Kac formulae genealogical and interacting particle systems with applications. Springer, Berlin Heidelberg New York
9. Honglu Yu (2003) Development of vison-based inferential sensors for process monitoring and control. PhD Thesis, McMaster University. http://www.macc.mcmaster.ca/vision/Research/Publications.htm
10. Davidson PR , Jones RD, Andreae JH, Sirisena HR (2002) Simulating closed- and open-loop voluntary movement: a nonlinear control-systems approach. IEEE Trans Biomed Eng 49:1242–1252
11. Ikonen E, Najim K (2002) Advanced process identification and control. Marcel Dekker, New York
12. Ikonen E, Kortela U (1994) Dynamic model for a bubbling fluidized bed coal combustor. Control Eng Pract 2(6):1001–1006
13. Ikonen E, Najim K, Kortela U (2000) Neuro-fuzzy modelling of power plant flue-gas emissions. Eng Appl Artif Intell 13(6):705–717
14. Ikonen E, Kovacs J (2006) Learning control of fluidized bed combustion processes for power plants. In: Kalogirou S (ed) Artificial intelligence in energy and renewable energy systems. Nova Publishers (in press)
15. Hsu CS (1985) Cell-to-cell mapping—a method of global analysis for nonlinear systems. Springer, Berlin Heidelberg New York
16. Poznyak AS, Najim K, Gomez-Ramirez E (2000) Self-learning control of finite Markov chains. Marcel Dekker, New York