Accurate Dynamic Models for Type 1 Diabetes Identified from Novel Clinical Data

Daniel A. Finan,[1] Signe Schmidt,[2] John Bagterp Jørgensen,[1] Niels Kjølstad Poulsen,[1] Kirsten Nørgaard,[2] and Henrik Madsen[1]

[1]*Department of Informatics and Mathematical Modeling, Technical University of Denmark, Kongens Lyngby, Denmark*
[2]*Department of Endocrinology, Hvidovre University Hospital, Hvidovre, Denmark*

The practical, day-to-day treatment regimen for people with type 1 diabetes (T1DM) entails self-administration of exogenous insulin in order to regulate blood glucose concentrations as close to normal levels as possible. The most efficacious dosing is achieved by measuring, as often as possible, blood glucose concentration through finger-sticks, and delivering insulin accordingly. To manage T1DM properly, therefore, is painful and requires constant decision making. By contrast, an *artificial pancreas* is a biomedical device in development that will automatically regulate blood glucose concentration while freeing patients of the daily burden of self-management.

A critical component of T1DM treatment is dosing insulin so as to offset the carbohydrate content of meals. By current standards, this often involves administration of insulin injections (called boluses) coincident with the meals. Moreover, these factors are commonly taken in a prescribed ratio, known as the insulin-to-carbohydrate ratio.

The control algorithm for an artificial pancreas may well be based on a mathematical model of the patient's glucose-insulin dynamics, as in a model predictive control framework. In such a model-based algorithm, it is advantageous to use an accurate, but simple, model. Linear dynamic models may provide sufficiently accurate predictions, and have other inherent advantages like straightforward, computationally tractable identification, and potential to be re-estimated online, thereby adapting to the evolving dynamics of the patient.

Unfortunately, the simultaneity and uniform proportionality of the meals and the insulin boluses confounds accurate estimation of the model parameters. To avoid this pitfall, we have devised a new in-clinic protocol based on design-of-experiment considerations that yields information-rich data for model identification. The protocol involves separating key factors that influence glucose concentration: meals, insulin boluses, and bouts of exercise.

The novel data give rise to more accurately identified models. A variety of empirical models was identified from the data: difference-equation models like autoregressive exogenous-input (ARX) and autoregressive moving-average exogenous-input (ARMAX) models, transfer-function (TF) models, and state-space models. In addition, "gray" forms of these models were identified which incorporate simple physiological elements such as estimates of subcutaneous-to-intravenous insulin absorption and appearance rate of glucose in the blood from a carbohydrate meal.

The quality of the identified models was based on several measures including:

- Accuracy with which the model was fit to the training data
- Values of parameters and/or combinations of parameters (e.g., steady-state insulin-to-glucose gain)
- Accuracy of predictions for independent (test) data

Identification results for one experiment are shown in Fig. 1. The experiment included three inputs: a meal, an episode of exercise, and a correction bolus, as depicted in the figure. The particular model shown was a transfer function model that was identified from the experimental data. With the exception of the initial, minor glycemic excursion between trial time -60 min and trial time 0 min, the model prediction is very accurate.

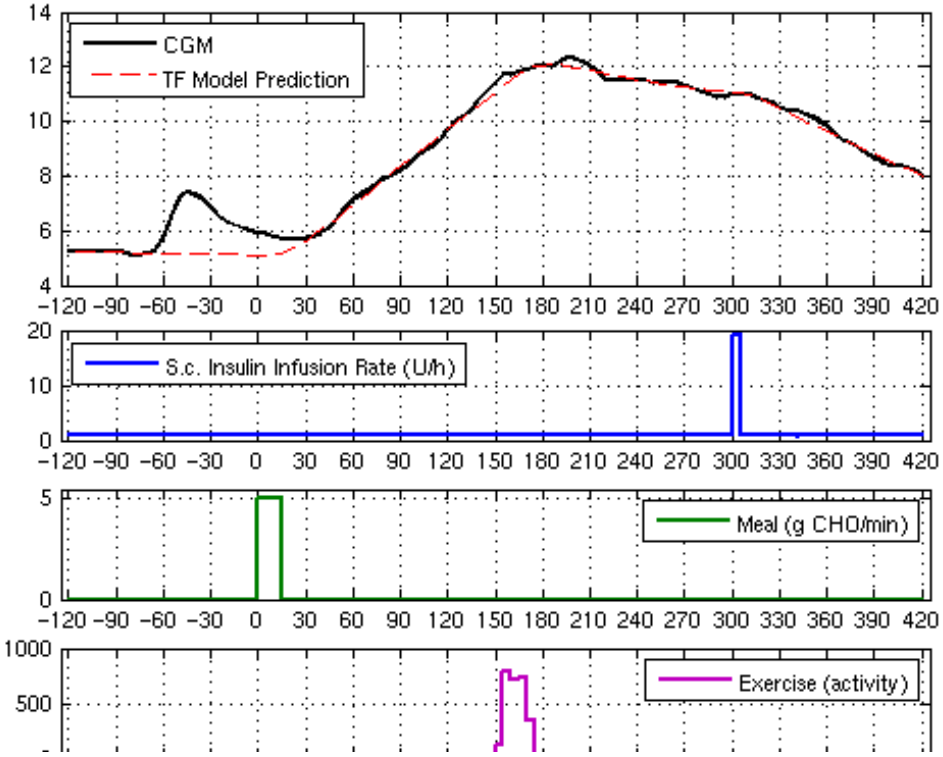


Fig. 1. Identification results for one subject. The TF model prediction is infinite-step. The experimental inputs included an unbolused meal at trial time 0 min, a bout of moderate exercise at trial time 150 min, and a correction bolus at trial time 300 min.

Financial support from the Danish Strategic Research Council is gratefully acknowledged.

Commissioning a Distillation Column Simulator

Ramkrishna Ghosh and Kurt-Erik Häggblom

Process Control Laboratory, Department of Chemical Engineering,
Åbo Akademi University, Åbo, Finland

Abstract

Energy saving has become an extremely important issue in the chemical process industries. Distillation columns, in particular, consume huge amounts of energy. One way of minimizing the energy consumption is improved control, which enables operation closer to certain constraints.

It has been estimated that 75% of the cost associated with an advanced control project typically goes into model development (Gevers, 2005). Hence, efficient modeling and system identification techniques suited for industrial use and tailored for control design applications are needed. This task is especially difficult for "ill-conditioned" MIMO systems such as distillation columns. Even for a linear system, this ill-conditioning makes the system behavior resemble that of a (strongly) nonlinear system. Because of this, identification, modeling and control of ill-conditioned systems are demanding tasks.

In an industrial environment, the identification usually has to be carried out while the plant is in normal operation. It is then essential to keep the variation of inputs and outputs within specified limits and to limit the duration of the identification experiments. However, this also limits the information available for system identification. Thus, there is a trade-off between how much one is prepared to "pay" for the information and the information needed for system identification.

These kinds of identification issues can be investigated by means of a pilot-scale distillation column at Åbo Akademi. However, in order to enable more effective identification studies, a distillation column simulator has been constructed using MathWork's Simulink as programming environment. Because the simulator is to be used in conjunction with the real distillation column, it is desired that the behavior of the simulator is close to that of the real column. A significant number of previously performed identification experiments with the distillation column are available for the tuning of the simulator.

In order facilitate the simulator tuning, we study the effects of column and mixture property parameters appearing in the simulator model on observable static and dynamic properties. In particular, we want to find out which parameters most strongly affect these properties in various input-output relationships. Besides the tuning issue, this information also has more general interest.

Dynamic modeling of combustion in a BioGrate furnace: a sensitivity analysis on the fuel quality and combustion air supply

Alexandre Boriouchkine, Alexey Zakharov, Sirkka-Liisa Jämsä-Jounela

Aalto University, School of Science and Technology, Department of Biotechnology and Chemical Technology, Process Automation Research Group, 00076 Aalto, Finland. e-mail: aboriouc@cc.hut.fi

Abstract: This paper considers dynamic modeling of the bed combustion in a furnace of the BioGrate boiler. The developed dynamic model is heterogeneous, including solid and gas phases. Furthermore, the model considers chemical reactions in both, gas and solid phases. In addition, fuel movement on the grate is included in to the model. The energy required by the process is employed through a radiation function validated by industrial data from a BioGrate boiler at Trolhättan, Sweden. The model is implemented in MATLAB environment and tested with the industrial data. The results are presented and discussed.

1. Introduction

Increasing utilization of renewable energy has created new energy efficiency challenges for industry. Biomass is one of the most important raw material for renewable energy. All the available biomass sources have to be considered for energy production. Fuel properties of biomass vary a lot depending on its origin, on processing and handling for fuel. Variable properties cause fluctuations in combustion and set challenges to develop new combustion control strategies.

One of the latest successful processes developed, which use wood waste as a fuel, is a BioGrate-boiler technology developed by MWBiopower. The combustion of wood waste is, however, a very complex process involving several highly coupled chemical reactions. Furthermore, operational conditions of the furnace greatly affect the yields of chemicals produced during the combustion process, i.e., fractions of tars, gases and char. Moreover, not only the yields of chemicals differ under different combustion conditions, but also their reactivity in succeeding reactions. As a result of such complexity, optimization of a boiler control strategy requires a detailed process model [3].

The most recent publications, considering modeling the combustion of solid fuel on a grate, concentrate on the combustion of either straw or municipal waste. Shin and Choi [21] have developed a 1-D model of waste incineration to understand better phenomena occurring inside a municipal solid waste (MSW) incinerator. Van der Lans et al. [18] have developed a two-dimensional, homogeneous model for design and operation parameter optimization of a straw combustion process. Goh et al. [19] have developed the model of grate combustion of municipal solid waste in order to study the process, since efficient incinerator design requires extensive knowledge of the combustion process. Later, Yang et al. [20] developed a 2-D model of a MSW incinerator which was then verified using experimental data obtained from a pot reactor. Kær [22] developed a one-dimensional model to describe a fixed bed combustion of straw and using walking grate concept extended it to cover the moving bed of a straw boiler. Similarly, Zhou et al. [23] assumed insignificant horizontal temperature gradients in order to simplify the thermal conversion model on the grate of a straw boiler to one dimension.

This paper describes the developed model of a BioGrate furnace. The purpose for the modeling work was to construct a dynamic model providing an insight into the chemical and physical phenomena occurring inside the process. This paper is organized as follows: Section 2 describes the structure of a BioGrate boiler process, Section 3 presents the model and its aspects, Section 4 discusses the implementation details of the model, Section 5 presents the simulation results, Section 6 summarizes the results of simulations.

2. Process description of a BioGrate boiler

A BioGrate consists of following parts: a water filled ash space below the grate, while the grate itself is located above the reservoir. The BioGrate is covered with a heat insulating brick wall, which reflects the heat radiation back to the grate [3].

The grate consists of several ring zones. These zones are further divided into two types of rings: rotating and fixed. A half of the grate rings are rotating and the rest are fixed. Every second rotating ring rotates clockwise and the others rotate counterclockwise. This structure helps spreading fuel evenly upon the surface of the conical grate [3].

Fuel is fed into the center of the grate from below. In the middle of the cone the fuel dries as a result of heat radiation, which is emitted by the combusting flue gas and reflected back to the grate by the grate walls. The dry fuel then proceeds to the outer shell of the grate, where pyrolysis and char combustion occur. The ash and carbon residues fall off the edge of the grate into the water-filled ash pit [3].

The air required in combustion is fed into the grate from the bottom of the grate (primary air) and from the grate walls (secondary air). In addition, in order to ensure clean combustion, additional air can be fed from the top of the grate (tertiary air). Burning produces heat that is absorbed in several steps. First, the evaporator absorbs the energy in the flue gases. Next, part of the energy of the flue gases is transferred to superheaters. In the third phase, the heat is transferred to the convective evaporator. Finally, economizers remove the remaining flue-gas energy [3].

The operation principle of a power plant is based on the steam generation. As any other bio power plant, a BioGrate power plant comprises several parts, including a boiler, a turbine generator, a feed-water tank, a water treatment plant and a flue gas-cleaning system. Solid fuel is fed into the furnace of the boiler, where it is combusted, generating heat and flue gases. Flue gases contain fly ash

which comprises several harmful components. Therefore, flue gases are purified of fly ash before the release into the atmosphere. Therefore, the flue gases are subjected to several steps of the cleaning procedure and then emitted into the atmosphere. Instead, the heat acquired from the fuel is used for the steam production.

The steam produced in the boiler is led to a generator turbine, which converts its mechanical energy into electricity. As steam performs mechanical work, its pressure decreases, steam with decreased pressure is then used for heating utility streams, such as water [2]. After steam has released enough energy it condenses. The condensed steam is called condensate which along with pretreated feed water is fed into a feed-water tank. Inside the tank, liquid is heated with a bled steam from the turbine. This procedure increases energetic efficiency of the process [1].

3. Dynamic modeling

The current model of a BioGrate uses walking grate concept modified for a BioGrate furnace. In addition, the chemical reaction kinetics were selected, especially, to fit the operational conditions of the BioGrate. Furthermore, an experimental model was used to model the radiation distribution inside the furnace.

Biomass bed reacts in a series of four different chemical reactions: drying, pyrolysis, char gasification and char combustion [5]. Active drying starts when temperature of a particle reaches the boiling point of water. Then, the high temperature of a furnace initiates a pyrolysis reaction. The pyrolysis reaction produces three products: gases, char and tar. Gases are mainly composed of *CO*, CO_2, H_2, and C_1-C_3 hydrocarbons. Tar contains many organic components, such as levoglucosan, furfural, furan derivatives and phenolic compounds [6]. Next, each reaction will be discussed in details.

3.1. Continuity equation models, their parameters and assumptions

3.1.1. Assumptions

Several assumptions are made in order to simplify the modeling work. The assumptions are listed in descending order of importance:

1. The system is one dimensional, because the length of the grate is significantly longer than the height. Therefore, the temperature gradient in the horizontal direction is insignificant compared that in the vertical direction.
2. Plug-flow gas assumption [9]. The gas phase is assumed to be ideal [9], [10].
3. The solid is assumed to be a porous material [11].
4. Diffusion in the gas phase is neglected, since the effect of convection on transportation of the gas is significantly greater [5].
5. Pressure dynamics are ignored, because the release of gaseous species is negligible compared to the primary air flow and, as a result, pressure evolution can be neglected [9]
6. Heat produced in char combustion is assumed to be retained in the solid phase [9].
7. No volume reduction (shrinkage) occurs during drying, pyrolysis and combustion [9], [7]
8. The temperature of the gas released from solids is the same as that of the solids [9]
9. The temperature of solids in a discretized block is uniform [9]
10. The heat capacity of the wood is assumed to be constant [9]
11. No heat loss

Next, the simplified continuity equations are presented.

3.1.2. Solid Phase continuity equation

Solid phase reacts through drying, pyrolysis and char combustion reactions:

$$\frac{\partial \rho_s}{\partial t} = -R_f \tag{1}$$

where ρ_s is the density of the solid phase, and R_f the overall reaction rate of the solids.

3.1.3. Energy continuity equation of the solid phase

Energy equation for the solid phase considers heat conduction, heat exchange between phases, energy lost in drying and pyrolysis reactions, and energy gained in char combustion:

$$\begin{aligned}&\frac{\partial T_s}{\partial t} C_s \rho_s = \frac{\partial}{\partial x}\left(k_{cond} \frac{\partial T_s}{\partial x} \right) + k_{conv} v_p (T_f - T_s) - \dots \\ &R_{evap}\Delta H_{evap} - R_{pyr}\Delta H_{pyr} + R_{comb,C}\Delta H_{comb,C} - \dots \\ &R_{gasi,CO2}\Delta H_{gasi,CO2} - R_{gasi,H2O}\Delta H_{gasi,H2O}\end{aligned} \tag{2}$$

where T_s is the temperature of solid phase, C_s the heat capacity of the solid phase, ρ_s the density of the solid phase, x the vertical coordinate, k_{cond} the heat conduction coefficient of the solid phase, k_{conv} the heat convection coefficient between the gas and solid phases, v_p is the density number, T_f the temperature of the gas phase, and R_{evap} and R_{pyr} the reaction rates of drying, pyrolysis. Reaction rates $R_{comb,C}$, $R_{gasi,CO2}$ and $R_{gasi,H2O}$ correspond to reaction rates of char combustion, gasification with carbon dioxide and gasification with water steam, respectively. ΔH_{evap} and ΔH_{pyr} are the reaction enthalpies of drying, pyrolysis. Reaction enthalpies $\Delta H_{comb,C}$, $\Delta H_{gasi,CO2}$ and $\Delta H_{gasi,H2O}$ correspond to reaction enthalpies of char combustion, gasification with carbon dioxide and gasification with water steam, respectively.

The radiation reflected from the grate walls to the fuel bed is described through boundary conditions. The boundary conditions are defined as follows:

At the surface of the fuel bed, $x = a$

$$k_{cond} \left. \frac{\partial T_s}{\partial x} \right|_{x=a} = I_{in} - e\sigma T_s^4 \tag{3}$$

where I_{in} is the energy flux into the system, and $e\sigma T_s^4$ the energy flux out of the system.

To describe the energy flux, I_{in}, an experimental model was used. The model was defined from the experimental data of a BioGrate boiler located in Trolhättan, Sweden.

$$k_{cond}\left.\frac{\partial T_s}{\partial x}\right|_{x=0} = e\sigma T_s^4 \quad (4)$$

where $e\sigma T_s^4$ is the energy flux out of the system.

According to Yagi and Kunii [15], flowing fluid improves the heat conduction of the bed due to effect of axial dispersion; therefore. the overall heat conduction coefficient will become:

$$k_{cond} = k_{cond,0} + \alpha\beta \cdot \mathrm{Re}\,\mathrm{Pr} \quad (5)$$

where $k_{cond,\ 0}$ is the heat conduction of the bed with a stagnant fluid, and the right hand term represents the heat conduction due to axial dispersion. Re is the Reynolds number, Pr the Prandatl number, and $\alpha\beta$ is a geometrical constant for cylinders and spheres. The geometrical constant is reported typically to take values of between 0.1 and 0.13 [15]. In the model, the average $\alpha\beta$ = 0.115 was used.

Heat conduction coefficient from the study of Yagi and Kunii [15] was used to describe heat conduction in the bed while heat conduction coefficient for wood particles was based on the study [14].

3.1.4. Gas phase continuity equation

Reacted solid components of wood are transferred to gas phase, in addition, gas phase continuity equation considers gas flow:

$$\frac{\partial}{\partial t}(\rho_f \varepsilon_b Y_i) - \frac{\partial}{\partial x}(v_f \rho_f \varepsilon_b Y_i) = R_i \quad (6)$$

where ρ_f is the density of gas phase, ε_b the bed porosity, Y_i the mass fraction of the gaseous component i, v_f the gas flow velocity and R_i the rate of formation of gaseous component i.

3.1.5. Energy continuity equation of the gas phase

Assuming no heat loss will occur, the energy continuity equation can be denoted as follows:

$$\frac{\partial h_f}{\partial t}\rho_f = -\frac{\partial}{\partial x}\left(\varepsilon_b v_f h_f\right) - k_{conv} v_p (T_f - T_s) + \ldots \quad (7)$$
$$R_{comb,CO}\Delta H_{comb,CO} + R_{comb,H2}\Delta H_{comb,H2}$$

where h_f is an enthalpy of the gas phase, ρ_f the density of the gas phase, ε_b the bed porosity, v_f the gas flow velocity, and R_i the rate of formation of gaseous component i, k_{conv} is the heat convection coefficient between the gas and solid phases, v_p is the density number, T_f the temperature of the gas phase and T_s is the temperature of the solid phase.

3.2 Chemical reactions of the model

The thermal decomposition of wood comprises three main chemical reactions: drying, pyrolysis and char gasification with char combustion. In general, the chemical reactions can be depicted using experimental or semi-experimental models. However, since Arrhenius dependence equations are simple to use, and also accurate; therefore, they have been used in this work.

3.2.1 Moisture evaporation

Usually, fuels used in combustion processes contain moisture. Depending on the type of fuel, a fuel particle can contain various amounts of moisture. According to Thunman et al. [4], fuel particles can contain up to 60 wt% of moisture while char residue being as low as 10 wt% of the wet wood. Water can be bound to the structure of a wood particle or reside in its pores.

Di Blasi et al. [12] presented a simple, yet accurate model to describe drying kinetics in the updraft gasifiers, which use countercurrent combustion conditions:

$$R_{evap} = 5.6 \cdot 10^8 \exp\left(-88[kJ/mol]/\Re/T_s\right)\rho_{Water} \quad (8)$$

where ρ_{Water} is the density of water, T_s the temperature of the solid phase, and $\Re$ the gas constant.

3.2.2 Pyrolysis

After a particle has dried, the next reaction occurring is pyrolysis. In the pyrolysis reaction, a dry wood particle is decomposed into tar, volatile organic components and char. However, fractions of tar, gas and char in the product yield are strongly dependent on the reaction conditions of a combustion process.

Alves and Figueiredo [13] presented a mathematical model of wet wood pyrolysis. The simulation results of this model were experimentally validated in temperature range of 298 – 780 °C with a wet pine cylinder having the radius of 18.5 mm, with water content being 45-49 wt-%. Experimental and simulated results agreed.

Cellulose is reported to react with the following reaction kinetics [13]:

$$R_{devol,\ cel} = 2 \cdot 10^9 \cdot \exp(-146\,[kJ/mol]/\Re/T_s) \cdot \rho_{cel} \quad (9)$$

where ρ_{cel} is the density of cellulose, T_s the temperature of the solid phase, and $\Re$ the gas constant.

Hemicellulose is reported to react with the following rection kinetics [13]:

$$R_{devol,\ Hemi} = 7 \cdot 10^4 \cdot \exp(-83\,[kJ/mol]/\Re/T_s) \cdot \rho_{Hemi} \quad (10)$$

where ρ_{Hemi} is the density of hemicellulose, T_s the the temperature of the solid phase, and $\Re$ the gas constant.

3.2.3 Combustion of pyrolysis gases

The yield of pyrolytic gases is around 85 wt. % under the operation conditions of a BioGrate boiler, since under these conditions the gasifying pyrolysis is the dominant pyrolysis mode. Therefore, significant amount of energy, used by the boiler, comes from the combustion of gases; this fact poses the combustion of pyrolytic gases as the most important energy source. However, the composition of the gaseous products of pyrolysis reported in the study

of Dupont et al. [25], suggests that carbon monoxide has the highest concentration in the pyrolytic gas, while the fraction of other combustible gases remains under 10 wt. %. Therefore, in order to ensure the acceptable accuracy of the model, while keeping the model simple, only the oxidation of carbon monoxide to carbon dioxide is considered.

In addition to the oxidation of carbon monoxide, also the combustion hydrogen is included in the model because of three facts. First, char is known to react with water steam producing carbon monoxide and hydrogen. Second, significant amount of water steam is released during the drying reaction. Consequently, also the amount of produced hydrogen in char gasification reaction can become significant. Finally, hydrogen reacts rapidly in the presence of oxygen, producing significant amounts of energy.

In the presence of water steam and oxygen, carbon monoxide is known to follow the following kinetics [26]:

$$R_{comb,CO} = 1.3 \cdot 10^{14} \exp(-30[kcal/mole]/\Re/T_g) \cdot ... \\ \rho_{CO} \cdot \rho_{O2}^{0.5} \cdot \rho_{H2O}^{0.5} \quad (11)$$

where ρ_{CO} is the density of carbon monoxide, ρ_{O2} is the density of oxygen, ρ_{H2O} is the density of water steam, T_g is the temperature of the solid phase, and $\Re$ the gas constant.

Hydrogen is reported to react through the following kinetics [26]:

$$R_{comb,H2O} = 2.96 \cdot 10^{11} \exp(-6900[K]/T_g) \cdot ... \\ \rho_{H2} \cdot \rho_{O2}^{0.92} \quad (12)$$

where ρ_{H2} is the density of hydrogen, ρ_{O2} is the density of oxygen, T_g is the temperature of the solid phase, and $\Re$ the gas constant.

3.2.4 Char conversion reactions

Char combustion in the model is accounted for with the model presented in Janse et al. [24]. This model is chosen because it is valid over the temperature range 573-773K, which corresponds to the temperature of char combustion in BioGrate. In addition, the pyrolysis conditions of char particles, which were used to obtain the model parameters, are similar to the pyrolysis conditions inside a BioGrate.

$$R_{comb} = 5.3 \cdot 10^{5} \exp(-125[kJ/mole]/(\Re T_s)) \cdot ... \\ P_{O2}^{0.53} (1-X)^{0.49} \rho_C \quad (13)$$

where ρ_c is the density of the char, T_s is the temperature, of the solid phase, $\Re$ is the gas constant, and P_{O2} the pressure of oxygen, and X is the degree of the conversion of char.

Matsumoto et al. [27] have reported, in their study, that the random pore rate equations is the best option to describe the gasification reactions of char with carbon dioxide and water steam. Therefore, the rate equations presented in the study of Matsumoto et al. [27] for char gasification with carbon dioxide and water steam are used in the model.

Gasification reaction of char with carbon dioxide [27]:

$$R_{gasi,CO2} = P_{CO2}^{0.22} \cdot 2.24 \cdot 10^{3} \exp(-93.9[kJ/mole]/\Re/T_s) \\ \cdot (1-X)\sqrt{1-10\ln(1-X)} \cdot \rho_C \quad (14)$$

where ρ_c is the density of the char, T_s the temperature, of the solid phase, $\Re$ is the gas constant, and P_{CO2} the pressure of carbon dioxide, and X is the degree of the conversion of char.

Gasification reaction of char with water steam [27]:

$$R_{gasi,H2O} = P_{H2O}^{0.22} \cdot 9.99 \cdot 10^{4} \cdot ... \\ \exp(-136[kJ/mole]/\Re/T_s) \cdot ... \\ \cdot (1-X)\sqrt{1-10\ln(1-X)} \cdot \rho_C \quad (15)$$

where ρ_c is the density of the char, T_s the temperature of the solid phase, $\Re$ is the gas constant, and P_{H2O} the pressure of carbon dioxide, and X is the degree of the conversion of char.

4 Implementation of the models and the description of the testing environment

The model was implemented in the MATLAB environment, in which a set of finite difference methods was used to solve the continuity equations. The overall solving algorithm is presented in Figure 1.

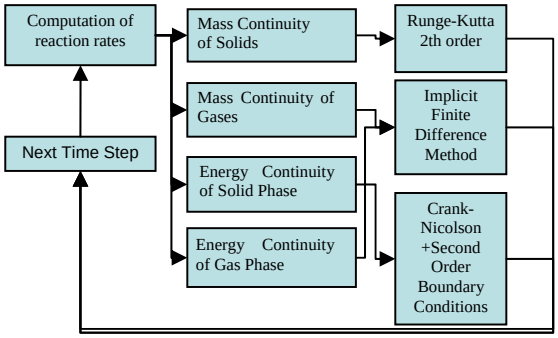


Figure 1. Model solving scheme.

5 Simulation Results

This section presents simulation results obtained with different fuel parameter values.

5.1 Simulation case I, studying the effect of fuel quality on the combustion of wood chips

Wood chips possess several quality properties, including the fuel bed porosity, the bed density and the moisture content. However, the moisture content is the most important property, since it varies significantly between different batches of the fuel. In contrast to the moisture content, the bed porosity and bed density, in case of wood chips, vary only insignificantly; therefore, they are of no interest in the current study. The simulation results with moisture contents of 40, 50 and 60 wt. % are shown in Figure 2.

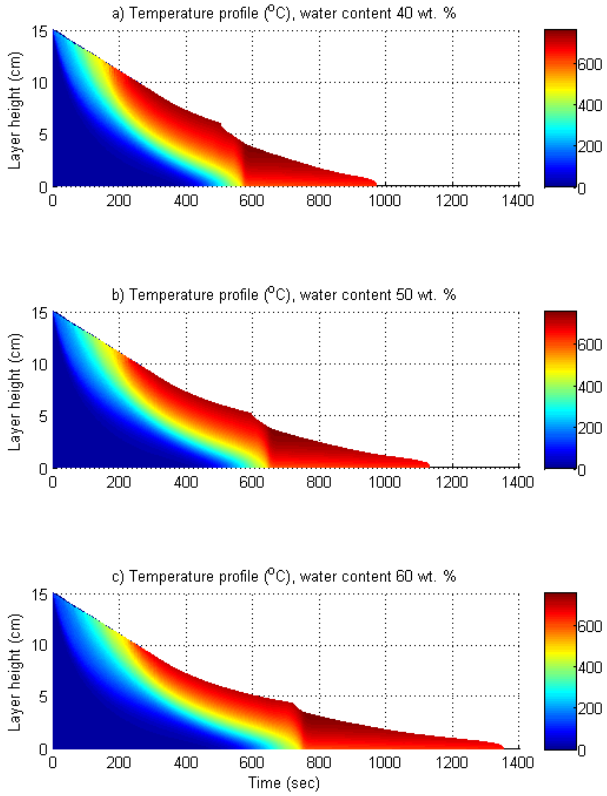


Figure 2. Temperature profiles of the fuel bed with fuel with a) 40 wt.% b) 50 wt. % and c) 60 wt. % moisture content

The results presented in Figure 2 suggest that, although the temperature of the fuel beds with a different moisture content remains within the same range, there are considerable differences in the combustion process. Not only the combustion time increases with increasing moisture content, but also the reaction front becomes narrower. The increase in combustion time can be explained by the fact that the higher the moisture content the longer time it takes for the water to evaporate. Furthermore, high moisture content prevents the heat flux from penetrating deeper into the fuel bed, because a larger amount of heat is required to dry the same amount of fuel. Similar behavior was also observed in the study of Yang et al. [28]. In the study [28], it was found that burning rate is inversely proportional to the moisture content of a fuel bed.

5.2 Simulation case II, studying the effect of the air flow on the combustion of wood chips

The air flow also has a significant effect on the combustion process, since it provides the oxygen required for chemical reactions, such as char oxidation. Simulations were conducted with the air flows of 0.5, 0.75, 1, 2 and 5 m^3/s, while particle sizes were 20, 35 and 50 mm. Table 1 presents the result obtained from the simulation, while Figure 3 visualizes the results presented in Table 1.

Table 1. Combustion time of fuels with different air flows.

Air flow m^3/s	Combustion time (s)		
	20 mm	35 mm	50 mm
0.5	1621	1590	1558
0.75	1398	1357	1336
1	1379	1320	1286
2	1373	1279	1230
5	1435	1269	1202

Figure 3 shows that, as the volume of the air flow increases from 0.5 to 0.75 m^3/s, the combustion times for all particle sizes decrease significantly from around 1600 s at 0.5 m^3/s, to 1350 – 1400 second at 1 m^3/s.

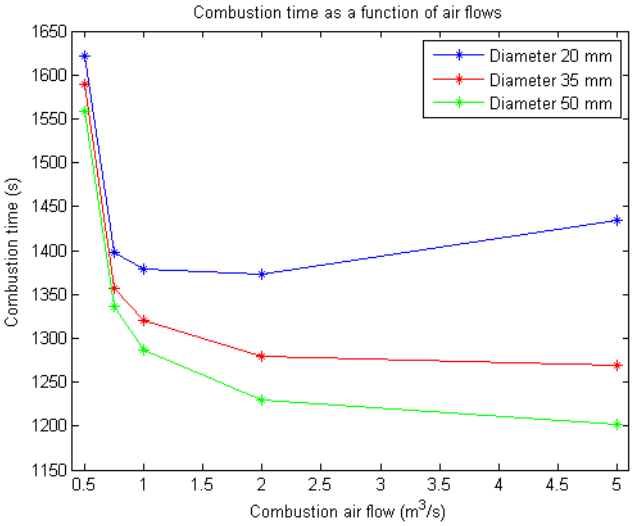


Figure 3. Combustion times as a function of the volumetric air flow.

This phenomenon is the result of oxygen deficiency at low air supplies in the combustion process, i.e., not enough oxygen is supplied to the process to burn the char at its maximum rate.

Figure 3 indicates that, indeed, with low air flows char burns slowly. However, as shown in Figure 3 also high air flows can slow down the combustion of the fuel. This phenomenon can be explained by the cooling property of the air flow. The air supplied to the process has a significantly lower temperature, especially, at the combustion front where the temperature of the solid phase is high. In addition to the significant temperature difference, air is supplied in an opposite direction to the reaction front, thus making heat conduction less efficient and narrowing the reaction front. This finding is also supported by the study of Thunman and Leckner [8]. Figure 4 shows the difference in reaction front thicknesses between airflows of 0.5 and 5 m^3/s.

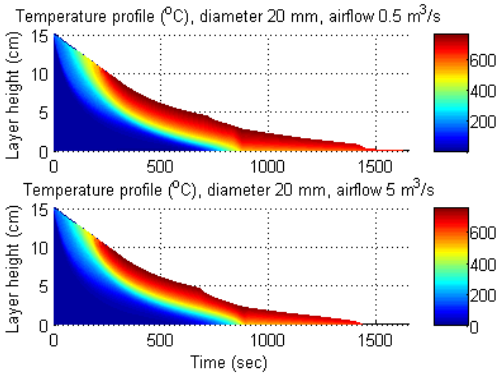


Figure 4. Temperature profiles of beds with 0.5 and 5 m^3/s airflows

Nevertheless, the cooling effect is not significant with large fuel particle diameters, compared to that of 20 mm particles. Figure 3 indicates that in case of 20 mm large particles the combustion time decreases while air flow is increased from 0.5 to 0.75 m^3/s. However, when the air flow is further increased above 2 m^3/s, the combustion

time starts decreasing. In contrast to 20 mm large particles, larger particles seem to be less affected by the cooling of the air flow. Furthermore, the combustion time continues decreasing as the air flow is increased above 2 m^3/s. These findings can be explained by the fact that for smaller particle sizes the density number, the ratio of particle area to unit volume, is larger than that of large particles. In addition, for small particle sizes, also the heat convection coefficient, which is responsible for heat exchange between the gas and solid phases, has a larger value than the coefficient of large particles. Therefore, in case of small particles the heat exchange between phases is more efficient, compared to large particles, thus the cooling property of the air flow affects small particles more than the larger ones. Horttanainen et al. [29] have concluded in their study that combustion air flow rate can be increased as particle size is increased. This finding is in agreement with the result obtained from simulation case II.

6 Conclusions

The investigation on the boiler furnace model was started by studying phenomena occurring in the BioGrate furnace. The furnace model was decomposed in to reaction rates and governing equations for mass and energy conservation.

The developed BioGrate model was than used to study the process phenomena occurring inside the BioGrate furnace with varying process conditions. In addition, a sensitivity analysis was made for different parameters, which affect the combustion process.

The sensitivity analysis showed that the combustion time increased linearly with the increase of moisture content. A study on the air flow effect indicated that oxygen deficiency slowed down the combustion process, however, excess air, on the other hand, increased the combustion time by cooling the solid phase. The results obtained from the simulator were found to be in agreement with the results found in literature.

7 References

1 Kiameh, P., Power Generation Handbook, McGraw-Hill, 2002, 557 p.

2 Joronen, T., Kovács, J., Majanne, Y., *Voimalaitosautomaatio*, Suomenautomaatioseura Oy, Helsinki 2007, 1st edition, 276 p.

3 WWW, Anon, Wärtsilä Oyj, Brochure,

http://service.wartsila.com/Wartsila/global/docs/en/power/media_publications/brochures/bioenergy_fi.pdf, 03.08.2009

4 Thunman, H., Davidsson, K., Leckner, B., *Separation of drying and devolatilization during conversion of solid fuels*, Combustion and Flame **137** (2004), pp. 242 - 250

5 Peters, B., Bruch, C., *A flexible and stable numerical method for simulating the thermal decomposition of wood particles*, Chemosphere **42** (2001), pp. 481 – 490

6 Di Blasi, C., Heat, *Momentum and Mass Transport Through a Shrinking Biomass Particle Exposed to Thermal Radiation*, Chemical Engineering Science **5** (1996), pp. 1121-1132

7 Di Blasi, C., *Combustion and gasification rates of lignocellulosic chars*, Progress in Energy and Combustion Science **35** (2009), pp. 121 – 140

8 Thunman, H., Leckner, B., Ignition and propagation of a reaction front in cross-current bed combustion of wet biofuels, Fuel **80** (2001), pp. 473-481

9 Zhou H, Jensen, A. D., Glaborg, P., Jensen, P., A., Kavaliauskas, A., *Numerical modeling of straw combustion in a fixed bed*, Fuel **84** (2005), pp. 389-403

10 Kær S., K., *Straw combustion on slow moving grates-a comparison of model predictions with experimental data*, Biomass and Bioenergy **28** (2005), pp. 307-320

11 Yang, Y., B., Yamauchi, H., Nasserzadeh, V., Swithenbank, J., *Effects of fuel devolatilization on the combustion of wood chips and incineration of simulation municipal solid wastes in a packed bed*, Fuel **82** (2003), pp. 2205-2221

12 Di Blasi, C., Branca, C., Sparano, La Mantia, B., *Drying Characteristics of wood cylinders for conditions pertinent to fixed-bed countercurrent gasification*, Biomass and Bioenergy **25** (2003), pp. 45-58

13 Alves, S., S., Figueiredo, J., L., *A model for pyrolysis of wet wood*, Chemical Engineering Science **44** (1989), pp. 2861-2869

14 Janssens, M., Douglas, B., *Wood and Wood Products*, Handbook of Building Materials for Fire Protection, Edited by Harper, C., A., McGraw-Hill 2004, 542 p.

15 Yagi, S., Kunii, D., *Studies on Effective Thermal Conductivities in Packed Beds*, A.I.Ch.E. Journal **3** (1957), pp. 373-381

16 Fjellerup, F., Henriksen, U., *Heat Transfer in a Fixed Bed of Straw Char*, Energy & Fuels **17** (2003), pp. 1251-1258

17 Horttanainen M., Saastamoinen, J., Sarkomaa, P., *Operational Limits of Ignition Front Propagation against Airflow in Packed Beds of Different Wood Fuels*, Energy & Fuels **16** (2002), pp. 676-686

18 R.P. van der Lans, L. T. Pedersen, A. Jensen, P. Glarborg and K. Dam-Johansen, *Modelling and expirements of straw combustion in a grate furnace*, Biomass and Bioenergy **19** (2000) , pp. 199-208

19 Y. R. Goh, Y. B. Yang, R. Zakaria, R. G. Siddall, V. Nasserzadeh, J. Swithenbank, *Development of an Incinerator Bed Model for Municipal Solid Waste Incineration*, Combustion Science and Technology **162** (2001), pp. 37-58

20 Yang, Y., B., Goh,, Y.,R., Zakaria, R., Nasserzadeh, V., Swithenbank, J., *Mathematical modelling of MSW incineration on a traveling bed*, Waste Management 22 (2002), pp. 369-380

21 Shin, D, Choi, S, *The Combustion of Simulated Waste Particles in a Fixed Bed*, Combustion and Flame **121** (2000), pp. 167-180

22 Kær S., K., *Straw combustion on slow moving grates-a comparison of model predictions with experimental data*, Biomass and Bioenergy **28** (2005), pp. 307-320

23 Zhou H, Jensen, A. D., Glaborg, P., Jensen, P., A., Kavaliauskas, A., *Numerical modeling of straw combustion in a fixed bed*, Fuel **84** (2005), pp. 389-403

24. Janse, A., M., C., de Jonge, H., G., Prins, W., van Swaaij, W., P., M., Combustion Kinetics of Char Obtained by Flash Pyrolysis of Pine Wood, Ind. Eng. Chem. Res. 37 (1998), pp. 3909-3918

25. Dupont, C., Chen, Li., Cances, J., Commandre, J.-M., Cuoci, A., Pierucci, S., Ranzi, E., *Biomass pyrolysis: Kinetic modeling and expiremental validation under high temperature and flash heating rate conditions*, Journal of Analytical and Applied Pyrolysis **85** (2009), pp. 260-267

26. Babushik, V., I., Dakdancha, A., N., *Global kinetic parameters for high-temperature gas-phase reactions*, Combustion, Explosion, and Shock Waves **29** (1993), pp. 464-489

27. Matsumoto, K., Takeno, K., Ichinose, T., Ogi, T., Nakanishi, M., *Gasification reaction kinetics on biomass char obtained as a by-product of gasification in an entrained-flow gasifier with steam and oxygen at 900 – 1000 °C*, Fuel **88** (2009), pp. 519 – 527

28 Yang, Y., B., Sharifi, V., N., Swithenbank J., *Effect of air flow rate and fuel moisture on the burning behavioursof biomass and simulated municipal solid wastes in packed beds*, Fuel **83** (2004), pp. 1553-1562

29. Horttanainen M., Saastamoinen, J., Sarkomaa, P., *Operational Limits of Ignition Front Propagation against Airflow in Packed Beds of Different Wood Fuels*, Energy & Fuels **16** (2002), pp. 676-686

Data-Based Uncertainty Modeling of MIMO Systems

Hamed Jafarian and Kurt-Erik Häggblom

Process Control Laboratory, Department of Chemical Engineering,
Åbo Akademi University, Åbo, Finland

Abstract

Many robust control design methods require a linear model consisting of nominal model augmented by an uncertainty description. A general form for such a model is

$$G = G_0 + H_{21}\Delta(I - H_{11}\Delta)^{-1}H_{12} \tag{1}$$

where G is the transfer function of the true system, G_0 is a nominal model, and Δ is a perturbation causing uncertainty about the true system. Depending on the particular type of uncertainty (additive, input or output multiplicative, inverse types of uncertainty, combinations of various types of uncertainty), H_{11}, H_{12} and H_{21} can contain combinations of (known) constant matrices and the nominal model.

Assume that we have information about the true system in the form of a number of possible transfer functions G_k, $k = 1,\ldots,N$. The nominal model and the perturbation Δ_k associated with G_k are unknown, but they have to satisfy

$$G_k = G_0 + H_{21}\Delta_k(I - H_{11}\Delta_k)^{-1}H_{12}\ ,\ \ k = 1,\ldots,N\,. \tag{2}$$

How should G_0 be determined? It has been shown that $\|\Delta\|_\infty$ is a control relevant measure of the distance between G and G_0 for models of the form (1) and that the achievable stability margin by feedback control is inversely proportional to this distance. For a given type of uncertainty model, this suggests that G_0 should be determined by solving the optimization problem

$$\min_{G_0} \max_{k} \|\Delta_k\|_\infty \tag{3}$$

subject to the appropriate data matching condition (2). Obviously, the type of uncertainty model giving the smallest minimum is the best one according to this measure.

If information about the system is obtain through identification, input-output data are available. An attractive way of removing noise from the output is to fit a model G_k to the data and to calculate a noise-free output y_k by $y_k = G_k u_k$, where u_k is the input in experiment k. Since the purpose of the experiments in this context is to excite the system in various ways, the inputs do not tend to be persistently exciting in all individual experiments. Thus, G_k only applies to the particular input u_k, and the relevant information is input-output data $\{u_k, y_k\}$, $k = 1,\ldots,N$. This means that the model matching condition (2) should be replaced by the input-output matching condition

$$y_k = G_0 u_k + H_{21}\Delta_k(I - H_{11}\Delta_k)^{-1}H_{12}u_k\,. \tag{4}$$

It can be shown that the use of (4) instead of (2) results in a less conservative uncertainty model.

Our modeling approach is to model G_0 in the frequency domain using sampled frequency responses of the input-output data. Because of the availability of G_k, these are easy to calculate for standardized inputs. The available information is thus $\{u_k(\mathrm{j}\omega), y_k(\mathrm{j}\omega), \omega \in \Omega\}$, $k = 1, \ldots, N$, and we solve the optimization problem frequency-by-frequency, i.e.

$$\min_{G_0(\mathrm{j}\omega)} \max_k \left\| \Delta_k(\mathrm{j}\omega) \right\|_2 \quad \text{s.t. (4), } \forall \omega \in \Omega. \tag{5}$$

The uncertainty Δ_k is assumed to be unstructured.

For some types of uncertainty, the optimization problem can easily be formulated as a convex optimization problem. Additive uncertainty, for example, which in its basic form is described by

$$y_k = G_0 u_k + \Delta_k u_k \ , \ k = 1, \ldots, N \tag{6}$$

results in the optimization problem

$$\min_{G_0} \gamma \quad \text{s.t.} \begin{bmatrix} \gamma I & y_k - G_0 u_k \\ (y_k - G_0 u_k)^* & u_k^* u_k \end{bmatrix} \succcurlyeq 0 \ , \ k = 1, \ldots, N \ , \ \forall \omega \in \Omega \tag{7}$$

which is a convex optimization problem. Here A^* denotes the complex conjugate transpose of A and $P \succcurlyeq 0$ denotes that P is positive semidefinite.

Many types of uncertainty descriptions do not readily give a convex optimization problem. For example, an output multiplicative uncertainty described by

$$y_k = G_0 u_k + \Delta_k G_0 u_k \ , \ k = 1, \ldots, N \tag{8}$$

results in the optimization problem

$$\min_{G_0} \gamma \quad \text{s.t.} \begin{bmatrix} \gamma I & y_k - G_0 u_k \\ (y_k - G_0 u_k)^* & u_k^* G_0^* G_0 u_k \end{bmatrix} \succcurlyeq 0 \ , \ k = 1, \ldots, N \ , \ \forall \omega \in \Omega \tag{9}$$

which is non-convex due to the appearance of $G_0^* G_0$. An iterative solution by keeping $G_0^* G_0$ fixed during each iteration tends to produce local minima which are non-global. However, we show how this optimization problem, and similar ones for some other types of uncertainty, can be reformulated as a convex optimization problem.

For control design, G_0 is needed as a transfer function or a state-space model. In principle, we should determine such a model by replacing G_0 in the appropriate consistency relations, like those appearing in (7) and (9), by a suitable parameterization of G_0. However, so far we have not been able to obtain a satisfactory solution in that way. Instead, we have fitted a model to the calculated frequency responses $G_0(\mathrm{j}\omega)$, $\omega \in \Omega$. A drawback of this approach is that $\min \left\| \Delta(\mathrm{j}\omega) \right\|_2$ will increase, usually also $\min \left\| \Delta \right\|_\infty$, sometimes even drastically. We have studied various approaches of reducing the effects of this drawback.

Modeling for Control of Beer Quality

D.R. Warnasooriya[1], P.G. Rathnasiri[1], Bernt Lie[2]

[1]*Department of Chemical and Process Engineering, University of Moratuwa, Kattubedda, Moratuwa, Sri Lanka.*

E-Mail; dilantha1981@gmail.com, ratnasiri@cheng.mrt.ac.lk

[2]*Telemark University College, P.O. Box 203/Postboks 203, N-3901 Porsgrunn, Norway.*

E-Mail; Bernt.Lie@hit.no

Abstract

Beer is the most common alcoholic drink around the world. When talking about the beer quality, flavor of the beer is more concerned. Most of the brewers in Sri Lanka are using traditional methods to brewing beer. Most of brewers using pre identified recipe to produce mass production of beer. Therefore, beer quality i.e. flavor is varying brand by brand. It is important to study the variation of temperature how will effect to the final alcohol production and the flavor compound formation. Beer manufacturing industry can be used this knowledge to increase the production efficiency and the product quality.

It is very important to know about dynamics of forming flavor compounds. In this work the fermentation process is concerned since all the flavor compounds are formed during the fermentation. The mechanistic model is developed by based on the knowledge of biochemical processes in the yeast cell and previously developed mathematical models which are available in the literature. There are many beer models can be found such as Engasser et al.,(1981);Growth kinetic model, Gee at al.,(1988), Phisalaphong et al.(2006);Growth kinetic model and effect on temperature, W.Fred Ramirez and Jan Maciejowski.,(2007);Optimal beer fermentation, etc. The beer fermentation process is modeled and simulated in MATLAB environment.

Growth model, nutrient model, and the flavor model are considered and developed. Growth model consists of sugar consumption, biomass growth and ethanol formation models. Those models are developed with temperature dependant parameters to observe the effect of temperature. Three amino acids which are Valine, Leucine and Iso Leucine are considered for the Nutrient model. Consumption of these three amino acids is considered during fermentation. Flavor model is developed based on the growth model and the nutrient model. Flavor compounds are categorized into three groups which are Fusel alcohols, esters and vicinal diketones. Altogether nine parameters are considered as flavor compounds and the effects of temperature on those are simulated with MATLAB. Industrial temperature profile is obtained and applied for the developed model and simulated in MATLAB and the results are analyzed.

PI controller is applied to get identified temperature profile to obtain optimal flavor formation and the dynamic model is used for find suitable controller parameters for best control.

A two phase MPC and its application to a grinding process

Alexey Zakharov*, Alexandre Boriouchkine*, Sirkka-Liisa Jämsä-Jounela*

** Aalto University, P.O. box 16100, 00076, Aalto, Finland. e-mail: alexey.zakharov@ tkk.fi*

Abstract: The growing complexity of the control systems and the increased use of nonlinear models cause a dramatic increase in the computational requirements of MPCs. Therefore, more computationally efficient MPC are needed. This paper presents a two-phase MPC approach for decreasing computational demand without sacrificing its efficiency. The first phase of the MPC treats the input variables as independent decision variables of the objective optimization, since the largest part of the objective value arises from a few earliest sampling intervals. In contrast, the second phase combines input variables, defining the rest of the MPC objective value, in an open-loop control which is specified by a few independent decision variables. The method is compared against the traditional Quadratic Programming implementation of an MPC for the Grinding Plant control problem. The two-phase MPC demonstrates a better performance compared with the traditional controller with the same control horizon.

Keywords: MPC, dynamic optimization, grinding process, industrial application, stability

1. INTRODUCTION

Current economic conditions has set new challenges for productivity and keeping the operational conditions of processes within required boundaries. These challenges have resulted in the growing complexity of industrial control systems. Consequently, the increasing complexity of control systems requires more efficient advanced level (higher-level) control strategy which almost universally utilizes model predictive controllers (MPCs). However, the use of MPCs can be computationally heavy because of several factors, such as complex process models, as well as a greater number of manipulated and controlled variables, and constraint management. Nevertheless, a limited computational power forces to make a compromise between efficiency and computational time of a control strategy. However, in several applications the control strategy efficiency is critical. To shorten the computational time without losing efficiency, a modification must be done to existing MPC techniques. Several options exist for improving the efficiency of an MPC. One of such techniques is the dynamic programming, since it provides many useful insights into the MPC performance problems. However, dynamic programming reduces the problem to Hamilton-Jacobi-Bellman (HJB) equation that in most cases can not be solved analytically, because it is a partial differential equation, and solving HJB equation is computationally unattractive due to the high dimensionality. For this reason, Dynamic Programming is not applied directly to practical control problems, but instead, different simplified methods are used.

In the 60s, the development of Dynamic Programming-employing control techniques led to the invention of the linear quadratic regulator (LQR). The LQR is a rare case when the HJB equation can be solved analytically. The LQR was found to be useful for many practical applications and it might be considered as the direct predecessor of modern MPCs (so called 'zero generation' of MPC).

Unfortunately, the linear-quadratic regulator is limited by linear dynamics, quadratic objective function and absence of constraints, thus leaving many industrial problems out of its scope. Moreover, it is well acknowledged that economic operating points in typical process units often lie at the intersection of constraints. As a result, a successful industrial controller must keep the system as close to the constraints as possible without actually violating them. Thus, the next generation of MPC appeared having the following main features: linear process constraints, a linear process model, a quadratic objective and a finite time horizon (see Richalet et al. (1978), Prett et al. (1982)). The finite horizon was used to approximate the infinite horizon problem, which hardly can be solved. Since in the presence of constraints even the solution of the finite horizon optimization problem cannot be derived analytically, the quadratic programming was employed to perform the optimization.

In the early 90s, it was discovered that the constrained optimization can cause feasibility problems, especially, when large disturbances appear. Therefore, the most of the modern MPC software products have been enforced to use soft output constraints (Qin and Badgwell 2003).

On the other hand, because of the finite horizon formulation, MPC faced stability problems. Attempts to achieve stability included different prediction and control horizon approaches and the introduction of a terminal cost to the MPC objective. These methods were criticized in the study of Bitmead et al. (1990) as 'playing games', because there were no clear conditions to guarantee stability. Thereby, the stability of MPC was studied actively during the early 90's (Keerthi and Gilbert (1988), Mayne and Michalska (1990) are among the first papers exploring this question) and a comprehensive review of these studies is provided in Mayne et al. (2000).

Briefly, the stability is almost universally established through the use of the value function of MPC as a natural Lyapunov function. On the other hand, the Dynamic Programming provided some useful insights concerning the MPC stability. One example of that is the 'inverse optimality principle', which is used to ensure the stability of MPC by utilizing the fake HJB equation (for details, see for example Bitmead et al 1990, Magni and Sepulchre 1997).

It is well known that the performance of MPC depends on the quality of underlying model: an MPC is as good as its model. For that reason during the last decade the focus was moved to the nonlinear MPC utilizing of a more accurate nonlinear process model. Basically, the implementation of such MPC cannot be based on QP anymore. Therefore, the convex optimization techniques are employed instead of QP.

Consequently, the nonlinearity of the models along with other factors, such as the complexity of control systems, increases the computational requirements for MPCs. However, the computational requirements of MPCs are critical for many applications, especially, for large and fast processes. Therefore, many researchers have concentrated their efforts on reducing the number of on-line computations (Bemporad et al. 2002, Pannocchia et al. 2007, Rao et al. 1998).

In contrast to computational requirements and stability, another important property of an MPC, namely, optimality did not attract so much attention in the literature, even though the finite-horizon MPCs do not provide the optimal solution of problems. In general, the researchers do not focus on the exploration of optimality because of the idea that a close-to-optimal solution may be found through increasing the control horizon. However, an example is given in Di Palma, Magni (2007), where MPC performance is not a non-decreasing function of the optimization horizon. In addition, a longer control horizon also requires more computations and a compromise must be made between the close-to-optimal properties of the controller and its computational demands.

Simultaneously with traditional MPC development, some attempts were made to estimate the solution of HJB equation indirectly. For example, an iterative approach was proposed in Sardis and Lee (1979). Unfortunately, until today, highly efficient methods based on HJB equation have not been developed and the conclusion was made in Cannon (2004) that 'compared with conventional NMPC the computational burden of currently available methods for the HJB successive approximation approach remains prohibitive'.

Even HJB equations are unattractive for numerical implementation, the Dynamic Programming appears still to provide a useful insight into the MPC optimality. One example is presented in the work of Grune and Pannek (2009), where HJB equation was employed to estimate the 'degree of suboptimality' of MPC solutions, which was further used for adaptive determining of the MPC horizon.

Another idea risen from Dynamical Programming, (which present research is focused on) is the desire to have the terminal cost of MPC as close to the value function of the infinite horizon problem as possible (Mayne et al. 2000). If the value function is employed as a terminal cost, MPC provides the optimal solution even with the time horizon being unity. In particular, the solution of Riccati equation, which is the value function of the unconstrained infinite horizon problem, is proposed as the terminal cost for MPC objective in many papers (see Chmielewski and Manousiouthakis (1996), Sznaier and Damborg (1987)). Although the stability is attained within the approach, in general, it is not possible to expect that a 'good' approximation of the value function can be found. In particular, if the MPC setpoint lies on the border of constraints, quadratic functions cannot capture the essential asymmetry of the value function.

In the present paper, the emphasis is moved on the estimation of the value function of the infinite horizon problem, which provides close-to-optimal behaviour of the controller even with a short control horizon. Thus, the method achieves a decrease in computational demand without sacrificing its efficiency. An industrial application (a model of a Grinding process) is used to test the developed method.

The paper is organized as follows. Section 2 contains a description of the proposed MPC controller, and Section 3 a description of the grinding process. In Section 4, the results are presented and compared against a QP implementation of MPC, and Section 5 contains the conclusion.

2. DESCRIPTION OF THE TWO-PHASE MPC

2.1 The idea of the two-phase MPC

In this section, the two-phase method will be presented for a simple linear discrete state space dynamics:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k+1) &= Cx(k+1) \end{aligned} \qquad (1)$$

where $x = (x_1, x_2, ..., x_n)$ is the vector of the current state of the system, $y = (y_1, y_2, ..., y_m)$ is the vector of the system outputs, and $u = (u_1, u_2, ..., u_l)$ is the vector of the input variables. For the sake of simplicity, it is assumed that there is no noise in neither, the dynamics nor in the measurements, and the state of the system is exactly known. In addition, the process is assumed to have M linear constraints:

$$P_i y(k) \le q_i, i = 1,2,...,M \ . \qquad (2)$$

Under dynamics (1) and constraints (2), the optimal setpoint y^* is usually defined by a higher level of the control hierarchy (for example at the real time optimization layer). At this setpoint the steady state of the system x^* and the optimal steady state control u^* are defined using dynamics (1).

A typical objective function of a MPC with the control horizon equals N has the following form:

$$J_N(x(0),u) = \sum_{k=0}^{N-1} l(y(k), \Delta u(k)) + F(x(N), u(N-1)) \qquad (3)$$

where different forms of $l(y,\Delta u)$ may be used in different controllers and the terminal cost $F(x,u)$ is needed to stabilize the controller. In fact the widely used approach, which spreads the control action at control horizons N until the prediction horizon K, employs the terminal cost of the following form:

$$F(x(N),u(N-1)) = \sum_{k=N}^{K-1} l(y(k),0) \tag{4}$$

Another popular option is the solution of Riccati equation is used as the terminal cost.

On the other hand, dynamic programming theory provides the ideal candidate for the role of the terminal cost, which is able to guarantee both stability and optimality of the solution even with the control horizon equals one. Indeed, the finite horizon formulation is just a simplification of the original infinite horizon optimization problem with dynamics (1), constraints (2) and the following objective:

$$\min J_{\infty}(x(0),u) = \sum_{k=0}^{\infty} l(y(k),\Delta u(k)) . \tag{5}$$

According to dynamic programming, the optimal control action may be found as

$$\begin{aligned} &u^*(x(0),u(-1)) = \\ &= \arg\min_{u(0)} \left(l(y(0),\Delta u(0)) + V(x(1),u(0)) \right) \end{aligned} \tag{6}$$

where $V(x(1),u(0))$ is the value function introduced as the optimal value of the infinite horizon optimization problem:

$$V(x(1),u(0)) = \min_{u} \sum_{k=1}^{\infty} l(y(k),\Delta u(k)) . \tag{7}$$

In fact, MPC and Dynamic programming derive current control through minimizing items (3) and (6) respectively, and the value function plays the same role in Equation (6) as the terminal cost plays in Equation (3). Thus, MPC approach may be considered as an implementation of dynamic programming ideas, but with the 'inaccurate' approximation of the value function. Thus a control horizon longer than one must be employed to obtain the satisfactory performance of MPC even though the terminal cost inaccurately estimates the value function of the infinite horizon problem.

If the open-loop control u^*, which is optimal in state x, was known, function $V(x(N),u(N-1))$ could be easily estimated to any reasonable accuracy. However, the optimal control is unknown and a set U of second phase open-loop controls must be used to get a relatively good estimation of the value function as follows:

$$V_K^U(x(N),u(N-1)) = \min_{u\in U} \sum_{k=N}^{K-1} l(x(k),\Delta u(k)) , \tag{8}$$

here K is the second phase horizon. In particular, the common MPC with different prediction and control horizons uses set U consisting of a single open-loop control which expands the values of the input variables at the control horizon until the end of the prediction horizon.

A set of second phase open-loop controls U containing the optimal control $u^*(x(0),u(-1))$ allows to get the accurate result:

$$V(x(N),u(N-1)) = \lim_{K\to\infty} V_K^U(x(N),u(N-1)) . \tag{9}$$

Therefore the set of control strategies U must be 'divorce' in a sense that at any point $x(N),u(N-1)$ a close-to-optimal open-loop control can be found in U. On the other hand, the computational complexity of the MPC controller grows for wider sets. Thus, if the set of second phase open-loop controls is chosen as a parametric family of functions, the number of parameters must not be very high in order to avoid the high computational complexity of the method.

Since for any set U consisting of more than one open-loop control the computation of the terminal cost (8) involves an optimization of the system dynamics after the control horizon, this optimization is called 'the second phase' of the proposed MPC.

2.2 Two sets of second phase open-loop controls

Let us consider the following one-parametric set of functions presented in Figure 1:

$$g_{\alpha}(k) = \alpha\exp(-c_1 k) + (1-\alpha)\exp(-c_2 k) , \tag{10}$$

where coefficients c_1 and c_2 are fixed. In order to define a open-loop control for the whole MPC, it is needed to define an individual control for each input variable. This can be done in the following way:

$$u_i(N+k-1) = \left(u_i(N-1) - u_i^*\right) g_{\alpha(i)}(k) + u_i^* , i = 1,...,l , \tag{11}$$

here $u*$ is the steady state optimal control. Thus the system open-loop control is defined by vector $\alpha = (\alpha_1,\alpha_2,...,\alpha_l)$, where every element defines the control for the respective manipulated variable.

The proper choice of constants c_1 and c_2 may be a problem if the described above set of open-loop controls is used. In order to avoid it, another set of second phase open-loop controls is introduced by adding a time scale parameter β as follows:

$$g_{\alpha,\beta}(t) = \alpha\exp(-c_1\beta t) + (1-\alpha)\exp(-c_2\beta t) . \tag{12}$$

The open-loop controls of the MPC are constructed similarly to Equation (11):

$$u_i(N+k-1) = \left(u_i(N-1) - u_i^*\right) g_{\alpha(i),\beta}(k) + u_i^* , i = 1,...,l \tag{13}$$

Here different parameters α_i are used for different input variables of the MPC, but a single parameter β is used to define the time scale for all input variables. In the present paper, the set of second phase open-loop controls defined by Equation (13) is used for two-phase MPC implementation and testing.

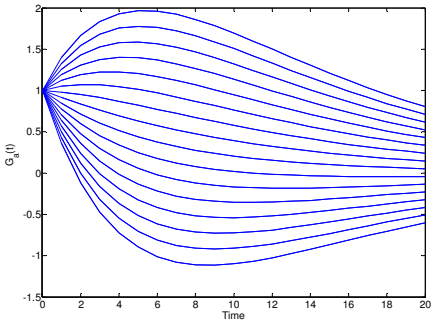


Figure 1. One-parametric set of functions $g_\alpha(k)$ with $c_1 = 0.2$ and $c_2 = 0.1$.

3. DESCRIPTION OF THE GRINDING PLANT

Communition is a huge consumer of electrical power because crushing rocks into powder requires a lot of energy. According to Pomerleau et al. (2000), grinding typically accounts for almost 50% of the costs of a concentrator and, as a result, the optimization of grinding mills is an extremely important research topic. The aim of economic optimization is to maximize the feed rate or to achieve the desired particle size distribution, thus making production more profitable. An overview of the control methods of grinding plants is provided in Hodouin, Jämsä-Jounela et al. (2001).

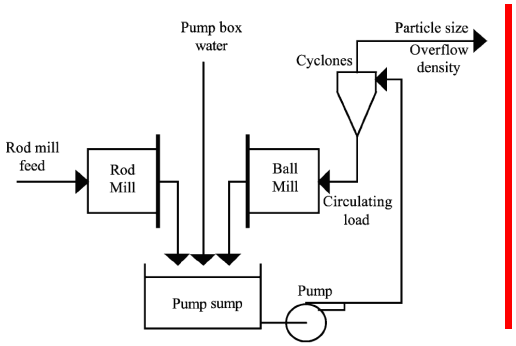


Figure 2. Grinding circuit, Lestage et al. 2002.

The ore is fed to the rod mill and then discharged into the pump sump. The slurry is then fed to a hydrocyclone, where it is separated into the overflow product and a recycled part, which is fed back to the ball-mill (for more details, see Lestage et al. 2002). The whole circuit is presented in Figure 2.

There are two manipulated variables available in the model:

- u_1: the rod-mill feed (t/h)
- u_2: the pump sump water addition (m^3/h)

and four output variables:

- y_1: the hydrocyclone overflow density (% solids)
- y_2: the fraction of particles smaller than 325 mesh (47Am) in the product (%)
- y_3: the tonnage through the ball mill (t/h)
- y_4: the pump sump level (%)

Typically the grinding process can be described by means of a transfer function of second order. The transfer function presented in Lestage et al. 2002 is used to test the MPCs:

	u1	u2
y1,	$G_{11}(s) = \frac{0.0255(1-5600s)e^{-600s}}{(1+5300s)(1+750s)}$	$G_{12}(s) = \frac{-0.14(1+4050s)}{(1+3200s)(1+60s)}$
y2,	$G_{21}(s) = \frac{-0.2(1-900s)}{(1+5200s)(1+750s)}$	$G_{22}(s) = \frac{0,012(1+39500s)}{(1+4400s)(1+50s)}$
y3,	$G_{31}(s) = \frac{13.8}{(1+5700s)(1+400s)}$	$G_{32}(s) = \frac{4.2(1-700s)}{(1+5000s)(1+5s)}$
y4,	$G_{41}(s) = \frac{5,749}{(1+5500s)(1+210s)}$	$G_{42}(s) = \frac{1,962}{(1+4700s)}$

The model is converted into the discrete time state space form (1) with a sample time equals 300s. The system is described with 15 state variables and four controlled variables.

The constraints are defined as product specifications: the hydrocyclone overflow density (y_1) must be above 48% to meet the flotation requirements, and must be below 52% to avoid sedimentation problems. Product specification fineness (y_2) is defined as 47% of the particles smaller than 47 μm. In order not to overload the ball mill, the throughput (y_3) must not exceed 820t/h. The pump sump level (y_4) must remain between 15% and 85%. The constraints are defined in the following order: lower and upper constraints on y_1, lower and upper constraints on y_2, upper constraint on y_3, and lower and upper constraints on y_4. Thus, matrices P and q take the following form:

$$P = \begin{array}{c} \begin{matrix} y_1 & y_2 & y_3 & y_4 \end{matrix} \\ \begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -1 \end{bmatrix} \end{array},$$

$q = [52\%, -48\%, 47\%, -47\%, 820\text{t/h}, 85\%, -15\%]^T$.

4. COMPARISON OF THE TWO-PHASE MPC AGAINST THE QP IMPLEMENTATION

Next, the two-phase MPC method is compared against the soft-constrained QP implementation of MPC.

In order to achieve smooth trajectories, the following objective function is used for both controllers:

$$F(x) = \sum_{k=0}^{K} (u(k) - u(k-1))' R (u(k) - u(k-1)) + \sum_{k=1}^{K} \sum_{i=1}^{M} Q_i (\max(P_i y(k) - q_i, 0))^2 + + \sum_{k=1}^{K} (y(k) - y^*)' S (y(k) - y^*) \quad , \qquad (14)$$

where diagonal matrixes R, Q and S are defined as follows:

$$R = \begin{pmatrix} 25 & 0 \\ 0 & 10 \end{pmatrix},$$
$$Q_1 = Q_2 = 100, Q_3 = Q_4 = 25, Q_5 = Q_6 = Q_7 = 1. \qquad (15)$$
$$S_{11} = 5, S_{22} = 0, S_{33} = 0{,}01, S_{44} = 0{,}05$$

To test the controller's ability to follow changing operating conditions, the constraints are varied as shown in Table 1. In every case, the optimal steady states, presented in Table 2, are obtained by maximizing the throughput of the plant.

Table 1. Constraints

Time period (h)	Overflow solids (%)		Particles < than 47µm (%)		Ball mill through t/h	Pump sump level (%)	
	min	max	min	max	max	min	max
0…0.25	48	52	47	47	820	15	85
0.5…5	48	52	49	49	820	15	85
5…10	50	52	49	49	820	15	85
10…15	50	52	47	47	820	15	85

Table 2. Setpoints

Time period (h)	Overflow solids (%)	Particles < than 47µm (%)	Ball mill throughput t/h	Pump sump level (%)
0…0.25	51,83	47,0	820,0	62.06
0.25…5	48,0	49,0	811,95	64.5
5…10	50,0	49,0	739,32	30,87
10…15	51,83	47,0	820	62,06

The prediction horizon K in QP formulation of MPC is always 10 steps longer than the control horizon N. For the two-phase MPC the second phase length is also taken to be 10 steps. QP MPC is tested with control horizons equal to 2 and 10 and two-phase MPC is tested with control horizon equal 2.

The second-phase open-loop control set described by Equation (13) is used for the two-phase MPC implementation. Since there are only two input variables, a second-phase control is defined by three parameters: α_1 and α_2 select the shape of the open-loop controls of the first and the second input respectively, while a parameter β is used for time scaling. Discrete values of α_1 and α_2 are considered with a step equal 0.4 while β is taken in the following form: $\beta = 0.9^k$ with k having only integer values.

Two controlled variables are compared in Figure 3 and the manipulated variables are presented in Figure 4. In general it is clear that the two-phase MPC with control horizon $N = 2$ demonstrates behaviour, which is closer to QP MPC with longer control horizon $N = 10$ rather than the same control horizon $N = 2$.

The efficiency of the presented methods may be evaluated using the objective function values of every method presented in Table 4. Again, two-phase MPC achieves the same value of the objective function as the QP MPC with the longer prediction horizon $N = 10$ does. QP MPC with the short control horizon $N = 2$ demonstrates 13% bigger objective value.

Table 4. Objective for QP MPC and two-phase MPC

	QP MPC, N = 2	QP MPC, N = 10	MPC 2 phase N = 2
Constraint 1	971	978	993
Constraint 2	1 845	1 358	1 392
Constraint 3	0	0	0
Constraint 4	0	0	0
Setpoint 1	496	446	468
Setpoint 3	1 326	1 106	1 139
Setpoint 4	1 039	889	922
Control 1	827	886	781
Control 2	1 037	1 003	968
Objective	**7 543**	**6 669**	**6 665**

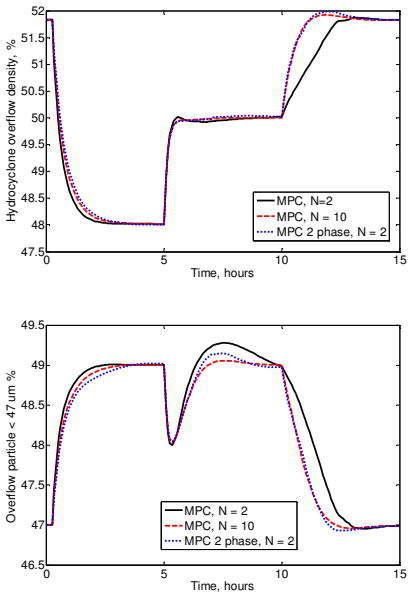


Figure 3. Controlled variables: a – Hydrocyclone overflow density % solids, b – % of particles < 47 µm

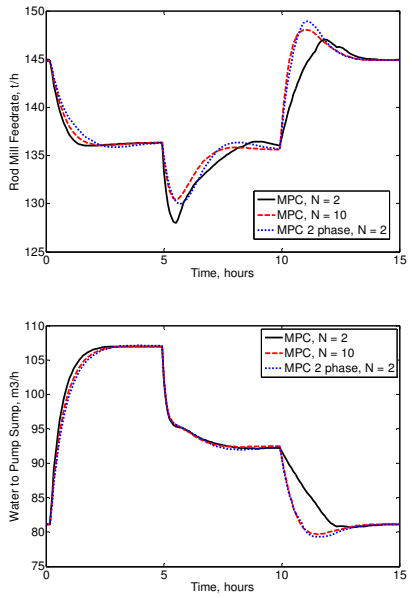


Figure 4. Manipulated variables: a – rod mill feedrate, b – water to pump sump

5. CONCLUSION

A two-phase MPC controller is described in the paper, where the best second phase open-loop control is chosen from a predefined set of open-loop controls. Thus, a more accurate estimation of the value function is used as a terminal cost in comparison with the QP formulation. The numerical comparison against the QP formulation of MPC has shown that similar performance might be achieved with a shorter control horizon. Thereby a progress in the trade-off between the performance and computational demands of MPC is made.

REFERENCES

Fleming W.H., Rishel R.W., (1975), *Deterministic and stochastic control*. New York, Springer.

Richalet J., Rault A., Testud J. L., Papon J., (1978), Model predictive heuristic control: Applications to industrial processes, *Automatica* 14, pp. 413-428.

Prett D.M., Ramarker B.L., Cutler C.R., (1982), Dynamic matrix control method, *United States Patent 4349869*.

Qin S.J., Badgwell T.A. (2003), A survey of industrial model predictive control technology, *Control Engineering Practice*, 11, pp. 733-764.

Bitmead R.R., Gevers M., Wertz V., (1990), *Adaptive optimal control–The thinking man's GPC*. Englewood Cliffs, NJ: Prentice –Hall.

Keerthi S. S., Gilbert E. G., (1988), Optimal, infinite horizon feedback laws for a general class of constrained discrete time systems: Stability and moving-horizon approximations, *Journal of Optimization Theory and Application,* 57, pp. 265-293.

Mayne D. Q., Michalska H., (1990), Receding horizon control of on-linear systems, *IEEE Transactions on Automatic Control,* 35(5), pp. 814-824.

Mayne D.Q., Rawlings J.B., Rao C.V., Scokaert P.O.M. (2000), Constrained model predictive control: stability and optimality, *Automatica* 36, pp. 789-814.

Magni L.,Sepulchre R., (1997), Stability margins of nonlinear receding-horizon control via inverse optimality, *System & Control letters*, 32, pp. 241-245.

Bemporad A., Morari M., Dua V., Pistikopoulos E.N. (2002), The explicit linear quadratic regulator for constrained systems, *Automatica* 38 pp. 3-20.

Pannocchia G., Rawlings J.B., Wright S.J. (2007), Fast, large-scale model predictive control by partial enumeration, *Automatica*, 43, pp. 852-860.

Rao C.V., Wright S.J., Rawlings J.B. (1998), Application of Interior-Point Methods to Model Predictive Control, *Journal of optimization theory and applications*, 99(3) pp. 723-757.

Di Palma F., Magni L., (2007), *On optimality of nonlinear model predictive control*, System and Control Letters 56, p 58-61.

Sardis G.N, Lee C.G., (1979), An approximation theory of Optimal Control for trainable manipulators, *IEEE Transactions on system, man, and cybernetics*, 9(3), pp. 152-159.

Cannon M., (2004), Efficient nonlinear model predictive control algorithms, *Annual Reviews in Control*, 28, pp. 229-237.

Grune L., Pannek J., (2009), Practical NMPC suboptimality estimates along trajectories, *System & Control Letters*, 58, pp 161-168,

Chmielewski D.J., Manousiouthakis V., (1996), On constrained infinite-time linear–quadratic optimal control, *System and Control Letters*, 29, pp. 121-129.

Sznaier M., Damborg M.J. (1987), Suboptimal control of linear system with state and control inequality constraints, *Proceedings of 26th Conference on Decision and Control*, Los Angeles, CA, December 1987.

Pomerleau A., Hodouin B., Desbiens A., Gagnon E. (2000), A survey of grinding circuit control methods: from decentralized PID controllers to multivariable predictive controllers, *Powder Technology,* 108, pp. 103-115.

Hodouin D., Jämsä-Jounela S.-L., Carvalho M.T., Bergh L. (2001), State of the art and challenges in mineral processing control, *Control Engineering Practice*, 9, pp. 995-1005.

Lestage R., Pomerleau A., Hodouin D. (2002), Constrained real-time optimization of a grinding circuit using steady-state linear programming supervisory control, *Powder Technology,* 124, pp. 254-263.

ARX MPC for people with type 1 diabetes

Dimitri Boiroux, Daniel A. Finan, John Bagterp Jørgensen, Niels Kjølstad Poulsen and Henrik Madsen

Abstract

Type 1 diabetes is a chronic disease characterized by a lack of production of pancreatic insulin, consequently leading to high blood glucose concentrations (hyperglycemia). Hyperglycemia has negative health effects in the long term such as eye, nerve, and kidney disease. Exogenous insulin must be injected to keep the blood glucose in the normoglycemic range (approximately 60 – 140 mg/dL, or 3.3 – 8 mmol/L). However, the dosing of exogenous insulin must be done carefully, because low blood glucose concentrations (hypoglycemia) can have immediate and severe consequences like insulin shock, coma, or even death. Currently, insulin administration is performed by the subject with type 1 diabetes based on infrequent glucose measurements (in the form of finger-sticks), often resulting in an unsatisfactory blood glucose control.

An artificial pancreas is a medical device that injects exogenous insulin automatically in order to regulate the glucose concentration. Blood glucose measurements are obtained from a continuous glucose monitor (CGM). Insulin is administrated either continuously through an insulin pump, or at discrete times using an insulin pen. A control algorithm uses previous glucose measurements and insulin injection information to compute the optimal insulin administration for the current conditions.

We use model predictive control (MPC) to compute the optimal insulin administration for 20 virtual type 1 diabetes subjects. The system (i.e., subject) has one manipulated input (insulin infusion rate), one disturbance input (carbohydrate meals), and one measured output (blood glucose concentration). The subject is represented by a system of nonlinear differential equations describing the dynamic effects of insulin and meals on blood glucose [4]. Twenty parameter sets are used in the study, each representing a different virtual subject.

The model used in the MPC is a low order autoregressive exogenous-input (ARX) model [3]. Due to both the linearity and relative parsimony of the ARX model, there is a significant amount of

subject/model mismatch in the model predictions, reflecting real-world conditions. In general, a simple ARX MPC cannot reject a step disturbance without a resulting offset; thus, the state vector is reformulated using an extended ΔARX description (E-ΔARX) [4], i.e.

$$(1-q^{-1})A(q^{-1})y(t)=(1-q^{-1})B(q^{-1})u(t)+(1-\alpha q^{-1})e(t)$$

in which q^{-1} is the backward shift operator, A and B are polynomials, $e(t)$ is a white noise process, and $0 \leqslant \alpha \leqslant 1$ is a tuning parameter.

The reference signal is time-varying, and is based on the optimal open-loop glucose profile [2]. Insulin-on-board constraints are implemented to avoid overdosing insulin. State estimation is based on a Kalman filter using the noise model described in [1] to simulate a realistic CGM.

We present the MPC results for simulations of the 20 virtual subjects with type 1 diabetes. In particular, we investigate the effects of the prediction horizon length on the control quality of blood glucose and the robustness of the solution.

References

[1] Breton M, Kovatchev B. Analysis, modeling, and simulation of the accuracy of continuous glucose sensors. J Diabetes Sci Technol. 2008;2(5):853-862.

[2] Cobelli C, Dalla Man C, Sparacino G, Magni L, De Nicolao G, Kovatchev B. Diabetes: models, signals, and control. IEEE Rev Biomed Eng. 2009;2:54-96.

[3] Finan DA, Zisser H, Jovanovic L, Bevier WC, Seborg DE. Practical issues in the identification of empirical models from simulated type 1 diabetes data. Diabetes Technol Ther. 2007;9(5):438-450.

[4] Huusom J.K., Poulsen N.K, Jørgensen S.B., Jørgensen J.B. Tuning of methods for offset free MPC based on ARX model representations. 2010 American Control Conference (ACC 2010). *Accepted*.

[5] Kovatchev B., Breton M, Man CD, Cobelli C. In silico preclinical trials: a proof of concept in closed-loop control of type 1 diabetes. J Diabetes Sci Technol. 2009;3(1):44-55.

Tuning of ARX-based Model Predictive Control for Offset-free Tracking.

Jakob Kjøbsted Huusom[a], Niels Kjølstad Poulsen[b], Sten Bay Jørgensen[a] and John Bagterp Jørgensen[b]

[a]CAPEC, Department of Chemical and Biochemical Engineering
[b]Department of Informatics and Mathematical Modelling
Technical University of Denmark

Keywords: Model Predictive Control, ARX-model, Controller Tuning, Recursive estimation

Model Predictive Control (MPC) is a control technology that uses a model of the system to predict the process output over some future horizon. The controlled input signal is determined by solution of an open-loop optimization problem using the model of the system. The first part of the optimal control sequence is implemented on the process. Feedback is obtained by repeating this procedure as new measurements become available. Advanced control strategies such as Model Predictive Control have gained wide spread interest in many areas in the chemical industries, due to fast algorithms, a well established theory and growing number of successful industrial implementations. The main feature is that the optimal control signal is determined as a constrained optimization which utilizes future predictions of the plant behaviour. Hence the controller has a plant model embedded for state estimation. The achieved closed loop performance is therefore dependent on the quality of the future predictions. The performance of the state estimator is on the other hand dependent on the accuracy of the process and the noise model. In this contribution, we discuss closed loop performance of MPC based on ARX models when applied to systems with unmeasured step disturbances.

The majority of industrial MPC applications are today still based on linear models. ARMAX models (AutoRegresive Moving Average model with eXogenous input) can be identified using standard tools from time series analysis and systems identification. However, for multivariable systems it is difficult to select a structure for the ARMAX model. Furthermore, identification of the parameters in ARMAX models constitutes a non-linear non-convex optimization problem. If the input-output model is simplified to an ARX model (AutoRegresive model with eXogenous input), the estimation problem becomes a convex optimization problem. Furthermore the MIMO system can be handled as easily as SISO systems. The discrete time ARX model structure is given as

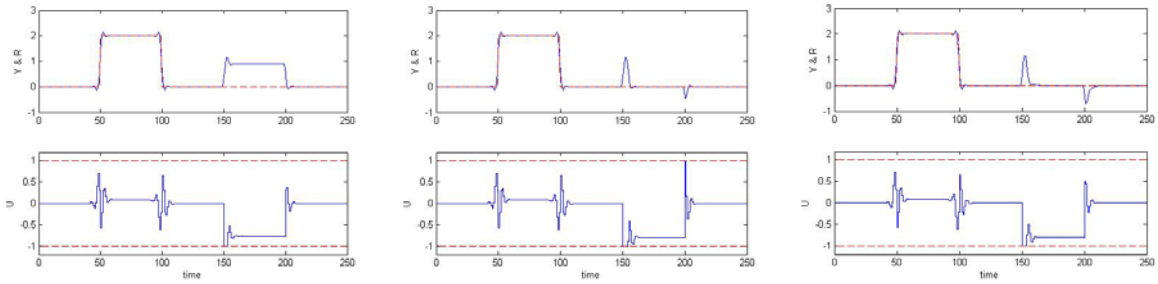
$$A(q^{-1})y_k = B(q^{-1})u_k + \varepsilon_k \qquad \varepsilon_k \sim N(0,\sigma^2)$$

where $A(q^{-1})$ and $B(q^{-1})$ are polynomials in the back shift operator q^{-1}.

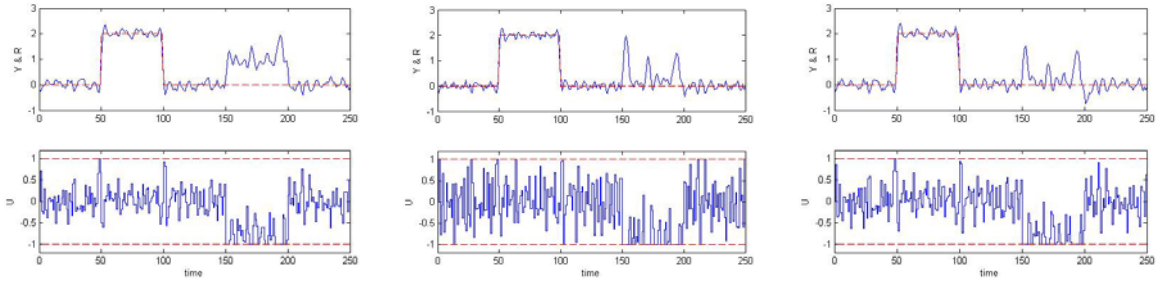
Unmeasured step disturbances are common in the process industries and appear for instance when a feed source changes. The composition of crude oil in refineries may change significantly when feed is changed from one well to another. Similar unknown steps may occur in a cement plant when raw minerals are changed. Unmeasured steps may also be used to represent the inevitable model-plant mismatch. To reject such disturbances the basic MPC formulation needs to be expanded, such that model errors are introduced in order to include integrators or augment the system with disturbance models. Alternatively the model predictive control algorithm can be combined with a recursive disturbance estimation algorithm. The closed loop performance of the system will depend on the nature of the disturbance and how the disturbance rejection is facilitated by the control algorithm.

This contribution will analyse a range of methods for achieving offset-free tracking in ARX-based model predictive control. Considerations regarding pre-tuning of the free parameters in the controller are also provided based on the infinite horizon, unconstrained model predictive controller, i.e. the LQG control design. The tuning aims at balancing the ability to reject unmeasured disturbances vs. noise sensitivity while keeping both the closed loop input and the output variance small. This type of trade off is illustrated on a small example. A series of closed loop simulations of the MPC has been performed using the third order ARX model as the true system. Three MPC implementations are tested on a fixed scenario. The total simulation horizon is 250 samples. Between time 50 and 100 a step is introduced in the reference and between 150 and 200 an unmeasured step disturbance is acting on the system. The input will be constrained between ±1. The models used by the controller are the ARX model, the ΔARX model and finally the EΔARX model. The Δ in model name indicates that the noise is modelled as integrated white noise. The EΔARX model use an extended model formulation of the noise and has a free parameter α which can vary between 0 and 1 in order to span the range between the two other models.

Deterministic simulations



Noisy simulations



ARX-MPC ***ΔARX-MPC*** ***EΔARX-MPC(α=0.7)***

It is clearly seen on the figure that only the ΔARX and the EΔARX models can give offset free tracking. The output trajectories resulting from these two models are very similar but the EΔARX model give a less aggressive control action.

Current research involves extensions and benchmarking of the framework using ARX-based, offset-free MPC algorithms for a series of examples relevant for the process industry involving multivariable and time delay systems.

Comparison of Decentralized Controller and MPC in Control Structure of a CO_2 Capturing Process

Mehdi Panahi, Sigurd Skogestad

Department of Chemical Engineering, Norwegian University of Science and Technology(NTNU), 7491 Trondheim, Norway

Abstract

In a previous study [1], we did the control structure design for a post combustion CO_2 capturing process using Skogestad's method [2]
Based on the designed structure, in this paper we investigate the performance of the control structure with large disturbances where some of the manipulated variables saturate.
To handle this kind of disturbances, a MPC has been designed and implemented on parts of the process and the results show that we achieve good performance. On the other hand, if we implement decentralized PID controllers, one needs to use the reverse pairing of what would be desired for good performance in order to avoid instability when we have saturation.

Keywords: Process control, Plantwide control, Self-optimizing method

Introduction

Absorption/Stripping CO_2 capturing processes are used in post-combustion part of power plants to remove CO_2 from flue gas streams. A simple typical flowsheet of such processes has been shown in Fig.1

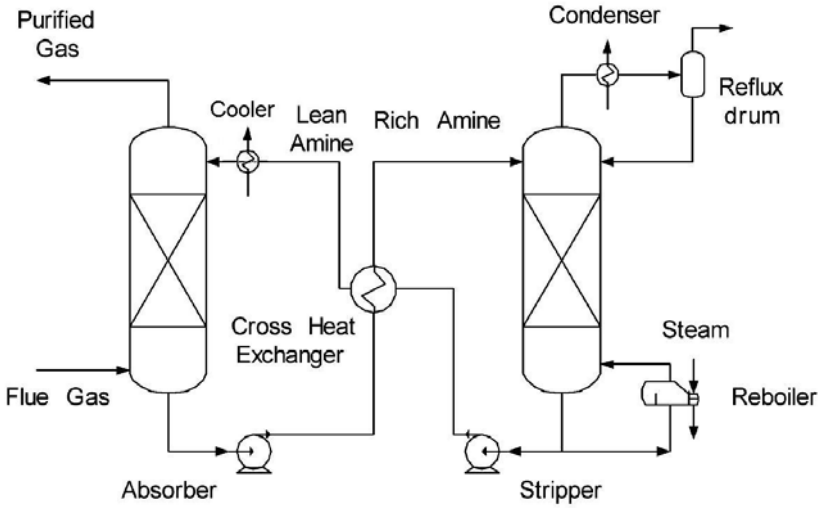


Fig.1- Absorption/Stripping CO_2 capturing process [2]

Energy requirements of CO_2 capturing processes are relatively very high compare to the net generated power in the main plant and operating of this process in the lower operation costs is important to save more energy. To keep this process in the minimum level of energy requirement, a robust control structure is needed.
We have designed a control structure using Skogestad's method to omit the necessity of reoptimization of the process when disturbances happen with an acceptable loss for the energy requirement of the process. Fig.2 shows the proposed control structure. [1]

In this structure, recycle amine flowrate is manipulated to remove 90% of the CO_2 content in the feed flue gas and temperature control of tray no.4 (counting from the top) in the stripper (20 trays) has been found as the best self-optimizing variable to be controlled by using the only unconstrained degree of freedom that is reboiler duty of the stripper. [1] The other control loops are level controllers or are the loops to control the variables that have been active during self-optimizing stepwise approach. [3]

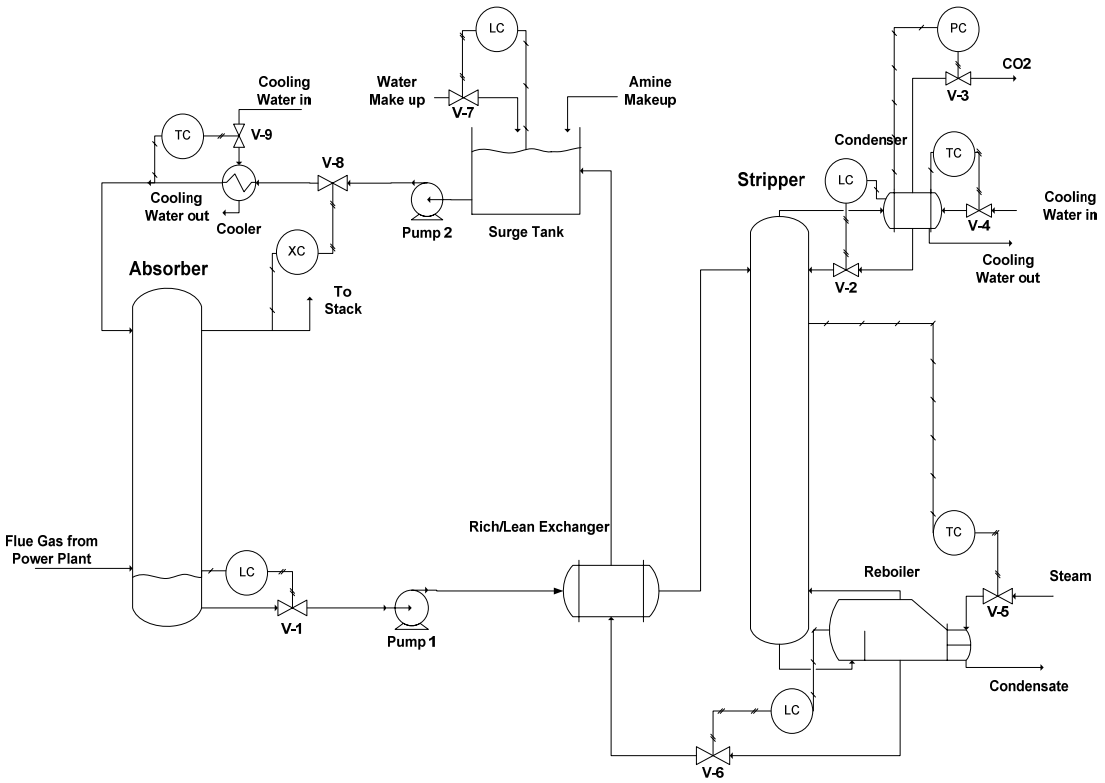


Fig. 2- Process flowsheet with proposed control loops using self-optimizing method [1]

One of the main disturbances in this process is changing of the feed flowrate into the absorber. In this paper we try to investigate and compare the performance of MPC and traditional PID controllers for stability of the proposed control structure. The MPC controller is designed and implemented using Honeywell technology that is named as Robust Model Predictive Control Technology (RMPCT) in UniSim process simulator. [4]

Performance of PID loops and MPC for stabilization of the process

The proposed control structure using decentralized PID controllers gives a stable system as long as the change in flowrate of the flue gas is less than +15% though saturation of reboiler duty happens when flowrate of flue gas reaches to +10%. But, when the accession of flowrate passes +15% then temperature of tray no.4 decreases significantly (Fig.4) and consequently the amount of CO_2 at the bottom of stripper starts increasing and with recycling to the absorber, CO_2 is accumulated in the process and makes the plant unstable. (Fig.5)

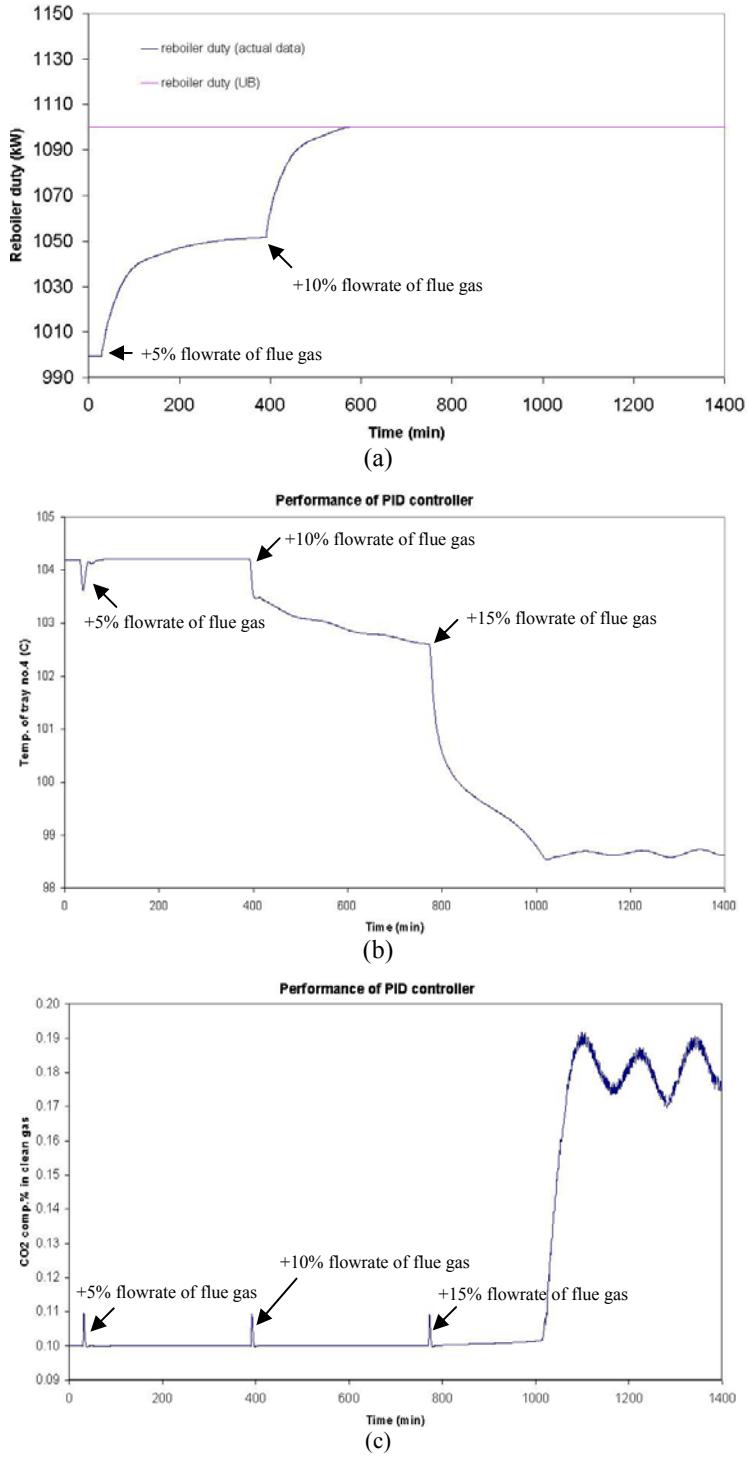
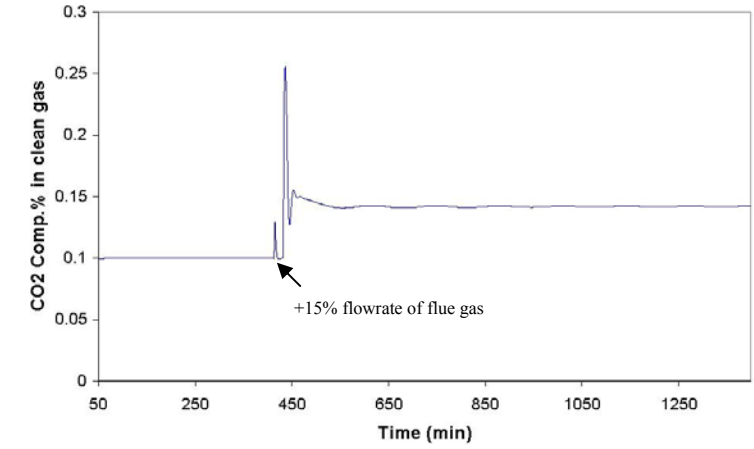


Fig. 3- a: saturation of reboiler duty when the flowrates of flue gas increases to +10%, b: when reboiler duty is saturated, by increasing feed flow to +15%, temperature of tray no.4 decreases significantly, c: instability of the process when flowrate of flue gas increases +15%

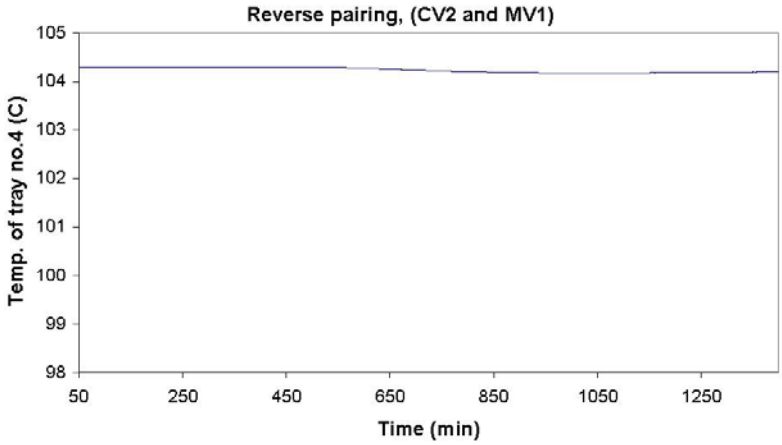
To avoid instability in our control structure we have considered two different ways: Reverse pairing and design a MPC.

Reverse pairing

We control the temperature of tray no.4 using recycle amine flowrate and give up controlling of CO_2 content in the clean gas when reboiler duty has become saturated. (Fig. 6a and b) Fig. 6b shows that by using recycle amine flowrate we can control tray temperature no. 4 in its setpoint value.



(a)



(b)

Fig. 6- By reverse pairing and giving up controlling the CO_2 content in the clean gas when reboiler duty is saturated, control structure is stable when flue gas increases +15%

MPC (RMPCT)

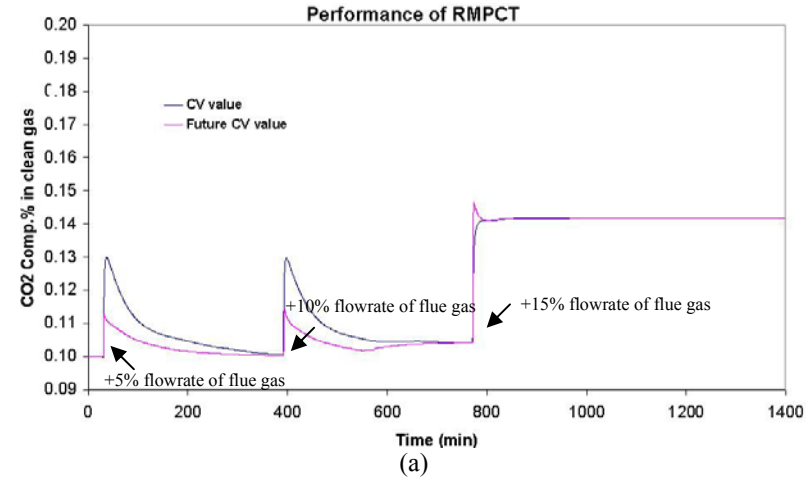
A MPC controller is used in order to achieve CO_2 composition control in the clean gas stream and temperature control of tray no.4 in the stripper. In order to keep track of the pressure drop over the stripper, the difference between bottom stage pressure and top stage pressure is included as an auxiliary CV in the identification routine. List of CVs, MVs and DVs for this controller are in table 1.

Table 1. List of CVs, MVs and DVs for design a MPC

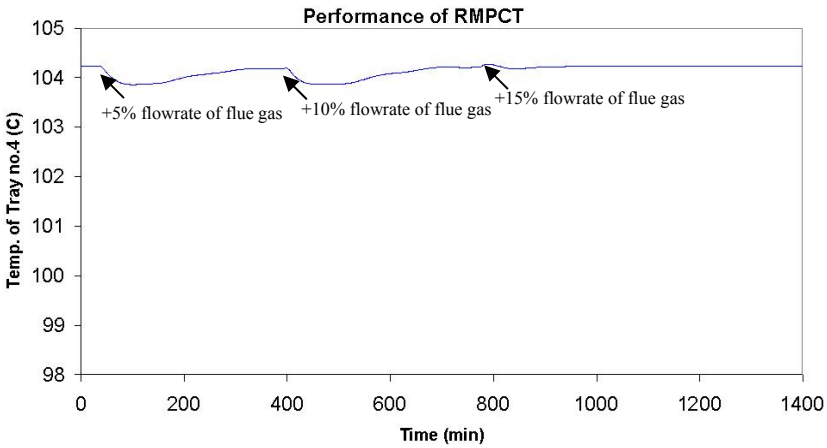
CVs	MVs	DVs
CV1- CO_2 content in clean gas CV2- Temperature of tray no.4 in the stripper CV3- pressure drop in stripper	MV1- Recycle amine flowrate to absorber MV2- reboiler duty of the stripper	DV1- change in flowrate of flue gas (feed)

Identification of the model and making the required data files for the MPC are made by Profit Design Studio (PDS) [5] and then the generated files are loaded in UniSim to run the RMPCT.

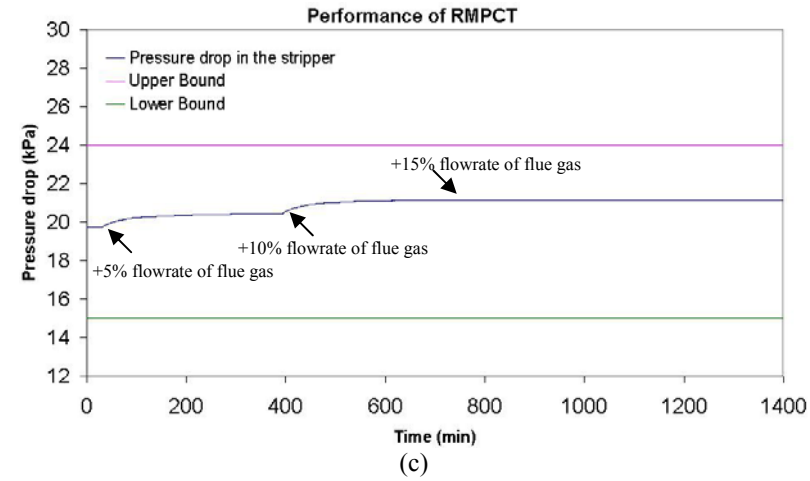
Fig. 7 shows the performance of RMPCT for controlling this process when disturbance with different magnitude happens. As Fig. 6a shows in +15% change of flowrate of flue gas, RMPCT gives up controlling of CO_2 content in the clean gas while it control tray temperature in its setpoint.



(a)



(b)



(c)

Fig. 7- a: CO_2 content in clean gas stream, RMPCT has given up controlling of CO_2 content in +15% of the disturbance, b: Temperature of tray no.4 in the stripper is always in its setpoint c: pressure drop of stripper

Conclusion

Performance of RMPCT and decentralized PIDs for stabilization of a designed control structure by self-optimizing method for a CO_2 capturing process has been considered when large disturbances have happened. Results show that using RMPCT, the control structure will be stable while to keep the process stable using PID loops, reverse pairings of some MVs and CVs has to be done.

Acknowledge

We would like to acknowledge to Bjørn Einar Bjartnes from Honeywell Company for helping in implementing of the RMPCT in UniSim.

References

[1] Panahi, M., M. Karimi, S. Skogestad, M. Hillestad, H. F. Svendsen, 2010, Self-Optimizing and Control Structure Design for a CO_2 Capturing Plant, presented in 2nd gas processing symposium, Doha, Qatar, Jan.12-14.

[2] Skogestad, S., 2004, Control Structure Design for Complete Chemical Plants, Computers and Chemical Engineering, 28, 219-234

[3] Skogestad, S., 2000, Plantwide control: The search for the self-optimizing control structure, Process Control, 10, 487-507

[4] UniSim design R380, Honeywell company

[5] Profit Design Studio (PDS R310) , Honeywell company

Potential of Economic Model Predictive Control for Management of Multiple Power Producers and Consumers

Tobias Gybel Hovgaard*, John Bagterp Jørgensen**

*Danfoss A/S, Nordborgvej 81, DK-6430 Nordborg, Denmark
tgh@danfoss.com
**DTU Informatics, Technical University of Denmark, Richard Petersens Plads, Building 321, DK-2800 Kgs Lyngby, Denmark
jbj@imm.dtu.dk

Keywords : Model Predictive Control, Optimization, Energy Systems, Refrigeration Systems

Economic Model Predictive Control is a receding horizon controller that minimizes an economic objective function rather than a weighted least squares objective function as in Model Predictive Control (MPC). We use Economic MPC to operate a portfolio of power generators and consumers such that the cost of producing the required power is minimized. The power generators are controllable power generators such as combined heat and power generators (CHP), coal and gas fired power generators, as well as a significant share of uncontrollable power generators such as parks of wind turbines. In addition, some of the power consumers are controllable. In this paper, the controllable power consumers are exemplified by large cold rooms or aggregations of super markets with refrigeration systems. We formulate the Economic MPC as a linear program. By simulation, we demonstrate the performance of Economic MPC for a small conceptual example.

We consider a number of dynamically independent systems that have the one thing in common that they all influence a quantity that has to meet a set of constraints. To be more specific we have a portfolio of multiple power generators which delivers a total production for the entire portfolio. On the demand side a cold room consumes power in order to keep the temperature within certain bounds. Furthermore a reference consumption (which is assumed to be predictable) combining all other power consumers and non-controllable power producers like farms of wind turbines is added to the demand side. By adding the cold room to the optimization problem the potential savings gained by controlling flexible loads on the demand side are revealed. The idea is that the thermal capacity in the refrigerated goods can be utilized to store "coldness" such that the refrigeration system can pre-cool when the energy is free (i.e. there is an over production from the generators). Thereby a lower than normally required cooling capacity can be applied later, for a period of time when the energy prices are above zero again. The demands to the temperature in the cold room are not violated at any time since the same total cooling capacity is applied though shifted in a more optimal way. We exploit the special property of the refrigeration system that the dynamics of the temperature are rather slow

The 16th Nordic Process Control Workshop (NPCW'10)
Department of Automatic Control and the Process Industrial Centre, Lund University, Lund, Sweden, August 25-27, 2010

while the power consumption can be changed rapidly.

For the power generation and cooling problem we should select the cheapest possible feasible trajectory of inputs. Since the cost is related to producing the power, the problem can be stated as:

$$\min_{\{u_k\}_{k=0}^{N}} \quad \phi = \sum_{k=0}^{N} c' u_k \tag{1}$$

with:

$$c = [c_1, c_2, \ldots, c_n, 0]^T \quad u = [u_1, u_2, \ldots, u_n, T_e]^T \tag{2}$$

u_i is the input to power plant i, T_e is the input to the cold room and c_i is the cost related to u_i.

The problem in (1) is subject to:
- system equations:

$$\begin{array}{ll} x_{k+1} = Ax_k + Bu_k + Ed_k & k = 0, 1, \ldots, N \\ y_k = Cx_k + Du_k & k = 0, 1, \ldots, N+1 \end{array} \tag{3}$$

- limitations on the input:

$$\begin{array}{ll} u_{min} \leq u_k \leq u_{max} & k = 0, 1, \ldots, N \\ \Delta u_{min} \leq \Delta u_k \leq \Delta u_{max} & k = 0, 1, \ldots, N \end{array} \tag{4}$$

- cooling capacity must be positive.
- production must be higher than demand at all times.

The last constraint is softened to allow the production to be lower than the demand such that the we do not end up with an infeasible problem that cannot be solved. However the penalty on underproduction is selected sufficiently large such that power demands are met whenever possible.

The cost function and the constraints are formulated as a linear program (LP) on the form:

$$\min_{x} \ \phi = g' x \tag{5}$$

s.t.

$$Ax \geq b \tag{6}$$

where A is a block-angular constraint matrix. The LP in (5) and (6) can be solved with standard solvers.

Simulations of a small case study with two power plants (a fast and expensive and a slow but cheaper) and one cold room reveals a potential for significant savings.

Fault Detection for the Benfield Process using a Parametric Identification Approach

Johannes P. Maree[1], Fernando R. Camisani-Calzolari

Department of Engineering Cybernetics
Norwegian University of Science and Technology
7491 Trondheim, Norway

The Sasol group of companies specialises in diverse fuel, chemical and related manufacturing and marketing operations. Sasol has interests in oil and gas exploration and production, in crude oil refining and liquid fuels marketing. The efficient and economical recovery of carbon dioxide (CO_2), used in various processes at Sasol, is accomplished by utilizing the Benfield process for CO_2 extraction. The Benfield process is a thermally regenerated cyclical solvent process that uses an activated inhibited Hot Potassium Carbonate solution to remove CO_2, H_2S and other acid gas components.

The current operating philosophy for the Benfield process is to keep it simultaneously hydraulic and loaded with CO_2, as far as possible, to meet optimal profit margins from the gas circuit operations. The hydraulic load is defined as the maximum volume of CO_2 that can be processed. Regeneration efficiency has been identified as one of the major efficiency measures by UOP (Benfield technology licensor). Regeneration efficiency is a measure of how much steam is required per unit volume of CO_2 removed, and gives an indication of the unit cost, the overall pressure drop in the regeneration column, and the solution health. This regeneration is directly dependant on the CO_2 absorption in the wash columns, which in its turn is again affected by the wash flow-rates. Regular foaming and flooding diminishes the efficient CO_2 absorption into the Potassium Carbonate wash solution, resulting in inefficient regeneration. Foaming and flooding is caused by abnormally high differential pressures on either the top, middle or bottom bed. Bed differential pressures increase with increasing liquid and gas loads. High differential pressures that change erratically indicate flooding or foaming. Abnormal high and stable differential pressures indicate partial blockages in packed beds or liquid/gas distributors. Abnormal differential bed pressures are classified as multiplicative faults in abnormal process behaviour, which can be detected by monitoring parameter fluctuations.

This work proposes a parametric model-based approach to Fault Detection (FD). The proposed framework combines a parity space approach (subspace SID), to identify initial process models, with a joint state and parameter estimation method (Extended Kalman filter), to monitor parameter fluctuations used for FD. The motivation for subspace SID methods is that these methods have proven to be computational efficient where no a-priori process knowledge is necessary to estimate a system model. Subspace SID methods thus allow the user to identify black box models, which can be used to monitor processes. The challenge in fault detection with subspace methods comes in how to monitor and evaluate the vast amount of system parameters efficiently and elegantly. Re-identifying the process using subspace methods, necessary to track parameter changes, is also not a feasible solution to fault detection. The reason is due to the vast amount of data samples necessary, which must contain well-excited process

[1] This research was conducted by Mr. Johannes P. Maree, for the fulfillment of the degree Master of Engineering (Electronic Engineering), at the University of Pretoria, South Africa, under the supervision of Prof. Fernando R. Camisani-Calzolari.

dynamics. An elegant solution to the FD problem, using subspace SID methods, would thus be to identify an initial system process model, using a subspace method, where the initial identified parameters of the subspace model are re-cursively updated as new data becomes available. By updating the process parameters, without complete system re-identification, the user is able to track parameter changes, which contributes to the FD and plant-model mismatch problem.

The proposed subspace SID method is based on two consecutive orthogonal decompositions to identify a system in closed-loop. The subspace SID method furthermore guarantees the estimation of stable system matrices by utilizing the shift-invariant property of the extended observability matrix. Extended Kalman filtering is used to recursively update a joint set of initial system states and parameters, using current sampled process data and initial estimated parameters, obtained via the subspace method. To detect a fault in the updated parameter set, it is necessary to compare the updated parameter set, with the initial identified parameter set. This can be accomplished by considering appropriate matrix measures, which will accentuate any fundamental parameter matrix differences. The infinity matrix norm is proposed for detecting discrepancies between the absolute difference of the initial identified, and recently updated parameter set. The infinity matrix norm is used to reduce (averaging) the matrix of differences to one value which is monitored against a threshold value.

Diagnosis of oscillation due to multiples sources using wavelet transforms

Selvanathan Sivalingam and Morten Hovd

Abstract— **Control loop oscillations are a common type of plant-wide disturbance and the root-causes can be one or more among poorly tuned controllers, process or actuator non-linearities, presence of model plant mismatch and oscillatory disturbances. The oscillations are the most prominent indications of deteriorated controller performance. This article addresses the detection and diagnosis of oscillations in measurements due to multiple sources under the framework of internal model control. A pattern recognition based approach using cross wavelet transforms is proposed to pinpoint the source(s) of oscillation in the control loop. The phase information in wavelet domain between input and output signals is exploited to diagnose the source of oscillation.**

I. INTRODUCTION

It is well known that performance degradation in control loops manifest as one or more of the following: (*i*) poor set point (SP) tracking (*ii*) oscillations (*iii*) poor disturbance rejection and (*iv*) excessive final control element variation. Industrial surveys over the last decade indicate that only about one-third of industrial controllers provide acceptable performance and about 30% [1], [2], [3] of the control loops exhibit oscillation. Since oscillations can lead to loss of energy, isolating the root cause for oscillations is important for improving the performance of the oscillating control loops. Also, the presence of the oscillations in a plant increases the variability of the process variables and thus may cause loss of product quality. The oscillations (or vibrations), in general, are a very drastic form of plant performance degradation, which can, in many cases, be induced by the feedback mechanism itself.

Oscillations are attributed to one or more among poor controller tuning, process or actuator non-linearities, presence of model plant mismatch or oscillatory disturbances. A tool to help the engineer should therefore automatically bring oscillatory loops to his or her attention, characterize them and highlight the presence of plant wide oscillations. Several authors have addressed the detection of oscillatory measurements in process data. Early works appear in [4] followed by [5], [6], [7], [8]. [4] proposed a technique to detect oscillating loops "on-line" using the IAE criterion. This method does not assume any particular shape for oscillation and only requires the measurement to deviate significantly from the set point. [4] also proposed a diagnostic procedure for finding the source of oscillation and eliminating it. The diagnostic procedure is carried out by disconnecting the feedback (*i.e.* switching the controller to manual mode). This approach is simple and efficient and probably the most comprehensive procedure available for diagnosing root cause for oscillations. However, switching the controller to manual mode may not always be allowed, especially if the loop is deemed critical. Further, it will not be possible to apply this approach on thousands of loops in a routine fashion. [5] presented an offline technique for detecting oscillation using a regularity factor. This method requires the user to specify the root-mean-square value of the noise and a thresholds a nontrivial task when applied to hundreds of loops.

[5] and [9] proposed a set of procedures to detect and diagnose oscillating loops using offline data. They combine the techniques of controller performance assessment along with operational signatures (OP-PV plots) and spectral analysis of the controller error for diagnosis. This technique, though not completely automated, can distinguish the cause of oscillation as one of the following: (*i*) poor tuning (*ii*) nonlinearity or (*iii*) external disturbance. However, the downside lies in manually inferring the loop signatures that are based on spectral analysis or on a map of controller output (OP) versus process variable (PV) and isolating the oscillating portion from the entire data. [10] presented a simple, practical approach to distinguish oscillating loops that are caused by external disturbances and static friction. This approach is based on cross-correlation between the controller output (OP) and process output (PV). The cross-correlation technique failed when the data had intermittent oscillations and the set-point was also changing. [11] also proposed a technique to identify stiction using nonlinear filters. The method assumed that information such as mass of stem, diaphragm area, and so on for each valve is readily available. Since in a typical process industry facility there can be thousands of control loops, it may be nearly impossible to build/maintain a knowledge base of control valves, making this technique difficult to implement.

[12] used higher order statistics for detecting nonlinearity in data and have extended the method for diagnosing stiction by fitting an ellipse of the OP-PV plot and inferring the stiction from an assumed stiction model. The success of this approach lies in correctly identifying the oscillation period and its start and end point in the OP-PV data. [8] proposed non-negative matrix factorization for detection and diagnosis of plant-wide oscillations based on source separation techniques. As can be seen, the task of detecting stiction or other nonlinearities in valves from routine operating data is a challenging task. To summarize, data driven techniques

Selvanathan Sivalingam and Morten Hovd are with Department of Engineering Cybernetics, Norwegian University of Science and Technology, Norway 7491. Email addresses: selvanathan.sivalingam@itk.ntnu.no and morten.hovd@itk.ntnu.no

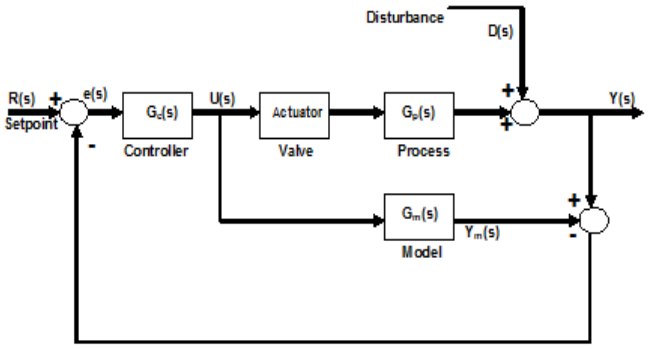


Fig. 1. Schematic representation of internal model control with actuator

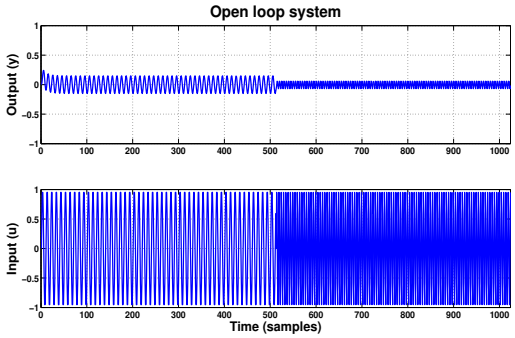


Fig. 2. Time domain behavior of input and output signals considered for interpretation of wavelet analysis

that are presented in the literature till date are useful in (a) assessing the performance of the controller by calculating a figure of merits given that the cause of poor-performance is only due to either an aggressive or sluggishly tuned controller in pure feedback control, (b) detecting oscillating loops with an user-specified parameter, and (c) limited diagnosis of the cause of oscillation based on cross-correlation, power spectral analysis, or OP-PV plots. The current approaches lack (a) the capability to efficiently diagnose oscillations due to multiple sources, (b) the ability to diagnose the causes of time-varying oscillations and (c) an automated means of oscillation diagnosis.

In this work, we have attempted to address some of the aforementioned drawbacks by using wavelet and cross wavelet transforms.

II. Problem Statement

Oscillations in model based control loops occur due to either one of (*i*) valve stiction (*ii*) model plant mismatch, (*iii*) external oscillatory disturbances or combination of any of these. It becomes vital to diagnose the causes of oscillations in order to take the appropriate remedial action. A procedure based on pattern recognition techniques using cross wavelet transform is devised in this article to diagnose the cause(s) of the oscillation. The problem is setup in the internal model control (IMC) framework (Figure 1).

III. Proposed Methodology

Cross wavelet transform of input and plant and that of input and model output are computed and thereby a specific pattern is sought for root cause diagnosis of oscillation using the direction of wavelet phase difference between the variables.

To illustrate the idea of cross-wavelet transform for an input-output system, an open-loop process with $G_p(s) = \frac{1}{10s+1}$ is considered. The process is simulated for a sinusoidal input having two frequencies and the time domain plots of input and output are given in Figure 2. The cross wavelet transform plot between two quantities u and y is shown in Figure 3.

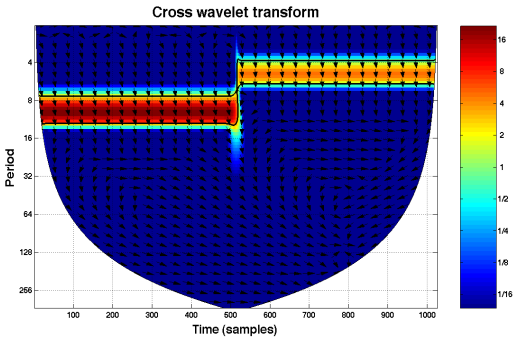


Fig. 3. Cross wavelet transform between input and output signals

It is known from Figure 3 that the quantities u and y show high common power at two frequencies between two different time intervals (0.1 Hz, 0-511 and 0.2 Hz, 512-1024) and the arrows indicate the direction of the wavelet phase between u and y *i.e.,* u leads y by 90° (pointing down). Based on the properties of cross wavelet transform, wavelet phase difference and linear time invariant systems theory, the following methodology is proposed to diagnose the source(s) of oscillation in a control loop.

The quantities controller output (u), process output (y) and model output (y_m) of an oscillating control loop are obtained either from simulation or from industry. The cross wavelet transforms, $W_{uy}(f,\tau)$ and $W_{uy_m}(f,\tau)$ are computed. By comparing the direction of wavelet phase, the following conclusions can be drawn.

- If the oscillating source is only due to valve stiction, the cross wavelet transform plots should not only exhibit harmonics but also discontinuities.
- If the source is due to gain mismatch, the plots $W_{uy}(f,\tau)$ and $W_{uy_m}(f,\tau)$ of should be identical since the phase spectrum is independent of any changes in gain. The arrows in the plots of $W_{uy}(f,\tau)$ and $W_{uy_m}(f,\tau)$ will be in same direction.
- If the source is due to delay mismatch, the plots of $W_{uy}(f,\tau)$ and $W_{uy_m}(f,\tau)$ will vary in phase direction since the phase spectrum depends on delay changes.

IV. Simulations

A control system consisting of a process characterized by the transfer function $G_p = \frac{K_p}{\tau_p s + 1} e^{-d_p s}$ and model $G_m = \frac{K_m}{\tau_m s + 1} e^{-d_m s}$ is simulated with IMC controller for a unit step change in the set point. The different case studies analyzed for the diagnosis of oscillation in a control loop are (*i*) oscillation due to valve stiction (*ii*) oscillation due to valve stiction and oscillatory disturbance (*iii*) oscillation due to gain mismatch (*iv*) oscillation due to gain mismatch and oscillatory disturbance and (v) oscillation due to delay mismatch.

A simple yet efficient one parameter model proposed by [4] is used to generate oscillations due to valve stiction. The model is

$$x(t) = \begin{cases} x(t-1) & |u(t) - x(t-1)| \leq d \\ u(t) & \text{otherwise} \end{cases} \quad (1)$$

Here $u(t)$ and $x(t-1)$ are present and past valve outputs, $u(t)$ is the present controller output, and d is the valve stiction band. The valve stiction band is expressed in terms of the percentage or fraction of valve movement corresponding to the amount of stiction present in the valve. For instance, if 100 units of force are required to open the valve completely from completely closed position and 10 units of force is required to overcome the amount of static friction in the valve, stiction band is 10% or 0.1. The stiction band of 0.1 is used in the simulation. Model plant mismatch is introduced by changing the values of gain or delay appropriately in the process. The sinusoidal disturbance of frequency 0.01 Hz is considered for the simulation. The time domain plots of controller output (u), plant output (y) and model output (y_m) for different simulation studies is shown in Figures 4, 5, 6, 7 and 8.

The cross wavelet transform computed between controller output and plant output is compared with that computed between controller output and model output. In the case of oscillation due to valve stiction (Figure 4), the plots of cross wavelet transform (Figures 9 & 10) not only show harmonics but also discontinuities which are the characteristics of a sticky valve. Figures 11 and 16 clearly indicate the presence of the valve stiction as one of the sources of oscillation between 800 and 1600 s and the other being the oscillatory component of frequency 0.01 throughout.

If the oscillation is only due to MPM, there will be clearly a single frequency in the cross wavelet transform plot. In the case of gain mismatch (Figure 6), the plots of cross wavelet transform between controller output and plant output and controller output and model output (Figures 13&14) produce identical plots since the phase spectrum is independent of the changes in gain.

The control loop whose time domain trends are characterized by Figure 7 is diagnosed to have gain mismatch as one of the sources of oscillation between 800 and 1600 s and other being the oscillatory component of frequency 0.01 Hz

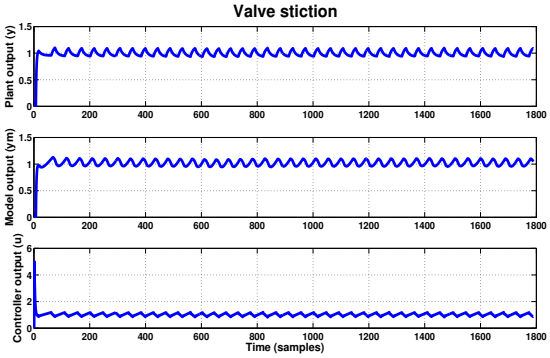


Fig. 4. *Time domain behavior of plant, model and controller outputs for the valve stiction as the source of oscillation.*

(Figures 15 & 16). The presence of oscillatory component can not be due to the presence of delay mismatch since the presence of gain and delay mismatch at a same time will lead to system instability. Hence, the loop can be said to have the external oscillatory disturbance over the entire period and gain mismatch between 800 and 1600 s as the sources of oscillation.

The control loop whose outputs are given in Figure 8 is analyzed for diagnosing the source(s) of oscillations. Figures 17 and 18 indicate the presence of a single frequency component and a directional change in the phase difference. The source of oscillation can be either oscillatory disturbance or delay mismatch.

V. Conclusions

A pattern recognition technique for the diagnosis of control loop oscillations in internal model control systems due to multiple sources using cross wavelet transform of two quantities has been developed. A diagnostic study of oscillation due to either one of valve stiction, model plant mismatch, oscillatory disturbance or combination of these has been presented. The oscillations due to valve stiction manifest as harmonics as well as discontinuities in the cross wavelet transform plots whereas oscillation due to model plant mismatch leaves distinct signatures in the phase information (arrows). If the oscillations are due to gain mismatch, no change is observed in the phase spectrum computed between controller output and plant output and controller output and model output. On the other hand, oscillation due to delay mismatch or oscillatory disturbance results in a directional change in the phase difference computed between controller output and plant output and controller output and model output.

Further study to distinguish between the oscillatory disturbance and delay mismatch as sources of the oscillation is currently underway. In a parallel study, it is found in our preliminary extensions of this work that the results demonstrated here can be applied to a more generalized problem of the diagnosis of poor control loop performance.

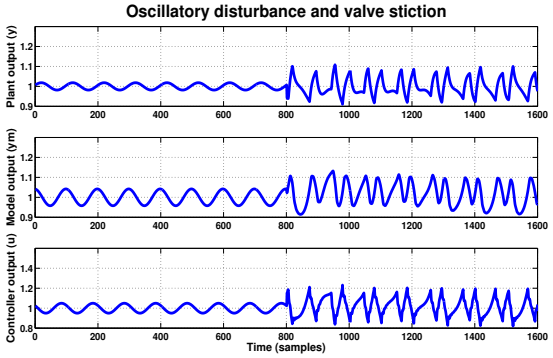


Fig. 5. *Time domain behavior of plant, model and controller outputs for the case oscillatory output and valve stiction as the sources of oscillation*

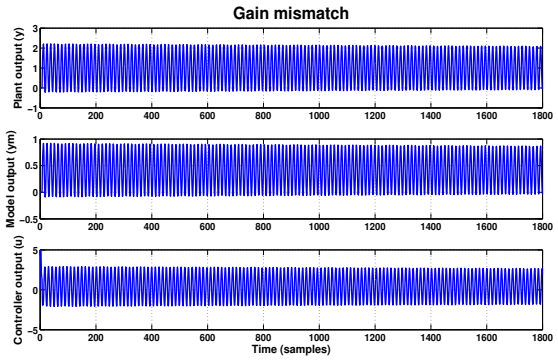


Fig. 6. *Time domain behavior of plant, model and controller outputs for the case of gain mismatch as the source of oscillation.*

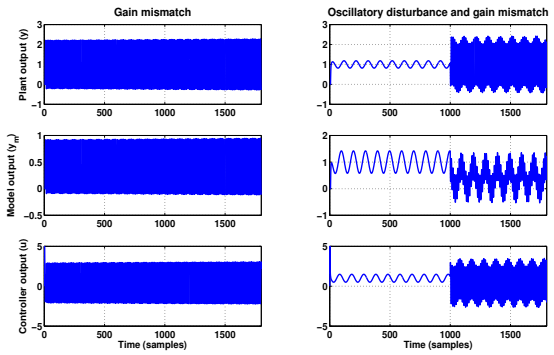


Fig. 7. *Time domain behavior of plant, model and controller outputs for the case of oscillatory disturbance and gain mismatch as the sources of oscillation.*

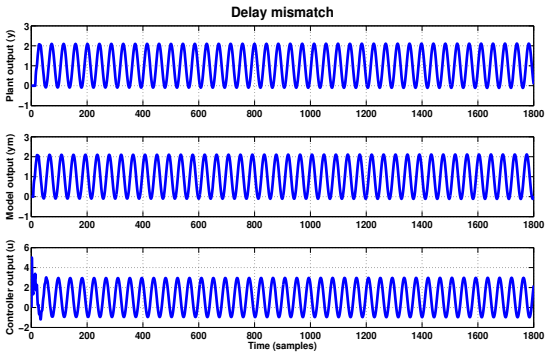


Fig. 8. *Time domain plots of plant, model and controller outputs for the case of delay mismatch as the source of oscillation.*

REFERENCES

[1] W. Bialkowski, "Dreams versus reality: a view from both sides of the gap," *Pulp and Paper Canada*, vol. 94, pp. 19–27, 1993.

[2] L. Desborough and R. Miller, "Increasing customer value of industrial control performance monitoring: Honeywell's experience," *Proc. AIChE Symp. Ser.*, vol. 98, pp. 153–186, 2002.

[3] D. Ender, "Process control performance: not as good as you think," *Control Engineering*, vol. 40, pp. 180–190, 1993.

[4] T. Hägglund, "A control-loop performance monitor," *Control Engineering Practice*, vol. 3, pp. 1543–1551, 1995.

[5] N. Thornhill and T. Hägglund, "Detection and diagnosis of oscillation in control loops," *Control Engineering Practice*, vol. 5, pp. 1343–1354, 1997.

[6] K. Forsman and A. Stattin, "A new criterion for detecting oscillations in control loops," in *CP8-3*. Karlsruhe, Germany.: European control conference, 1999.

[7] R. Rengaswamy, T. Hägglund, and V. Venkatasubramanian, "A qualitative shape analysis formalism for monitoring control loop performance," *Engineering Applications of Artificial Intelligence*, vol. 14, pp. 23–33, 2001.

[8] A. Tangirala, J. Kanodia, and S. Shah, "Non-negative matrix factorization for detection and diagnosis of plant wide oscillations," *Industrial and Engineering Chemistry Research*, vol. 46, pp. 801–817, 2007.

[9] N. Thornhill, B. Huang, and H. Zhang, "Detection of multiple oscillations in control loops," *Journal of Process Control*, vol. 13, pp. 91–100, 2003.

[10] A. Horch, "A simple method for the detection of stiction in control valves," *Control Engineering Practice*, vol. 7, pp. 1221–1231, 1999.

[11] A. Horch and A. Isaksson, "A method for detection of stiction in control valves," in *On-line-fault detection and supervision in the chemical process industry*. Lyon, France: IFAC Workshop, 1998, p. 4B.

[12] M. Choudhury, S. Shah, and N. Thornhill, "Detection and quantification of control valve stiction." Boston, USA: DYCOPS, 2004.

[13] D. Bloomfield, R. McAteer, B. Lites, and P. Judge, "Wavelet phase coherence analysis: Application to a quiet-sun magnetic element," *The Astrophysical Journal*, vol. 617, pp. 623–632, 2004.

[14] C. Torrence and G. Compo, "A practical guide to wavelet analysis," *Bulletin of the American Meteorological Society*, vol. 79, pp. 61–78, 1998.

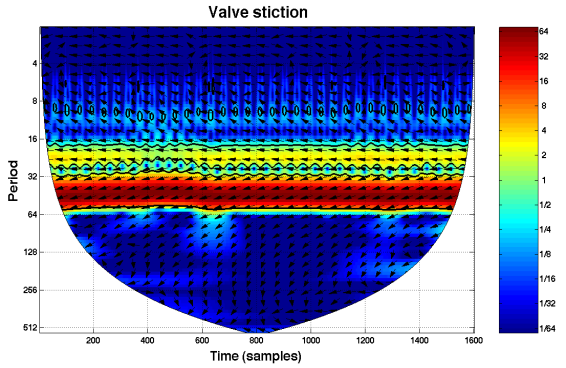


Fig. 9. *Cross wavelet transform plot between u and y_p when the oscillation is only due to valve stiction.*

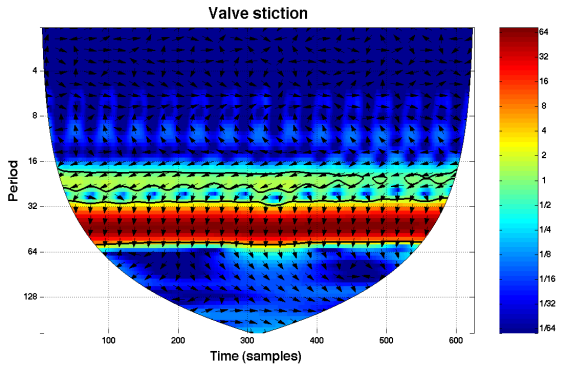


Fig. 10. *Cross wavelet transform plot between u and y_m when the oscillation is only due to valve stiction.*

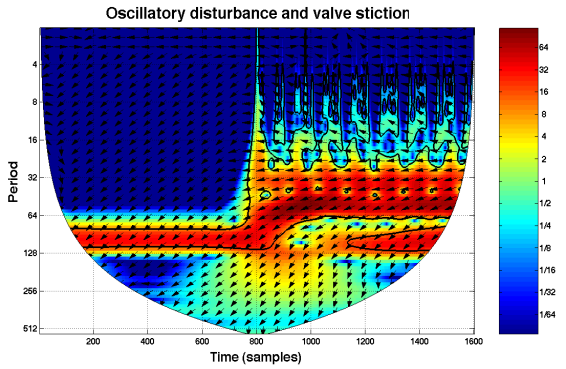


Fig. 11. *Cross wavelet transform plot between u and y_p when the oscillation is due to oscillatory disturbance and valve stiction.*

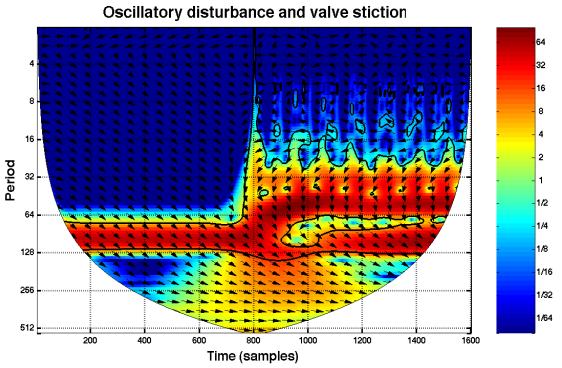


Fig. 12. *Cross wavelet transform plot between u and y_m when the oscillation is due to oscillatory disturbance and valve stiction.*

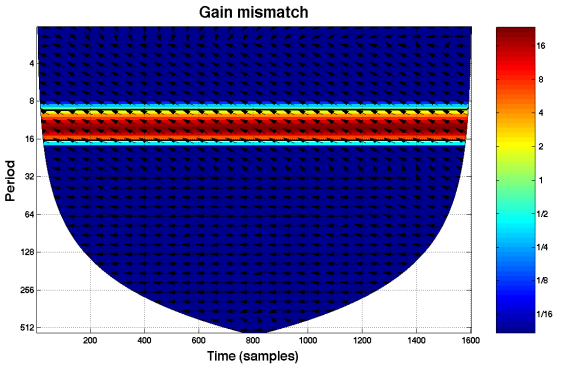


Fig. 13. *Cross wavelet transform plot between u and y_p when the oscillation is gain mismatch*

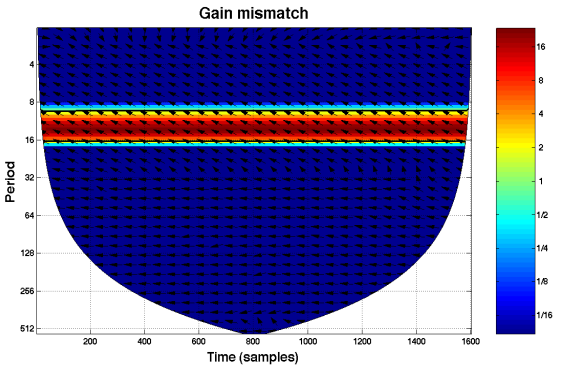


Fig. 14. *Cross wavelet transform plot between u and y_m when the oscillation is gain mismatch*

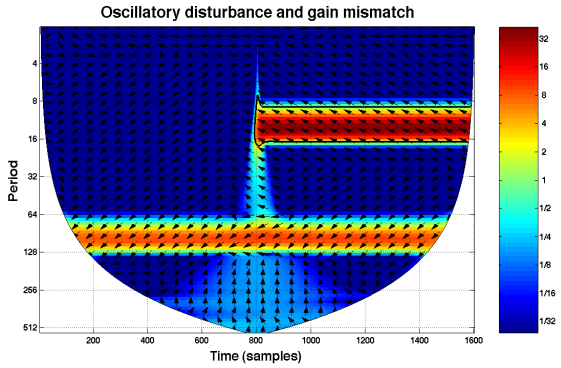


Fig. 15. *Cross wavelet transform plot between u and y_p when the oscillation is due to oscillatory disturbance and gain mismatch.*

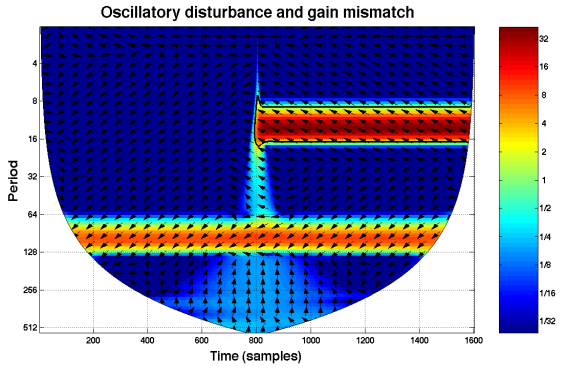


Fig. 16. *Cross wavelet transform plot between u and y_m when the oscillation is due to oscillatory disturbance and gain mismatch..*

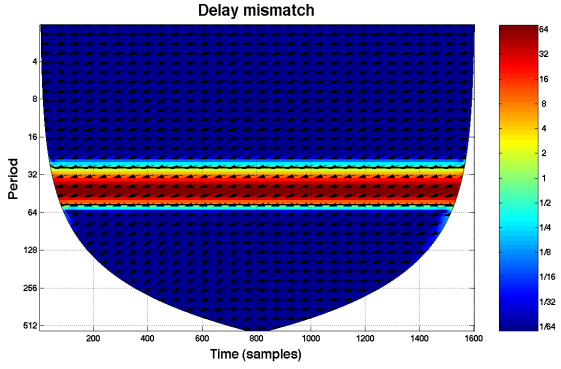


Fig. 17. *Cross wavelet transform plot between u and y_p when the oscillation is due delay mismatch..*

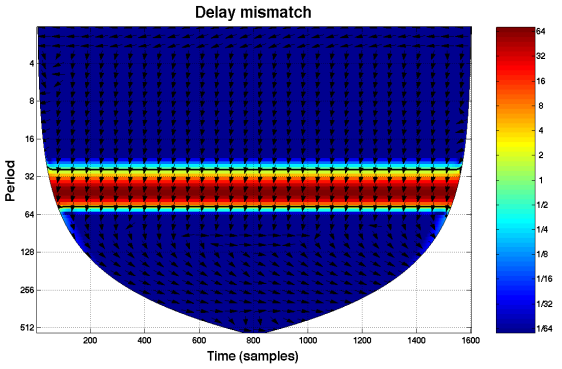


Fig. 18. *Cross wavelet transform plot between u and y_m when the oscillation is due delay mismatch..*

Availability Estimations for Utilities in the Process Industry

Anna Lindholm
Automatic Control, Lund

Hampus Carlsson
Perstorp AB

Charlotta Johnsson
Automatic Control, Lund

I. Introduction

An important performance measure of a plant is the plant-availability. The higher availability the better, since a high availability implies a possibility for a large production volume and thereby an increased profit for the company. One way of increasing the plant-availability is by eliminating, or minimizing the effect of disturbances. The cause of a disturbance can be personnel, material or equipment, where material includes both raw materials and utilities.

The aim of this work is to increase the plant-availability by decreasing the effects of plant-wide disturbances caused by utilities. The first step is to determine the set of utilities that can be present at an industrial site, what disturbances these utilities can suffer, and how frequent and safety-critical these disturbances are. A later step will be to determine the effects on the plant-availability, and ways to decrease or eliminate these effects.

The research is performed within the framework of PIC-LU (Process Industrial Centre at Lund University) supported by the Foundation of Strategic Research (SSF).

II. Utilities

Utilities, in opposite to raw materials, are materials that are used plant-wide and are crucial for plant operation but are not part of the final product. Common utilities are

- **Steam**: The steam net is commonly used to supply energy for distillation. Other uses are to supply energy for endothermic reactions and to heat a reactor at start-up. There could be several steam nets at the same site, for example one net with high pressure steam and one with low pressure steam.
- **Cooling water**: The cooling water system is used for cooling at exothermic reactions and in the condensing phase of distillation.
- **Electricity**: Electricity is needed in order for the instruments, e.g. pumps, to operate.
- **Water treatment**: A Water treatment utility is used for purification of process water, precipitation and ground water.
- **Combustion of tail gas**: A system for combustion of tail gas, such as a flare, is a safety device needed at unforeseen events.
- **Nitrogen**: Nitrogen is needed to maintain pressure in vessels.
- **Feed water**: Feed water is used in boilers to produce steam.
- **Instrument air**: Instrument air is needed for the pneumatic instruments to work.
- **Vacuum system**: Vacuum is used to lower the boiling point of a liquid to facilitate distillation and to remove gas produced in reactions.

A flowchart can be made for each of the utilities, showing how the utility flows through the areas of the production site. An example is showed in Figure 1.

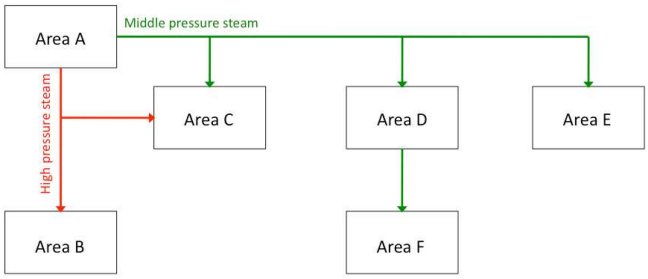


Fig. 1. Example of a utility flowchart for steam. The site contains 6 areas, A-F, and has two steam nets producing high and low pressure steam respectively.

A utility could suffer from different disturbances. For example, a steam net could suffer disturbances such as too high or too low steam pressure. One way of defining when a disturbance on a utility occurs is to set limits, such that if the parameter goes outside this limits, the disturbance will have economical or safety consequences.

III. Availability

The availability of a production unit is according to ISO22400, draft 1, the fraction of the main usage time, which is the producing time of the unit, and the planned busy time, which is the time that the production unit is used for the execution of a manufacturing order. The availability of a utility can be estimated by taking the fraction of time when the utility does not suffer a disturbance over the total time. If measurements are available of the key parameters that define the disturbance, the availability can be computed directly from historical data.

IV. Future Work

When availabilities for the different utilities have been computed, the consequences of each disturbance must be evaluated. When both frequency and severity of all disturbances on utilities are known, focus on handling the disturbance with highest severity$\times$frequency-factor to improve availability of the entire production plant as much as possible.

Detection and Isolation of Oscillations Using the Dynamic Causal Digraph Method

Tikkala, Vesa-Matti*; Zakharov, Alexey; Jämsä-Jounela, Sirkka-Liisa
Aalto University School of Science and Technology
Department of Biotechnology and Chemical Technology

Abstract

This paper proposes a modification to the dynamic causal digraph (DCDG) method in order to address the detection and isolation of oscillations in a process. The proposed detection method takes advantage of the properties of residual signals generated by the DCDG method by studying their zero-crossings. The method is tested in an application to a board making process and the results are presented and discussed.

1 Introduction

Demands to keep industrial processes running efficiently with a high rate of utilization are increasing constantly due to the tightening global competition. Since, the modern industrial processes are complex and large-scale, operator-based monitoring cannot guarantee early enough detection and reliable diagnosis of the faults and abnormalities. Therefore, the detection and diagnosis of different abnormal and faulty conditions in the processes have become increasingly important.

Common problems causing inefficient operation and production losses in the process industry are oscillations. Oscillatory disturbances readily propagate in the process and cause extensive variation in the process variables. The oscillations are usually originated under feedback control, and they may have various causes which have been categorized by Thornhill & Horch (2007) into non-linear and linear causes. Non-linear causes include for example extensive static friction in the control valves, on-off or split range control, sensor faults, process non-linearities and hydrodynamic instabilities. The most common linear causes are poor controller tuning, controller interaction and structural problems involving process recycles (Thornhill & Horch, 2007). According to Choudhury *et al.* (2008), valve stiction is, however, the most common cause of these oscillations in control loops.

The detection and diagnosis of oscillations have been previously addressed by data-based methods which study, for example, the properties of controller error signals (Thornhill & Hagglund, 1997; Forsman & Stattin, 1999), the spectral properties (Thornhill *et al.*, 2003) or the nonlinearity of the measurement signals (Choudhury *et al.*, 2004; Thornhill, 2005). Also, a variety of multivariate methods, such as principal component analysis (Thornhill *et al.*, 2002) and non-negative matrix factorization (Tangirala *et al.*, 2007), have been applied to solve this diagnosis task. A recent trend has been to introduce process information into the diagnosis of plant-wide oscillations. Applications in which the process connectivity information has been integrated into data-based analyses to check hypotheses on the fault origin, have been presented (Yim *et al.*, 2007; Jiang *et al.*, 2009).

This paper aims at further development of the dynamic causal digraph (DCDG) method by addressing the detection and isolation of plant-wide oscillations. A detection algorithm, which is able to deal with oscillatory residuals, is proposed and integrated into the DCDG method. In this paper, the modified DCDG method is used to detect and isolate low-frequency oscillations caused by a valve stiction fault in a board machine process.

The paper is organized as follows. In Section 2, the dynamic causal digraph method and the new detection algorithm are introduced. The process and the test environment are described in Section 3. The results of the testing are presented in Section 4 followed by the conclusions in Section 5.

2 Enhanced DCDG Method for the Detection and Isolation of Oscillations

The dynamic causal digraph method employs the process knowledge formalized as a causal digraph model in order to perform the ordinary fault diagnosis tasks as presented by Cheng *et al.* (2010); Cheng (2009). In the enhanced DCDG, the detection of faults is performed using the proposed method which observes the zero-crossings in the residuals generated by a comparison

*Corresponding author: vesa-matti.tikkala@tkk.fi, Aalto university, PL 16100, FI-00076 Aalto

of cause-effect models and the process measurements. Next, the isolation is carried out by applying a set of inference rules to the residuals in order to extract the fault propagation path. Finally the arcs in the digraph that explain the faulty behavior are identified. The enhanced DCDG method is described in more detail in the following.

2.1 Fault detection

Fault detection is performed in two steps: residual generation and fault detection in the residuals using the modified cumulative sum (CUSUM) algorithm.

2.1.1 Residual generation with dynamic models

The dynamic causal digraph produces two kinds of residual to be used in fault detection and isolation: global (GR) and local residuals (LR). The global residual is produced from the difference between the measurement and the global propagation value:

$$GR(Y) = Y(k) - \hat{Y}(k), \tag{1}$$

where $Y(k)$ is the measurement and $\hat{Y}(k)$ is the global propagation value obtained by

$$\hat{Y}(k) = f_Y\left(\hat{U}(k-1), \hat{U}(k-2), \ldots\right), \tag{2}$$

where f_Y is a discrete-time model describing the cause-effect relationship from n predecessor nodes U_i to node Y. $\hat{U}(k-\tau) = \{\hat{u}_1(k-\tau), \ldots, \hat{u}_n(k-\tau)\}$ are the lagged global propagation values from the predecessors with time lags $\tau = 1, 2, \ldots$ depending on the system order.

The local residuals are subcategorized into three types: individual local residuals (ILR), multiple local residuals (MLR) and total local residuals (TLR) (Montmain & Gentil, 2000).

The individual local residual is produced by taking the difference between the measurement and the local propagation value with only one measured input, while all the others are propagation values from the parent nodes:

$$\begin{aligned} ILR_Y^m &= Y - \bar{Y}, \\ \bar{Y}(k) &= f_Y\left(\bar{U}(m, k-1), \bar{U}(m, k-2), \ldots\right), \end{aligned} \tag{3}$$

where

$$\begin{aligned} \bar{U}(m, k-\tau) &= \left\{ \bar{u}_i(k-\tau) \middle| \right. \\ \bar{u}_i(k-\tau) &= \left. \begin{cases} \hat{u}_i(k-\tau), i \neq m \\ u_i(k-\tau), i = m \end{cases} , 1 \leq i \leq n \right\}, \end{aligned} \tag{4}$$

$\hat{u}_i(k)$ is the lagged global propagated value from the predecessors, and $u_i(k-\tau)$ is the measurement for the i-th parent node.

Similarly, the $MLR_Y^{P_Y^l}$ is produced as

$$\begin{aligned} MLR_Y^{P_Y^l} &= Y - \bar{Y}, \\ \bar{Y}(k) &= f_Y\left(\bar{U}(P_Y^l, k-1), \bar{U}(P_Y^l, k-2), \ldots\right), \end{aligned} \tag{5}$$

where

$$\begin{aligned} \bar{U}(P_Y^l, k-\tau) &= \left\{ \bar{u}_i(k-\tau) \middle| \right. \\ \bar{u}_i(k-\tau) &= \left. \begin{cases} \hat{u}_i(k-\tau), i \notin P_Y^l \\ u_i(k-\tau), i \in P_Y^l \end{cases} , 1 \leq i \leq n \right\}, \end{aligned} \tag{6}$$

P_Y^l is the set of indices of the predecessors which use the measurement as an input. The $TLR(Y)$ is produced with $P_Y^l = P_Y$, where P_Y is the set of indices of all the predecessors of Y.

The residual generation scheme follows the DCDG method developed in (Montmain & Gentil, 2000).

2.1.2 Fault detection using the modified CUSUM method

The proposed detection method utilizes the cumulative sum (CUSUM) method presented by Hinkley (1971), by applying it to the detection of a change in the mean and variance of the zero-crossings in the residual signals. The CUSUM algorithm is defined for a positive change as follows:

$$\begin{aligned} U_0 &= 0 \\ U_n &= \sum_{k=1}^{n} d(k) - \mu_0 - \frac{\beta}{2}, \\ m_n &= \min_{0 \leq k \leq n} U_k, \end{aligned} \tag{7}$$

where β is a user-specified minimum detectable change, $d(k)$ the observed signal with nominal mean value equal to μ_0. Whenever $U_n - m_n > \lambda$, a change is detected, where λ is a design parameter, usually tuned according to the requirements for the false alarm and missed alarm rates.

The signal observed by the CUSUM algorithm, called the detection signal, is defined as follows

$$d(k) = \max\left\{ \bar{\Delta}t(k), \frac{\sigma^2_{\Delta t}(k)}{\bar{\Delta}t(k)} \right\}, \tag{8}$$

where $\max\{\cdot\}$-operator takes the maximum of its arguments, $\bar{\Delta}t(k)$ and $\sigma^2_{\Delta t}(k)$ are the mean and the variance of the time between consecutive zero-crossings in a residual, respectively. Both $\bar{\Delta}t$ and $\sigma^2_{\Delta t}$ are calculated in a moving window of length l: $[e(k-l), e(k)]$, where $e(k)$ is the residual.

In normal operation, when the residuals are assumed to be zero-mean Gaussian noise, $\bar{\Delta}t, \sigma^2_{\Delta t}(k) \approx 2$, since

Table 1: Fault isolation rules of the dynamic causal digraph

$CU(GR(Y))$	$CU(TLR(Y))$	$CU(ILR_Y(m))$	$CU(ILR_Y(i))$	$CU(MLR_Y(P_1))$	$CU(MLR_Y(P_2))$	Decision
0	0	0	0	0	0	No fault
1/-1	0	0^*	$1/\text{-}1^*$	0^*	$1/\text{-}1^*$	Fault propagates from the parent node m
1/-1	0	$1/\text{-}1^{**}$	$1/\text{-}1^{**}$	$1/\text{-}1^{**}$	0^{**}	Fault propagates from the nodes with subscript P_2
1/-1	1/-1	1/-1	1/-1	1/-1	1/-1	Local fault on variable Y

* $\forall i \neq m, i \in P_Y, m \in P_1, m \notin P_2, P_Y$ is the set of subscripts of parent nodes of the node Y.

** $\forall i, m, i \in P_Y, m \in P_Y, \forall P_1, P_2 \subseteq P_Y$.

the probability of $e(t)$ having a different sign than $e(t-1)$ is 0.5 for all t. Therefore, the nominal mean value of the observed signal in (7) can be set as $\mu_0 = 2$ and β and λ are then tuned to obtain robust detection with minimal false alarms. The window length l must be selected to be larger than one half of the expected period in the residual.

2.2 Fault isolation

2.2.1 Isolation of the fault propagation path

Fault isolation is performed recursively for the detected nodes by using a set of rules. These isolation rules, developed by Montmain & Gentil (2000), are converted into a table for the convenience of implementation, as shown in Table 1. After the isolation the nature of the fault is determined by using rules in Table 2.

Table 2: Fault nature rules of the dynamic causal digraph

$CU(GR(X))^*$	$CU(TLR(X))$	Fault nature
1/-1	1/-1	Local fault for that child node
1/-1	0	Process fault for the faulty node
0	1/-1	Measurement fault for the faulty node

* X is the subscript of any child node of the node Y.

2.2.2 Isolation of the faulty process component

In the case of a process fault, in addition to locating the fault on the variables (nodes), locating it on the arcs is also desirable. However, the MISO structure of the digraph causes problems by generating multiple possible results as $2^n - 1, n \geq 1$, where n is the number of input arcs of the fault origin node(s).

In order to decrease the number of possible results, an inference mechanism between the arcs proposed in (Cheng *et al.*, 2008) is used. The inference mechanism is based on an inter-arc knowledge matrix $\mathbf{M}$ defined for node U as follows

$$\mathbf{M}_U(i,j) = \begin{cases} 1, \text{ if inconsistency in arc } \langle U, i\rangle \\ \quad \text{causes inconsistency to } \langle U, j\rangle \\ 0, \text{ otherwise,} \end{cases} \tag{9}$$

where i and j refer to the matrix rows and columns, respectively.

Next, each set of suspected arcs is tested in order to determine whether the fault may be caused exactly by the current set of arcs. In order to do it the matrix $\mathbf{M}$ is multiplied with a vector representing the suspected arc set, which is defined as follows

$$\mathbf{sv}(i) = \begin{cases} 1, \text{ if } ARC(\mathbf{M}, i) \in S,\ 1 \leq i \leq N_a \\ 0, \text{ otherwise,} \end{cases} \tag{10}$$

where $ARC(\mathbf{M}, i)$ gives the arc corresponding to the ith row in the matrix $\mathbf{M}$. S is the set of suspected arcs. If the number of non-zero elements of $\mathbf{sv}$ have changed, the current suspected set of arcs must be excluded.

3 Description of the Process and the Valve Stiction Faults

This test focuses on the stock preparation of the board machine at Stora Enso's mills in Imatra, Finland. The simulation tests are run on a board machine simulator model in the APROS simulation environment.

3.1 The Board Machine Process

The board making process begins with the preparation of raw materials in the stock preparation section, as shown in the flowsheet in Figure 1. Different types of pulp are refined and blended according to a specific recipe in order to achieve the desired composition and properties for the board grade to be produced. The consistency of the stock is controlled with dilution water.

The blended stock passes from the stock preparation to the short circulation. First, the stock is diluted in the machine chest to the correct consistency for web formation. The diluted stock is then pumped with a fan pump, which is used to control the basis weight of the board, to cleaning and screening. Next, the stock passes to the head box, from where it is sprayed onto the wire in order to form a solid board web.

The excess water is first drained through the wire and later by pressing the board web between rollers in the

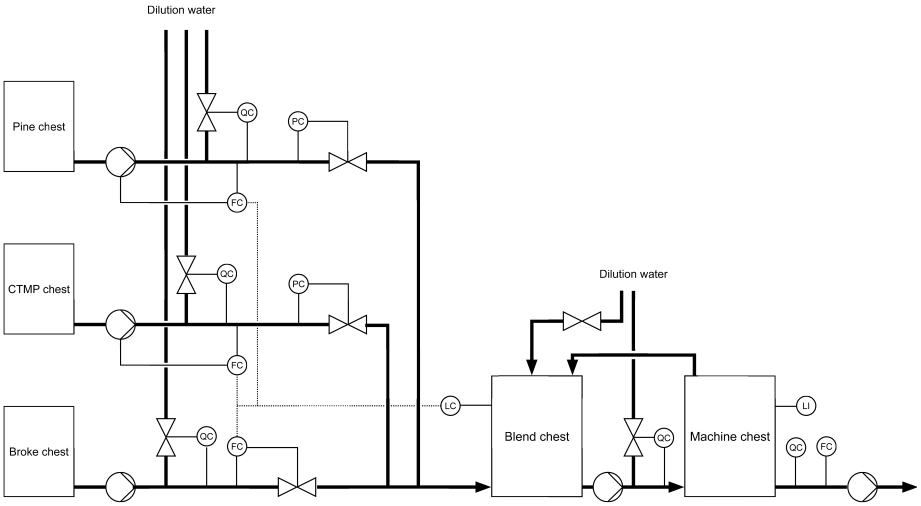


Figure 1: Flowsheet of the stock preparation of the board machine process.

Table 3: Variables of the causal digraph model for the stock preparation of the board machine.

Var.	Description	Type	Unit
vb	valve opening for the broke line	A	-
fb	mass flow of the broke	M	kg/s
vbd	dilution water valve opening for the broke line	A	-
fbd	dilution water flow for the broke line	E	kg/s
cb	broke consistency	M	%
rp	pine pump rotation speed	A	%
fp	mass flow of the pine stock	M	kg/s
vpd	dilution water valve opening for the pine line	A	-
cp	pine consistency	M	%
rc	CTMP pump rotation speed	A	%
fc	mass flow of the CTMP	M	kg/s
vcd	dilution water valve opening for the CTMP line	A	-
cc	CTMP consistency	M	%
$vmcd$	dilution water valve opening for the machine chest	A	-
cmc	consistency before the machine chest	M	%
pp	pressure before the pine valve	M	kg/s
pc	pressure before the CTMP valve	M	kg/s
ct	consistency of the machine chest	M	%

A: Actuator signal, M: Measurement signal

press section. The remaining water is evaporated off in the drying section using steam-heated drying rolls.

The variables used in the causal digraph model of the stock preparation are listed in Table 3.

3.2 Valve Stiction Faults

A control valve is the most common final control element used in the process industry (Choudhury *et al.*, 2008). Therefore, the diagnosis of faults in valves is of great importance. Stiction, short for static friction, is a problem in control valves since it can cause significant disturbances in the process variables. A valve suffering from excessive stiction sticks when the control signal, for example, changes the direction and does not move until the force required to move the valve shaft exceeds a certain limit. When the valve starts to move, it jumps and then follows the control signal before it sticks again. A sticking valve is likely to cause oscillations when it is involved in a control loop.

The stiction in valves has been modelled and studied e.g. by Stenman *et al.* (2003) and Choudhury *et al.* (2005). This paper considers a stiction fault in a pressure control valve which causes the control loop to oscillate and disturbs the operation of the plant.

4 Testing and Results

4.1 Simulation Environment and Fault Simulation

The Imatra board machine model was developed by Stora Enso and VTT in the APROS environment. It was originally constructed on the basis of modeling and simulation studies carried out during 1998–2002 for Stora Enso's Imatra mills. It has been previously used for grade change simulations and in studies reported by Lappalainen *et al.* (2003).

A valve stiction fault was simulated in the stock preparation part of the board machine using the APROS board machine model. The faulty valve is located in the CTMP line and is used to control the feeding pressure p_c of the blend chest. The two-parameter data-driven valve stiction model proposed by Choudhury *et al.* (2005) was implemented in the APROS simulation software for the simulation. The deadband and slip-jump parameters of the stiction model were set to $S = 0.06$ and $J = 0.06$ respectively. The fault was evoked by a step change to the setpoint of p_c.

The fault occurring at $t = 200$ causes an oscillation

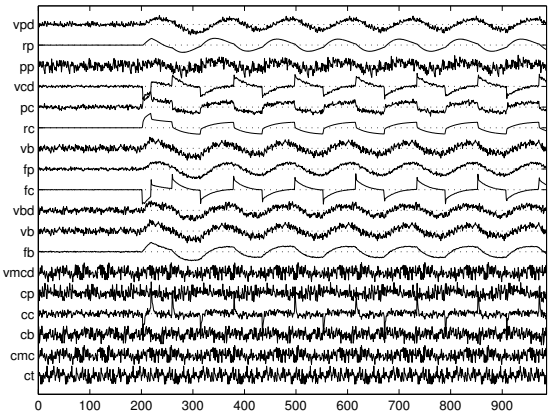


Figure 2: Normalized process variables during the fault simulation.

with a period of approximately 120 samples, which affects most of the variables in the stock preparation. Figure 2 shows the measured variables during the fault simulation and it demonstrates clearly the effect of the fault in the process.

4.2 Fault Detection and Isolation Results

First, the global residuals for all variables were produced by comparing the measured values of the variables and the estimates generated using the dynamic causal digraph model. Then, the detection signals were produced by calculating the mean and the variance of zero-crossings in the global residuals. The proposed detection method was applied to analyse the residuals in order to detect the faulty nodes. The parameters of the modified CUSUM method were set to the following: $\beta = 10$, $\lambda = 4$ and $l = 200$. The global residuals, detection signals and the detection results for variables f_c and c_c are presented in Figure 3. The fault is detected in both signals $GR(f_c)$ and $GR(c_c)$. The change in the detection signal $d_{f_c}(k)$ was detected for the first time at $k = 203$, three time instants after the fault occurred. However, the detection result is not reliable until $k = 270$. The global residual of c_c is detected later at $k = 315$.

Local residuals were generated in order to carry out the inference to isolate the origin of the fault. The local residual, the detection signal and the detection results of c_c are shown in Figure 4. The detection signal changes slightly after the fault occurrence, but no detection is however made.

The performance of the proposed detection method is satisfactory. The faults are detected with a reasonable delay and no false alarms are generated. Detection in variable c_{mc}, based on the structure of the process and the forecast of the fault propagation, was also expected.

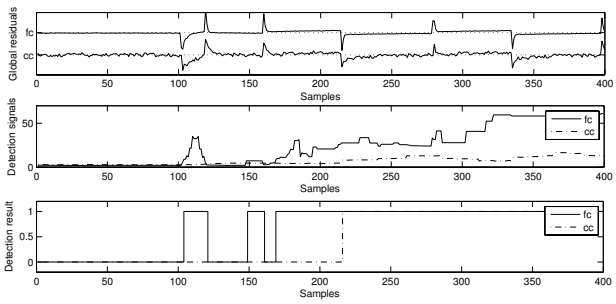


Figure 3: Global residuals $GR(f_c)$ and $GR(c_c)$, detection signals and the detection results

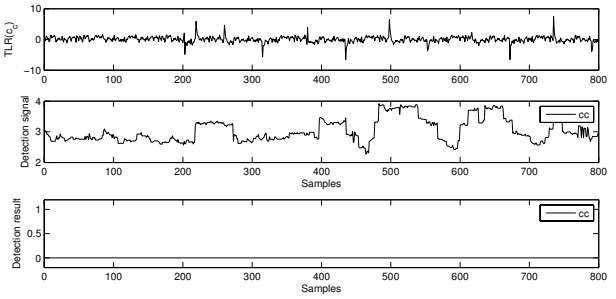


Figure 4: Local residual $TLR(c_c)$, detection signal and the detection results

However, based on the simulation studies, it was found out that the effect of the fault attenuates and therefore the change in the global residual of c_{mc} becomes undetectable.

The fault isolation rules presented in Table 1 were applied in order to extract the fault propagation path and the fault origin. The fault origin was located at the node f_c. The nature of the detected fault is diagnosed as a process fault according to the rules presented in Table 2.

Since the fault was a process fault, the structure of the digraph model gave three possible sources for the fault: v_{cd}, p_c and r_c resulting $(3^2 - 1) = 8$ possible sets of arcs explaining the faulty behaviour . The arc sets were analysed using the process knowledge matrix $\mathbf{M}$. However the number of the suspected sets could not be reduced in this case, since the input arcs to the node f_c are independent. If one input arc is faulty, it will not cause inconsistency in other input arcs.

5 Conclusions

A method for detecting oscillatory residual signals was presented in this paper. The method was integrated into the DCDG fault diagnosis method and tested in an application to a board making process.

The proposed method enables the detection and isolation of low-frequency oscillations caused by valve stiction faults in the process by exploiting the statistical properties of the residual signals. The results show that the proposed detection method is able to detect the fault successfully and to provide the information required for fault isolation.

The work presented in this paper represents the first step in addressing the detection and isolation of faults causing oscillatory behaviour in a process using the DCDG method. In future, the aim is to generalize the diagnosis methodology by developing new detection methods that are able to cover a wider range of faults occurring in industrial processes.

References

Cheng, H., 2009. *Causal Digraph Reasoning for Fault Diagnosis in Paper Making Applications*. Ph.D. thesis, Helsinki University of Technology, Helsinki, Finland.

Cheng, H., Nikus, M. & Jämsä-Jounela, S.-L., 2008. Fault diagnosis of the paper machine short circulation process using novel dynamic causal digraph reasoning. *Journal of Process Control*, vol. 18(7–8), pp. 676–691.

Cheng, H., Tikkala, V.-M., Zakharov, A., Myller, T. & Jamsa-Jounela, S.-L., 2010. Application of the enhanced dynamic causal digraph method on a three-layer board machine. *IEEE Transactions on Control System Technology*, (in press).

Choudhury, M. A. A. S., Shah, S. L. & Thornhill, N. F., 2004. Diagnosis of poor control loop performance using higher order statistics. *Automatica*, vol. 40, pp. 1719–1728.

Choudhury, M. A. A. S., Shah, S. L. & Thornhill, N. F., 2008. *Diagnosis of Process Nonlinearities and Valve Stiction*. 1st ed., Springer.

Choudhury, M. A. A. S., Thornhill, N. F. & Shah, S. L., 2005. Modelling valve stiction. *Control Engineering Practice*, vol. 13(5), pp. 641–658.

Forsman, K. & Stattin, A., 1999. A new criterion for detecting oscillations in control loops. In: *European Control Conferenre ECC'99*, p. 878, karlsruhe, Germany, August 31 - September 3, 1999.

Hinkley, D. V., 1971. Inference about the change-point from cumulative sum tests. *Biometrika*, vol. 58, pp. 509–623.

Jiang, H., Patwardhan, R. & Shah, S. L., 2009. Root cause diagnosis of plant-wide oscillations using the concept of adjacency matrix. *Journal of Process Control*, vol. 19(8), pp. 1347–1354.

Lappalainen, J., Vehviläinen, O., Juslin, K., Myller, T. & Tuuri, S., 2003. Enhancing grade changes using dynamic simulation. *TAPPI Journal*, vol. 2(12).

Montmain, J. & Gentil, S., 2000. Dynamic causal model diagnostic reasoning for online technical process supervision. *Automatica*, vol. 36, pp. 1137–1152.

Stenman, A., Gustafsson, F. & Forsman, K., 2003. A segementation-based method for detection of stiction in control valves. *International Journal of Adaptive Control and Signal Processing*, vol. 17(7–9), pp. 625–634.

Tangirala, A., Kanoda, J. & Shah, S. L., 2007. Nonnegative matrix factorization for detection and diagnosis of plantwide oscillations. *Industrial and Engineering Chemistry Research*, vol. 46, pp. 801–817.

Thornhill, N. F., 2005. Finding the source of nonlinearity in a process with plant-wide oscillation. *IEEE Transactions on Control Systems Technology*, vol. 13(3), pp. 434–443.

Thornhill, N. F. & Hagglund, T., 1997. Detection and diagnosis of oscillations in control loops. *Control Engineering Practice*, vol. 5(10), pp. 1343–1354.

Thornhill, N. F. & Horch, A., 2007. Advances and new directions in plant-wide disturbance detection and diagnosis. *Control Engineering Practice*, vol. 15, pp. 1196–1206.

Thornhill, N. F., Huang, B. & Zhang, H., 2003. Detection of multiple oscillations in control loops. *Journal of Process Control*, vol. 13, pp. 91–100.

Thornhill, N. F., Shah, S. L., Huang, B. & Vishnubhotla, A., 2002. Spectral principal component analysis of dynamic process data. *Control Engineering Practice*, vol. 10.

Yim, S. Y., Ananthakumar, H. G., Benabbas, L., Horch, A., Drath, R. & Thornhill, N. F., 2007. Using process topology in plant-wide control loop performance assessment. *Computers and Chemical Engineering*, vol. 31, pp. 86–99.

Optimal Control of the Oil Reservoir Water-flooding process

Eka Suwartadi, NTNU; Stein Krogstad, SINTEF ICT; Bjarne Foss, NTNU Norway

Abstract

In the second phase of oil recovery water flooding is a common way to sweep remaining oil in the reservoir. The process can be posed as a nonlinear optimization problem. This talk will address optimization of water flooding in the context of gradient based optimization. The gradient is computed using the adjoint method. In the optimization problem there will be constraints on both control inputs and state variables. The latter constraints are notoriously difficult to handle since they affect the efficiency of the adjoint method adversely. We propose some methods to mitigate this. Further, we present a second order adjoint method to avoid numerical problems which may arise in quasi Newton methods like the BFGS method.

State-Constrained Control Based on Linearization of the Hamilton-Jacobi-Bellman Equation

Torsten Wik, Per Rutqvist and Claes Breitholtz

***Abstract*— For continuous time state constrained stochastic control problems a method based on optimization is presented. The method applies to systems where the control signal and the disturbance both enters affinely, and it has one main tuning paramater, which determines the control activity. If the disturbance covariance is unknown, it can also be used as a tuning parameter (matrix) to adjust the control directions in an intuitive way. Optimal control problems for this type of systems result in Hamilton Jacobi Bellman (HJB) equations that are problematic to solve because of nonlinearity and infinite boundary conditions. However, by applying a logarithmic transformation we show how and when the HJB equation can be transformed into a linear eigenvalue problem for which there are sometimes analytical solutions and if not, it can readily be solved with standard numerical methods. Sufficient and necessary conditions for when the method can be applied are derived, and their physical interpretation is discussed. A MIMO buffer control problem is used as an illustration.**

I. INTRODUCTION

Consider a dynamical system described by

$$\dot{x} = f(x) + B(x)u + G(x)w, \tag{1}$$

where $x \in \Omega \subset \mathbb{R}^n$ is the system state, $u \in \mathbb{R}^m$ is the control signal, $f : \Omega \to \mathbb{R}^n$, $B : \Omega \to \mathbb{R}^{n\times m}$ and $G : \Omega \to \mathbb{R}^{n\times n_w}$ are functions that describe the system dynamics, and $w = \dot{\omega} \in \mathbb{R}^{n_w}$ is a Gaussian white noise (where ω is a Wiener process) having the covariance matrix W. The boundary $\partial\Omega$ of the state space Ω defines the state constraints, i.e. the system must be controlled such that x never leaves Ω.

The idea now is to formulate the control problem as an optimal control problem for which a control policy $u(t, x)$ can be determined analytically or by straight forward application of numerical methods. In general, the optimal control problem for a system on the form (1) requires the solution of a nonlinear partial differential equation (PDE), the so-called Hamilton-Jacobi Bellman equation, that is both nonlinear and will have infinite boundary conditions that are difficult to handle numerically. However, by applying a logarithmic transformation we show how and when the PDE can be transformed into a linear eigenvalue problem for which there are sometimes analytical solutions and if not, it can readily be solved with standard numerical methods.

Rutquist et al. [7] used this logarithmic transformation to linearize the HJB equation in the scalar one dimensional case. Concurrently, Itami [2] also studied the one-dimensional case and used the transformation to make a coupling between quantum mechanics (the Schrödinger equation) and the Hamilton equation, which was earlier pointed out also by Rosenbrock [4].

In a later study [5] the use of the transformation was further developped to several dimensions for the special case when the disturbance enters the system in the same way as the control input, i.e. $G = B$. Here, we generalize the method to the case when $G \neq B$, give necessary and sufficient conditions for when this linearization can be applied, and analyze the implications of these conditions. When linearization can be applied, it gives an intuitive control method for state constrained systems with only one main tuning parameter that determines the control aggression.

Two examples where the results are applied are presented here. The first one shows how to deal with disturbances that are not purely white. The other is a buffer example treating the pumping of wastewater to a wastewater treatment plant in Göteborg, Sweden.

II. THE OPTIMAL CONTROL PROBLEM

We define the problem as to find a feedback control policy $u(t, x)$ that minimizes

$$V(x(t), t) = \mathrm{E}\left\{\int_t^{t_f} \left(l(x(\tau)) + u^T(\tau)Qu(\tau)\right)\, d\tau + V_f(x(t_f))|x(t)\right\}, \tag{2}$$

where $t \in \mathbb{R}$ is the current time, $t_f > t$ is the final time, $l : \Omega \to \mathbb{R}^+$ describes the (time independent) cost (non-singular on Ω) associated with the state, $V_f : \Omega \to \mathbb{R}$ is the final cost, and u^TQu defines the cost of the control signal.

The stochastic HJB equation for the minimization of (2) can be formulated as

$$-\frac{\partial V}{\partial t} = \min_u\{l + (\nabla V)(f + Bu) + u^TQu + \frac{1}{2}\mathrm{tr}\big[(\nabla^T\nabla V)GWG^T\big]\}, \tag{3}$$

where $V : [t, t_f]\times\Omega \to \mathbb{R}$ is the so-called cost-to-go function (see for instance [1] for details on the derivation of this equation) and the gradient ∇V is defined as a row vector.

Minimization of this quadratic expression with respect to u gives the optimal control input

$$u = -\frac{1}{2}Q^{-1}B^T(\nabla V)^T. \tag{4}$$

T. Wik and C. Breitholtz are with Department of Signals and Systems, Chalmers University of Technology, SE 412 96 Göteborg, Sweden tw@chalmers.se, claesbr@chalmers.se

P. Rutquist is with Tomlab Optimization AB, Västerås, Sweden per.rutquist@gmail.com

Inserting the optimal u into (3) gives

$$-\frac{\partial V}{\partial t} = l - \frac{1}{4}(\nabla V)BQ^{-1}B^T(\nabla V)^T + (\nabla V)f + \frac{1}{2}\mathrm{tr}\big[(\nabla^T\nabla V)GWG^T\big] \tag{5}$$

with infinite boundary conditions on $\partial\Omega$ due to the state constraints. The solution to this equation inserted into (4) gives the optimal control policy. The problem though, is that this PDE is not readily solved because it is nonlinear in V and the infinite boundary conditions are difficult to handle numerically.

III. Linearization

Now, applying the transformation

$$V = -2\kappa \log Z, \tag{6}$$

where κ is an arbitrary real constant, gives

$$\nabla V = -\frac{2\kappa}{Z}\nabla Z$$
$$\nabla^T\nabla V = \frac{2\kappa}{Z^2}(\nabla Z)^T\nabla Z - \frac{2\kappa}{Z}(\nabla^T\nabla Z).$$

Equation (5) is then transformed into

$$\begin{aligned} 2\kappa\frac{\partial Z}{\partial t} =& zl - \frac{\kappa^2}{Z}(\nabla Z)BQ^{-1}B^T(\nabla Z)^T - 2\kappa(\nabla Z)f \\ &+ \frac{\kappa}{Z}\mathrm{tr}\big[(\nabla Z)^T(\nabla Z)GWG^T\big] \\ &- 2\kappa\mathrm{tr}\big[(\nabla^T\nabla Z)GWG^T\big]. \end{aligned} \tag{7}$$

If two matrices A and B have matching dimensions $\mathrm{tr}[AB] = \mathrm{tr}[BA] = \mathrm{tr}\big[A^TB^T\big]$, and the trace of a scalar equals the scalar itself. Therefore we have that

$$\begin{aligned} (\nabla Z)BQ^{-1}B^T(\nabla Z)^T &= \mathrm{tr}\big[(\nabla Z)^T(\nabla Z)BQ^{-1}B^T\big] \\ \mathrm{tr}\big[(\nabla Z)^T(\nabla Z)GWG^T\big] &= (\nabla Z)GWG^T(\nabla Z)^T, \end{aligned}$$

which gives

$$\begin{aligned} \frac{\partial Z}{\partial t} =& \frac{l}{2\kappa}Z + \frac{1}{2Z}(\nabla Z)(GWG^T - \kappa BQ^{-1}B^T)(\nabla Z)^T \\ &- (\nabla Z)f - \mathrm{tr}\big[(\nabla^T\nabla Z)GWG^T\big]. \end{aligned} \tag{8}$$

Now, if

$$GWG^T = \kappa BQ^{-1}B^T \tag{9}$$

the transformed HJB equation becomes a linear PDE:

$$\frac{\partial Z}{\partial t} = \frac{l}{2\kappa} - (\nabla Z)f - \mathrm{tr}\big[(\nabla^T\nabla Z)GWG^T\big] \tag{10}$$

with boundary conditions

$$Z = 0, \quad x \in \partial\Omega \tag{11}$$

and

$$Z_f \triangleq Z(t_f, \mathbf{x}) = \exp\left(\frac{-V_f}{2\kappa}\right). \tag{12}$$

For the finite time case this PDE can be solved using variable separation, i.e. $Z(t,x) = T(t)\phi(x)$ [6]. The time dependent part $T(t)$ has an analytical solution and the state dependent part $\phi(x)$ becomes a (partial) linear differential equation which may have an analytical solution or be readily solved numerically using an eigenvalue solver.

For the stationary problem $t_f \to \infty$ and the integral in (2) will not converge. However, since the process is ergodic the expected cost per unit time (λ) will eventually be the same everywhere in Ω, i.e.

$$\lambda = -\frac{\partial V}{\partial t} = \frac{2\kappa}{Z}\frac{\partial Z}{\partial t}, \tag{13}$$

which gives us the *linear eigenvalue problem*

$$\begin{aligned} \lambda Z &= lZ - 2\kappa(\nabla Z)f - \kappa\mathrm{tr}\big[(\nabla^T\nabla)GWG^T\big] \\ Z &= 0 \qquad \text{on } \partial\Omega, \end{aligned} \tag{14}$$

where the solution with the least λ is sought since it corresponds to the lowest cost.

Clearly, the problem of determining the optimal solution is greatly simplified by the transformation (6) if (9) also holds so that the problem becomes linear as well. In the following we will state the necessary and sufficient conditions for the linearization to be applicable, and an analysis of the implications of (9) on the optimal control.

Preliminaries

A singular value decomposition (SVD) of B gives

$$B = U_1\Lambda_B U_2^T.$$

where U_1 $(n \times n)$ and U_2 $(m \times m)$ are orthogonal matrices, and the $r_B = \mathrm{rank}(B)$ singular values on the upper left diagonal of Λ_B are the square roots of the nonzero eigenvalues of both BB^T and B^TB. Then the columns of U_1 and U_2 give the orthonormal bases for all four fundamental subspaces [8]:

$$\begin{aligned} E_{R_B} &= \text{the first } r_B \text{ columns of } U_1, \\ E_{N_B^T} &= \text{the last } n - r_B \text{ columns of } U_1, \\ E_{R_B^T} &= \text{the first } r_B \text{ columns of } U_2, \\ E_{N_B} &= \text{the last } m - r_B \text{ columns of } U_2, \end{aligned}$$

where E_{R_B} has columns that are the base vectors for the column space $\mathcal{R}(B)$, $E_{N_B^T}$ for the left nullspace $\mathcal{N}(B^T)$, $E_{R_B^T}$ for the row space $\mathcal{R}(B^T)$, and E_{N_B} for the nullspace $\mathcal{N}(B)$ of B.

For a symmetric positive definite matrix, such as the covariance matrix W, SVD is identical to diagonalization with orthogonal eigenvectors, i.e.

$$W = U_W\Lambda_W U_W^T,$$

where Λ_W is a $(n_w \times n_w)$ diagonal matrix with the real positive eigenvalues $\lambda_{W,i}$ of W on the diagonal. We may then define the square root of such a matrix as

$$W^{1/2} = U_W\Lambda_W^{1/2}U_W^T,$$

where $\Lambda_W^{1/2} = \mathrm{diag}(\sqrt{\lambda_{W,1}}, \ldots \sqrt{\lambda_{W,n_w}})$. Clearly, $W^{1/2}$ is also symmetric and positive definite, and $W^{1/2}W^{1/2} = W$. Congruously,

$$W^{-1/2} = U_W\Lambda_W^{-1/2}U_W^T,$$

where $\Lambda_W^{-1/2} = \mathrm{diag}(1/\sqrt{\lambda_{W,1}}, \ldots 1/\sqrt{\lambda_{W,n_w}})$.

Theorem

The HJB equation (3) can be linearized if and only if $\mathcal{R}(G) \subseteq \mathcal{R}(B)$, which is true if and only if

$$E_{N_B^T}^T G = 0, \tag{15}$$

The optimal control policy is then given by the linearized HJB (14) and

$$u = B^+ GWG^T (B^+)^T B^T \frac{1}{Z} (\nabla Z)^T, \tag{16}$$

where B^+ denotes the pseudoinverse of B. The corresponding cost matrix Q is implicitly given by

$$\begin{aligned} Q^{-1} = & \frac{1}{\kappa} \left(B^+ GWG^T (B^+)^T + E_{N_B} \Theta \Theta^T E_{N_B}^T \right. \\ & \left. + \epsilon E_\Delta E_\Delta^T \right), \end{aligned} \tag{17}$$

where $\kappa > 0$ is an arbitrary scalar, Θ is any full rank matrix, and

$$E_\Delta = (I_m - \Phi(\Phi^T \Phi)^{-1} \Phi^T) E_{R_B^T},$$

where $\Phi = B^+ GW^{1/2}$, and $\epsilon \to 0$.

Remarks

1) The test (15) corresponds to checking that all directions of Gw can be directly counteracted by Bu. It is noteworthy that this sufficient and necessary condition for HJB linearization has this clear physical interpretation.
2) Q determines the cost of control action through $u^T Q u$ in (2). In the same way as for W the SVD of Q is

   $$Q = U_Q \Lambda_Q U_Q^T,$$

   where $\Lambda_Q = \text{diag}(\lambda_{Q,1}, \ldots \lambda_{Q,m})$, determines the cost in the directions given by U_Q. Since the directions are preserved in the SVD of Q^{-1}:

   $$Q^{-1} = U_Q \Lambda_Q^{-1} U_Q^T, \tag{18}$$

   where $\Lambda_Q^{-1} = \text{diag}(1/\lambda_{Q,1}, \ldots 1/\lambda_{Q,m})$, Q^{-1} determines the "preferred" control action directions.
   - The second term in (17) is in the null space of B and therefore does not contribute to Bu.
   - The first term, on the other hand, corresponds to a complete matching of the directions of Gw (see the proof).
   - The last term, which corresponds to the possible directions of the control input that cannot directly counteract the directions of the disturbance, is only needed for Q to be formally invertible. However, by setting ϵ to a small number and invert Q^{-1} it becomes clear from large elements in Q what directions (combinations of input signals) that should/will be avoided.
3) κ is the main tuning parameter that determines the trade-off between cost on the states and cost of control activity. The smaller κ the more agressive the control.
4) If the covariance matrix W has not been determined from data, it is next to κ the remaining tuning parameter. If it is evident, when applying the control policy, that we are never close to a boundary in one direction, the covariance in the corresponding disturbance direction can be decreased. If the constraints in one direction are violated, the corresponding covariance should be increased.
5) For the case when we have input noise on all inputs, (17) reduces to $Q^{-1} = \kappa^{-1} W$, which is the case studied in [5].

Proof

We need to show that if the required conditions are fulfilled, (17) implies that the equality (9) holds when $\epsilon \to 0$.

Presuming Q is positive definite and symmetric, Q^{-1} is also positive definite and symmetric according to (18), and we may define the square root in the same way as shown for W. Equation (9) can then be written

$$GW^{1/2}(GW^{1/2})^T = (\sqrt{\kappa} B Q^{-1/2})(\sqrt{\kappa} B Q^{-1/2})^T.$$

Clearly, the HJB equation is linearized if and only if

$$GW^{1/2} = \sqrt{\kappa} B Q^{-1/2}. \tag{19}$$

Because $Q^{-1/2}$ has full rank, the columns on the right hand side can take any, and no other, vector values than those spanned by the columns of B, i.e. $\mathcal{R}(B)$. So, if and only if the columns of $GW^{1/2}$ are entirely in $\mathcal{R}(B)$ will (19) have a solution. According to the fundamental theorem of linear algebra [8] the left nullspace $\mathcal{N}(B^T)$ is the orthogonal complement of $\mathcal{R}(B)$. Hence, if the columns of $GW^{1/2}$ are entirely in $\mathcal{R}(B)$ all columns should be orthogonal to all base vectors for $\mathcal{N}(B^T)$, which is exactly the test (15).

Now, regard the columns of $Q^{-1/2}$ as vectors (in $\mathbb{R}^m$). Each of these vectors can be written as a sum of one vector in the rowspace $\mathcal{R}(B^T)$ of B and another vector in the nullspace $\mathcal{N}(B)$ of B, since these two spaces are orthogonal and span the entire $\mathbb{R}^m$. We may therefore split $Q^{-1/2}$ such that

$$Q^{-1/2} = Q_{R^T}^{-1/2} + Q_N^{-1/2},$$

where $Q_{R^T}^{-1/2}$ have columns entirely in $\mathcal{R}(B^T)$ and $Q_N^{-1/2}$ have columns entirely in $\mathcal{N}(B)$. Then

$$\begin{aligned} Q^{-1} &= Q^{-1/2}(Q^{-1/2})^T \\ &= Q_{R^T}^{-1/2}(Q_{R^T}^{-1/2})^T + Q_N^{-1/2}(Q_N^{-1/2})^T \\ &= Q_{R^T}^{-1} + Q_N^{-1} \end{aligned}$$

since $(Q_{R^T}^{-1/2})^T Q_N^{-1/2} = 0$ because of the orthogonality.

The component of the columns of $GW^{1/2}$ in $\mathcal{R}(B^T)$ are

$$Q_{G,R^T}^{-1/2} = \frac{1}{\sqrt{\kappa}} B^+ GW^{1/2}.$$

Thus, $Q^{-1} = Q_{G,R^T}^{-1} = Q_{G,R^T}^{-1/2}(Q_{G,R^T}^{-1/2})^T$ is enough to satisfy (9) since $BQ_N^{-1/2} = 0$ by definition. However, if B or G does not have full rank $Q^{-1} = Q_{R^T}^{-1}$ will not be invertible and we have to add symmetric matrices to give Q^{-1} full rank. If B is not full rank we add the basis for the nullspace. From the SVD of B we have that the columns of

E_{N_B} are an orthonormal base for $\mathcal{N}(B)$. Hence, we can fill the space using

$$Q_N^{-1} = E_{N_B}\Theta(E_{N_B}\Theta)^T = E_{N_B}\Theta\Theta^T E_{N_B}^T,$$

where Θ is an arbitrary full rank matrix.

Since

$$BQ_N^{-1}B = (BE_{N_B})(\Theta\Theta^T E_{N_B}^T)B^T = 0$$

Q_N^{-1} will not contribute to (9). Further,

$$Q_N^{-1}B^T(\nabla V)^T = \left((\nabla V)BQ_N^{-1}\right)^T = 0$$

because Q_N^{-1} is symmetric. Hence, Q_N^{-1} will neither have an effect on the control signal (4) and nor on the cost (2). This is important because it means that no control signals are wasted in the null space of B, and nor are any control signals generated that would not contribute to the cost function.

If the test (15) is fulfilled

$$Q^{-1} = Q_{G,R^T}^{-1} + Q_N^{-1}$$

will still not have full rank if $GW^{1/2}$ does not fill Q_{R^T}, i.e. there are missing dimensions in $\mathcal{R}(B^T)$. These dimensions are obtained if we project $\mathcal{R}(B^T)$ on $Q_{G,R^T}^{-1/2}$:

$$E_\Delta = (I - \Phi(\Phi^T\Phi)^{-1}\Phi^T)E_{R_B^T},$$

where $\Phi = B^+GW^{1/2}$. The columns of E_Δ corresponds to the directions of the control input not needed to directly counteract the disturbances. If we add

$$Q_{B-G,R^T}^{-1} = E_\Delta(E_\Delta)^T$$

to $Q_{G,R^T}^{-1} + Q_N^{-1}$, Q^{-1} becomes invertible but (9) will no longer hold. However, since the inversion of Q is not explicitly needed for the calculation of the optimal u, we may add $\epsilon Q_{B-G,R^T}^{-1}$ to get (17), i.e.,

$$Q^{-1} = Q_{G,R^T}^{-1} + Q_N^{-1} + \epsilon Q_{B-G,R^T}^{-1}$$

and then let $\epsilon \to 0$.

IV. Examples

First we illustrate with a very simple example how the vector spaces affect Q. Then two applications of the results are presented. The first one is the case when the disturbance enters as an addition to the control input, but is not purely white. The other application is a buffer example treating the pumping of wastewater to a wastewater treatment plant in Göteborg, Sweden.

A. Illustration of the components of Q^{-1}

Let

$$B = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad \text{and} \quad G = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix}$$

Clearly B has a null space since the last row are zeros. Further, we see that $\mathcal{R}(G) \subseteq \mathcal{R}(B)$ since G equals twice the first column of B. Hence, we may linearize the HJB and all three terms of (17) are non-zero. In fact, there will be one dimension in each term:

$$E_{N_B} = \frac{1}{\sqrt{3}}\begin{bmatrix} 1 \\ 1 \\ -1 \end{bmatrix} \quad \text{and} \quad E_\Delta = \begin{bmatrix} 0 & 0 \\ 0.61 & 0.35 \\ 0.61 & 0.35 \end{bmatrix}.$$

Also, because G equals twice the first column of B we realize that only u_1 should need to be used. Calculation of Q from (17) with $\kappa = 1$ and $\epsilon = 10^{-4}$ gives

$$Q = \begin{bmatrix} 0.46 & 0.27 & -0.27 \\ 0.27 & 500 & 500 \\ -0.27 & 500 & 500 \end{bmatrix}$$

which confirms that neither u_2 nor u_3 will be used in the optimal control.

B. Coloured disturbance

For physical reasons almost all disturbances are more or less of low pass character. However, if the system dynamics are slower than the time constants of the disturbances the assumption of white noise is often an acceptable approximation. In some cases though, there may be slow components that also need to be considered. Consider the following system with input disturbance

$$\begin{aligned} \dot{x} &= Ax + B(u + v) \\ y &= Cx + Du + e, \end{aligned} \tag{20}$$

where the disturbance v contains slow dynamics. From physical modeling, or spectral analysis, a model of the disturbance dynamics has been derived:

$$\begin{aligned} \dot{x}_w &= A_w x_w + B_w w \\ v &= C_w x_w + D_w w, \end{aligned} \tag{21}$$

where w is white Gaussian noise with covariance W. Inserting this disturbance model into (20) gives

$$\begin{aligned} \dot{x} =& Ax + Bu + BC_w x_w + BD_w w \\ =& Ax + B\tilde{u} + Gw \end{aligned} \tag{22}$$

where $\tilde{u} = u + C_w x_w$ and $G = BD_w$.

The disturbance state variables x_w are not controllable but a successful observer (for example a Kalman filter) giving estimates $\hat{x}$ and $\hat{x}_w$ close to the real states is assumed to be at hand. Assuming fully known states, x and x_w, the system (22) fits into (1) and because $G = BD_w$ we have that $\mathcal{R}(G) \subseteq \mathcal{R}(B)$, which implies that the condition (15) always holds. Hence, we may apply the linearization and solve the linearized HJB equation for (22) to obtain the control policy $\tilde{u}(x)$. The actually applied control signal is then

$$u = \tilde{u} - C_w\hat{x}_w.$$

The cost matrix Q is now acting on $\tilde{u}$ through

$$\tilde{u}^T Q\tilde{u} = u^T Qu + 2u^T QC_w x_w + x_w^T C_w QCx_w.$$

The first term is the same as before and has the same interpretation as before, and the last term has no effect on the optimal control policy since x_w is not controllable and does not depend on u. The second term is likely to have only a

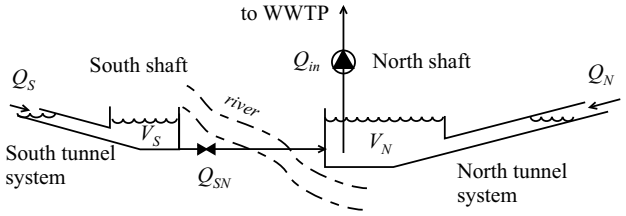


Fig. 1. The Rya WWTPs buffer system for their influent pumping station (24 m below ground level).

limited influence on u because $\mathrm{E}\{x_w\} = 0$. However, using SVD the added cost can be written as

$$u^T Q C_w x_w = u^T U_Q \Lambda_Q U_Q^T C_w x_w,$$

where $\Lambda = \mathrm{diag}(\lambda_{Q,1}, \ldots, \lambda_{Q,2})$ defines the costs in the directions of U_Q. It can therefore be seen as an additional cost for the influence of the slow dynamic disturbances ($C_w x_w$) in the directions of u already undesired because of the direct term.

If the disturbance does not enter as an addition to the input as in (20), i.e.

$$\dot{x} = Ax + Bu + \tilde{G}v$$

the treatment is only slightly changed if there exists a matrix M such that $\tilde{G}C_w = BM$. Then we get $u = \tilde{u} - M\hat{x}_w$ and $G = \tilde{G}D_w$. The interpretation of the linearization requirement remains the same, i.e. the direct terms from the white noise must be possible to directly counteract by the control signal.

C. A wastewater treatment buffert system

The Rya wastewater treatment plant (WWTP) treats the wastewater from the Göteborg region.Wastewater is transported to the Rya WWTP through a large tunnel system that can be considered as two separated systems of about the same size, separated by the river Göta älv (see Fig. 1).

Simplified, and with the notation

$$x = \begin{bmatrix} V_N \\ V_S \end{bmatrix}, \quad u = \begin{bmatrix} Q_{in} \\ Q_{SN} \end{bmatrix} \quad \text{and} \quad w = \begin{bmatrix} Q_N \\ Q_S \end{bmatrix},$$

where all variables are deviations from the operating point, the system can be described by

$$\frac{dx}{dt} = \begin{bmatrix} -1 & 1 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}.$$

w_1 and w_2 are considered (within the time scale considered) as white with a fairly strong correlation, since they are both the results of figuratively the same human habits and rain run off. A covariance matrix

$$W = \begin{bmatrix} 1 & 0.8 \\ 0.8 & 1 \end{bmatrix} \tag{23}$$

is assumed.

Pumping is one of the major costs for this plant and many other WWTPs. For this plant the average flow (including run-off and infiltration) is about 4 m^3/s and hence, every saved meter of elevation corresponds to more than 340 MWh/year. Remembering that we deal with deviations from an operating point, the variable cost for pumping is assumed to be proportional to $h_{1,\max} - h_1(V_N(t))$, where h_1 depends on the horizontal cross sectional area of the north shaft. However the levels in the north and south shafts should always be above a minimum level and below a maximum level to avoid overflows and the pumps from running dry.

The stationary control problem can now be formulated as

$$\begin{aligned} \min_u \ & \mathrm{E}\left\{h_{1,\max} - h(x_1) + u^T Q u\right\} \\ \text{s.t.} \quad & 0 \le x_1 \le x_{1,\max} \\ & 0 \le x_2 \le x_{2,\max} \end{aligned} \tag{24}$$

where x_1 and x_2 are the volumes above minimum level. We may assume that the relations between height and volume of the north shaft are roughly given by [3]

$$\begin{aligned} V_N &= 10^4 \cdot h_1^2 \\ V_S &= 10^4 \cdot (0.11 h_2^2 + 0.07 h_2^3). \end{aligned}$$

Thus, $h_1 = 0.01\sqrt{x_1}$, and for $h_{1,\max} = 4$ m and $h_{2,\max} = 6$ m we get $x_{1,\max} = 1.6 \cdot 10^5$ m^3 and $x_{2,\max} = 1.9 \cdot 10^5$ m^3.

For this example we have that both B and G have full rank, so

$$Q^{-1} = \frac{1}{\kappa} B^+ G W G^T (B^+)^T,$$

which gives

$$Q = \begin{bmatrix} 2.8 & -5 \\ -5 & 10 \end{bmatrix}$$

for $\kappa = 1$. Because the volume in the north shaft (x_1), is affected by both disturbance flows (through u_2) the cost for u_1 is less than the cost for u_2.

Solving the linearized HJB-equation (14) for $\kappa = 1$ and calculating u gives the result presented in Fig. 2. As can be seen, the control policy produces a control vector that aims away from the constraints, and guarantees that we do not violate them. However, the cost for control activity is not sufficiently low to keep the level high in the north shaft, which was desired in order to reduce the pumping costs. If $\kappa = 0.01$ we allow more control activity and then it should be possible to stay closer to the maximum level in the north shaft. This is confirmed in Fig. 3. The average relative height in the north shaft was increased from 3.0 m ($0.89 \cdot 10^5$ m^3) to 3.40 m ($1.15 \cdot 10^5$ m^3). Note that the full buffer capacity in the south shaft is used in both cases.

Now, assume we have misjudged the disturbance so that the variance of the disturbance in the North shaft is in fact only a tenth what we thought, i.e.

$$W = \begin{bmatrix} 0.1 & 0.25 \\ 0.25 & 1 \end{bmatrix} \tag{25}$$

if the correlation coefficient is unchanged. In Fig. 4 a simulation of that situation is shown ($\kappa = 0.01$). Because of the overestimated variance the controller keeps the level in the north shaft unnecessarily low. To push the level in the north shaft a little higher we decrease the variance of the

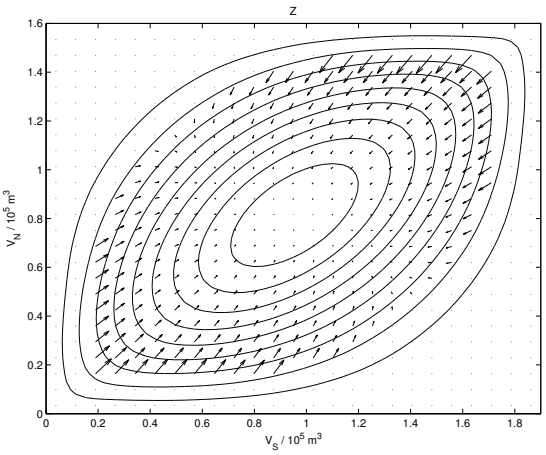


Fig. 2. Contour of the solution Z and direction of the input Bu when $\kappa = 1$. The stationary point where the system has its maximum probability to be is inside the inmost contour.

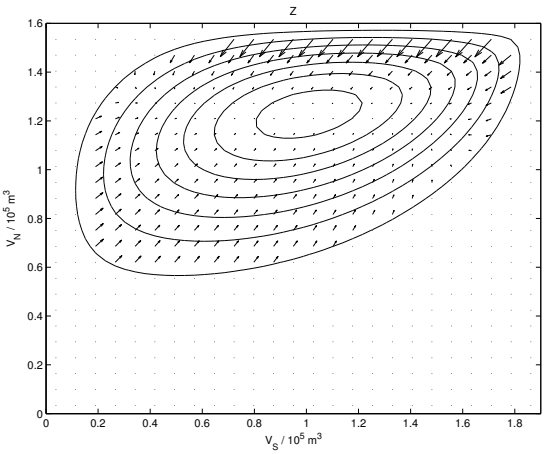


Fig. 3. Contour of the solution Z and direction of the input Bu when $\kappa = 0.01$. The stationary point where the system has its maximum probability to be is inside the inmost contour.

influent flow to the north shaft (keeping the cross correlation coefficient). As can be seen in Fig. 5, redoing the design with the smaller and correct W means that we can stay closer to the upper limit.

V. Conclusions

A method for state constrained control based on stochastic optimal control has been presented. The method is a result of an exact linearization of the HJB-equations, which has implications on how the cost for control is defined. However, the implications have been thoroughly investigated and found to be in agreement with intuitive reasoning about suitable cost matrices. There is one main tuning parameter, which determines the control activity, and then the covariance matrix of the white noise disturbance can also be used for tuning the preferred (or undesired) directions of the control. The method assumes white noise but it has been shown how it can also be applied in cases when the disturbance

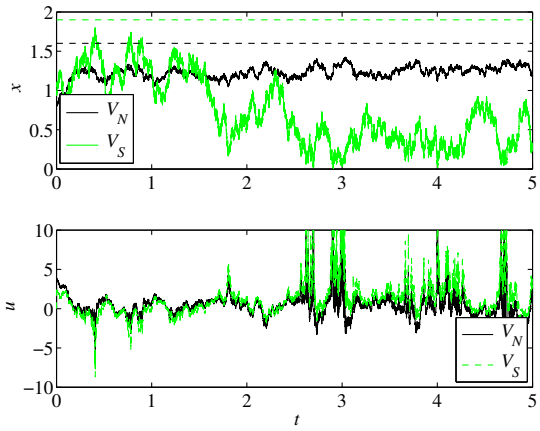


Fig. 4. Simulation of the controlled system with the policy calculated for W given by (23) and simulated for the true W given by (25).

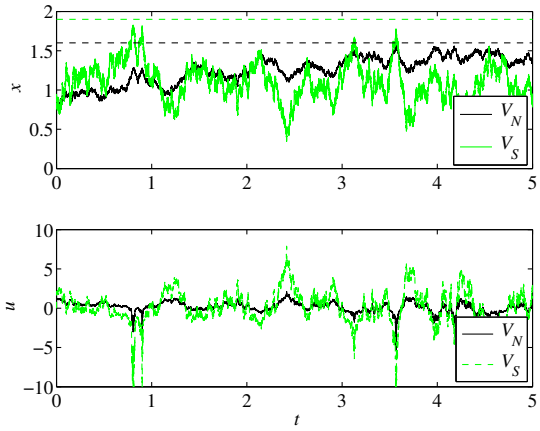


Fig. 5. Simulation of the controlled system with the policy calculated for W given by (25) and simulated for the same (true) W.

is coloured. A buffer example, pumping of wastewater to a wastewater treatment plant from two different shafts in a sewer system, has illustrated the use of the method.

References

[1] P. Dorato, T. C. Abdallah, and V. Cerone. *Linear Quadratic Control: An Introduction*. Prentice-Hall, 1995.

[2] T. Itami. Nonlinear optimal control as quantum mechanical eigenvalue problems. *Automatica*, 41:1617–1622, 2005.

[3] J. Lindqvist, T. Wik, D. Lumley, and G. Äijälä. Influent load prediction using low order adaptive modeling. In *2nd IWA Conference on Instrumentation, Control and Automation*, Busan, South Korea, 2005.

[4] H.H. Rosenbrock. *Physics letters*, 110A:343–346.

[5] P. Rutquist, C. Breitholtz, and T. Wik. On the infinite time solution to state-constrained stochastic optimal control problems. *Automatica*, 44:1800–1805, 2008.

[6] P. Rutquist, C. Breitholtz, and T. Wik. Finite-time state-constrained optimal control for input affine systems with actuator noise. *Automatica*, Submitted.

[7] Per Rutquist, Claes Breitholtz, and Torsten Wik. An eigenvalue approach to infinite-horizon optimal control. In *Proc. 16th IFAC World Congress*, Prague, Czech Republic, jul 2005.

[8] G. Strang. *Linear Algebra and its Applications*. Harcourt Brace and Company, Florida, 3 edition, 1988.