Title	Model predictive control of the solid oxide fuel cell stack temperature with models based on experimental data
Author(s)	Pohjoranta, Antti; Halinen, Matias; Pennanen, Jari; Kiviaho, Jari
Citation	Journal of Power Sources. Elsevier. Vol. 277 (2015), Pages 239 - 250
Date	2015
URL	http://dx.doi.org/10.1016/j.jpowsour.2014.11.126
Rights	Post-print version of the article. This article may be downloaded for personal use only.

Manuscript accepted for publication in the Journal of Power Sources, 2014-11-26

Model predictive control of the solid oxide fuel cell stack temperature with models based on experimental data

Antti Pohjoranta[*], Matias Halinen, Jari Pennanen, Jari Kiviaho

VTT Technical Research Centre of Finland, P.O. Box 1000, FI-02044, Finland

[*]Corresponding author: antti.pohjoranta@vtt.fi, tel. +358 40 570 9825

Abstract

Generalized predictive control (GPC) is applied to control the maximum temperature in a solid oxide fuel cell (SOFC) stack and the temperature difference over the stack. GPC is a model predictive control method and the models utilized in this work are ARX-type (autoregressive with extra input), multiple input-multiple output, polynomial models that were identified from experimental data obtained from experiments with a complete SOFC system. The proposed control is evaluated by simulation with various input-output combinations, with and without constraints. A comparison with conventional proportional-integral-derivative (PID) control is also made. It is shown that if only the stack maximum temperature is controlled, a standard PID controller can be used to obtain output performance comparable to that obtained with the significantly more complex model predictive controller. However, in order to control the temperature difference over the stack, both the stack minimum and the maximum temperature need to be controlled and this cannot be done with a single PID controller. In such a case the model predictive controller provides a feasible and effective solution.

Keywords

SOFC temperature control, ARX modeling, generalized predictive control, MIMO control

1 Introduction

The solid oxide fuel cell (SOFC) technology enables the combined production of heat and power with a high electric efficiency and from a wide range of hydrocarbon fuels. The main obstacle for wide-spread utilization SOFC power systems is their high lifecycle cost, which in turn, is due to the high investment cost related to SOFC systems and the relatively short lifetime of the SOFC stack. The lifetime of a SOFC stack is effectively determined by the rate of the stack performance degradation which depends partly on to the inherent properties of the stack, such as structure and materials, but also on the conditions that stack is operated in. Automatic control can be used to optimize the operating conditions so that the part of stack performance degradation which depends on the operating conditions is minimized.

One significant factor affecting the SOFC stack performance and performance degradation rate is the temperature in which the stack is operated [1]. In particular, a too high operating temperature increases the stack degradation rate unnecessarily, and a too low operating temperature will decrease the stack voltage and its efficiency, especially when operated with hydrocarbon fuels instead of pure hydrogen. Also the spatial temperature variations inside the SOFC stack can be a cause for stack degradation [2, 3]. The larger the temperature variations are, the bigger are the mechanical stresses posed on the stack structure due to mismatch of the stack materials' thermal expansion coefficients. Hence, both the absolute temperature as well as the temperature distribution inside a SOFC stack should be kept within desired boundaries during both steady-state and transient operation, and model predictive control, which incorporates operating constraints can be used to achieve this target.

Physical modeling of SOFCs and SOFC systems is today at a very advanced level. Good overviews of dynamic SOFC modeling and its status-quo are found in e.g. [4, 5]. One typical application for dynamic models is using them to develop control for the modeled processes, and several efforts on SOFC system control have also been reported. These works can be roughly divided into two categories: (i) control aiming primarily for SOFC load following, e.g. [6-13] and (ii) SOFC temperature control, e.g. [14-18]. Avoiding fuel starvation is typically a part of the load-following control, but in [19] the control focus is also on avoiding carbon formation in the SOFC. Both the fuel starvation limit and the chance of forming solid carbon in the system pose relevant operating constraints on SOFCs and thus on their control.

The temperature control developments found in the literature include both simple decentralized, as well as complex advanced control approaches. In [14, 17] the control is based on decentralized PI and PID controllers (proportional-integral-derivative), which are tuned by utilizing the model so to provide a desired SOFC stack temperature response. In both cases the air flow is used as the manipulated variable. Advanced control is used in [15] and [16, 18] where the control is based on a variable structure control and H_∞ control, respectively. In [15] the air and fuel flow are considered as manipulated inputs, while in [16], the air flow and air temperature before the stack are the manipulated variables. In all the said cases the control development is based on a 0-, 1- or 2-dimensional physical SOFC model. For the purpose of advanced control development, such models need to be simplified somehow and the mathematical treatment of the control derivation becomes easily rather involved, which always raises the threshold for SOFC practitioners to apply these results in their work.

This paper aims to present a simple means for SOFC stack temperature regulation and control by applying the generalized predictive control (GPC) method [20], which is a variant of model predictive control (MPC). In contrast to other works, the model utilized in this work is a linear polynomial input-output model and is identified directly from experimental data [21], whereby an extensive part of physical modeling and/or model simplification is avoided. As first result, the obtained control algorithm is tested by simulation and various input-output combinations are examined in closed-loop. The behavior of the system controlled by GPC with and without typical practical constraints is analyzed and a comparison with PID control is made to clarify when the development of an advanced control algorithm is worthwhile. Attention is given to how the control handles the continuous rise of the SOFC stack temperature, induced by stack performance degradation.

The control problem tackled here is deliberately limited to stack temperature control for several reasons. The thermal inertia of the stack and the system effectively dominates the transient behavior of both the system temperatures and the stack voltage, assuming that fuel and air starvation are avoided. The stack temperature is a property that must and can be controlled actively by the adjustable system inputs, whereas the values of such properties as the fuel flow rate and anode off-gas recycle flow rate are often dictated by their feasible operating bounds and the maximal efficiency criterion together. Similarly, it is considered reasonable to assume that the system load current is a non-controllable input to the system as its value is dictated rather by the desired load output than by the control system. Finally, as there are several means to affect the stack temperature, it is considered essential to find that combination of these means which has least adverse effects to system operation in terms of e.g. efficiency. To summarize, the load current is assumed to be an external input and its value together with the maximal efficiency requirement and system specifications essentially dictate the minimal air and fuel flow rates. Therefore the control effort is focused on providing the SOFC stack with the best possible operating conditions in terms of stack temperature and stack temperature distribution.

There are several similarities between this and earlier works. In particular, the approach to the modeling and control development is analogous to that in [10, 19], where also system identification, Kalman filtering and model predictive control were used. The control aim, however, was not temperature control and the control model was a set of linear parameter-varying models instead of a linear time-invariant (LTI) model which is used here. Model predictive control and system identification to obtain an LTI model for control purposes was also applied in [8, 12], primarily for load-following control development, however. In all these works [8, 10, 12, 19] the control model was identified from simulated data, whereas in this paper, models identified directly from experimental data are used.

Section 2 contains a brief overview of the experiments and the data treatment related to this work, but for a more detailed description of this part as well as hardware-related details, the reader is directed to [21] and the references therein. Model identification for control development purposes and the process of developing a generalized predictive control algorithm are described briefly in Section 3. GPC is then applied to the case problem in Section 4. Simulation results, with a brief discussion are given in Section 5. Together with [21] these papers describe a complete process from experiment design to model identification and model predictive control development for a full solid oxide fuel cell system.

2 Experimental and model identification work

The experiments, the model identification process and the development of a Kalman filter -based estimator for the stack temperatures are described fully in [21]. Therefore only a brief overview is given here.

2.1 Experiments and data pre-treatment

The case-system in this work is a complete 10kW SOFC system with a single planar SOFC stack [22, 23]. The data was obtained from designed experiments in which the system was operated in varying operating conditions, reached by manipulating four selected input variables: load current, air flow, air inlet temperature and natural gas feed (see Table 1). To collect relevant data over the full system operating range, a so-called fractional factorial experiment design was carried out twice, around two different central operating points, one at the system nominal point and the other at a part-load operating point [21].

The stack maximum and minimum temperatures as well as the stack cathode outlet temperature were recorded as response. The internal stack temperatures were measured with thermocouples placed on a single, representative cell at the middle of an 80-cell SOFC stack. The response values considered in this work were obtained by taking the maximum and minimum values of the internal temperature measurements.

Data sets for model identification and control development were produced by pre-treating the measurement data in two and three steps:

1) The data was down-sampled from 1 sample per second to 1 sample per 5 minutes.
2) Data offset removal was carried out by subtracting the nominal operating condition (NOC) values from the data. The I/O data series obtained after steps 1) and 2) are denoted $u_{1\ldots4}$ and $y_{1\ldots3}$ for the inputs and outputs, respectively. These data series effectively describe the system behavior as a deviation from its nominal operating point.
3) The degradation effects were removed from the output data by removing a linear trend from the data ("de-trending"). The output data series obtained after steps 1)-3) are denoted by $y_{1\ldots3,DT}$.

The utilization of the various data series is clarified in Section 2.2.

Finally, the full input-output data series was divided into two parts, one for model identification and one for validation. The input/outputs variables' descriptions, symbols and nominal operating condition values are collected into Table 1. Figure 1 illustrates the effects of de-trending for the first response variable, stack maximum temperature. Also the full data series as well as the periods of the identification and validation data segments are indicated in the figure.

Table 1 - Considered system inputs and outputs.

	Symbol for physical variable	Unit	Nominal value	Symbol for variable after pre-treatment
Inputs				
Load current	I	A	160	u_1
Air flow	$\dot{V}_{air}$	dm^3 min^{-1}	1062	u_2
Air inlet temperature	$T_{air,in}$	°C	735	u_3
Fuel flow	$\dot{V}_{NG}$	dm^3 min^{-1}	27.90	u_4
Outputs				
Maximum temperature	T_{max}	°C	773.3	y_1, $y_{1,DT}$
Minimum temperature	T_{min}	°C	692.9	y_2, $y_{3,DT}$
Cathode outlet temperature	$T_{cath,out}$	°C	737.3	y_3, $y_{3,DT}$

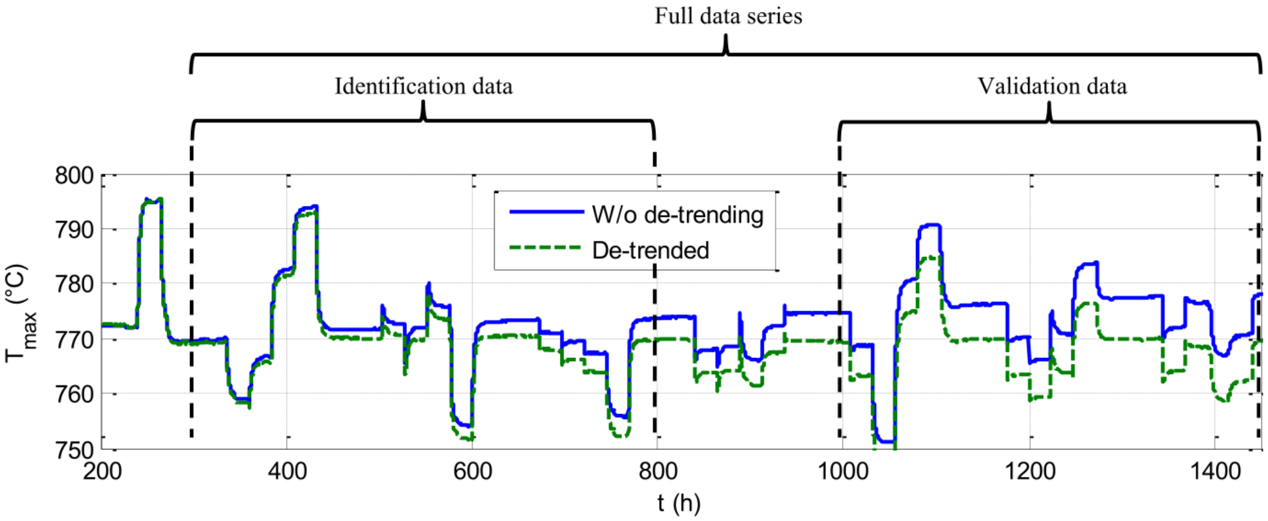

Figure 1– The stack maximum temperature (i.e. first output) with and without de-trending. The periods of the full data segment as well as the identification and validation data are also indicated.

2.2 Model identification for control development

For the purpose of simulation-based control development, a total of three models are necessary:

1) The process model, denoted M_1, which is used to simulate the process when testing and evaluating the control algorithm.
2) The control model (M_2) which is used to implement the control algorithm. That is, this model is used to calculate the control algorithm parameters.
3) The estimation model (M_3) which is used to implement the estimation of non-measureable properties of the process.

All models, especially the process model (M_1) should represent the physical process as accurately as possible. However, for control development, the dynamic characteristics of the physical process are of higher importance than absolute steady-state accuracy. Steady-state modeling errors may be compensated for by feedback and integral action in the control. It is entirely possible to use the same model as all $M_{1\ldots3}$ but this does not enable reliable examination of the control algorithm characteristics as there would be no modeling error. Therefore, in this work the following modeling approach is adopted:

1) The process model, M_1, is identified from all the available data and its accuracy is optimized by allowing high parameter dimensions. A degradation trend, identified from the data, is also added to the output of M_1 (clarified below).
2) The control model, M_2, is identified from the de-trended identification data (a subset of the available data). The M_2 outputs include only those outputs which are controlled, i.e. the stack minimum and maximum temperature. The parameter dimensions for M_2 are lower than for M_1, the process model, and equal to the dimensions of the estimation model M_3.
3) The estimation model, M_3, and the estimator are taken directly from [21]. M_3 and M_2 have the same parameter dimensions but the identification data used to calculate their parameters is pre-treated differently. Specifically, no data de-trending is applied when identifying M_3, as clarified in [21].

Remarks – In this work, the estimator is applied only in one of the test cases (Case 4). In the rest of the cases, it is assumed that the estimator is accurate enough so that the output of M_1 can be directly fed back to the control system. This simplification is done only to make the control development process more illustrative and it is justified by the results of [21], where an accurate estimator for the case process was developed. It must be recognized, however, that in practice, when an estimator is used, the accuracy of the controlled output depends directly also on the estimator accuracy. For control development purposes, however, it simplifies the analysis to not have the estimation inaccuracy included. Furthermore, it should be noted that, contrary to common practice, the estimation model is not identified here based on the process model output, but based on the original measurement data. Doing so increases the estimation error but again simplifies the process.

All models $M_{1...3}$ are multiple-input-multiple-output (MIMO), discrete-time models and all models were identified by using the processes described above and detailed in [21]. All models are of the ARX-type (autoregressive model with extra inputs) given by (1).

$$A(q)y(t) = B_{id}(q)u_{id}(t) + e(t) \qquad (1)$$

In (1) A and B_{id} are matrix valued polynomials in q with orders n_a and n_b, respectively, and they contain the identified model parameters. q is the forward shift operator. u_{id} $(m_{id} \times 1)$ and y $(n \times 1)$ denote the pre-treated input and output signals, respectively, used for model identification. The model includes a process dead-time of n_k time-steps.

For process simulation purposes, a linear degradation trend with slope d_0 ($n \times 1$, °C/[t]) and intercept d_1 ($n \times 1$, °C) is added to M_1. For M_2 and M_3 only the nominal condition output, removed from the data during model identification, is added to the output (i.e. d_0 can be assumed zero for M_2 and M_3). Noting that the first parameter matrix A_1 in A is the identity matrix ($A_1 = I_{n\times n}$), the output of $M_{1...3}$ is obtained as given in (2).

$$y(t) = -\sum_{i=1}^{n_a} A_i\, y(t-i) + \sum_{i=0}^{n_b} B_{id,i}\, u_{id}(t-i) + d_0 t + d_1 \qquad (2)$$

The user-defined model parameters and the input/output structure as well as the output trend parameters of all models are given in Table 2.

Table 2 - Fixed model parameters and input-output configuration for process model and control model.

	Process model (M_1)	Control model (M_2)	Estimation model (M_3)
m_{id}	4	4	4
n	3	2	3
n_a	5	3	3
n_b	5	3	3
n_k	0	1	1
Inputs	$u_{id} = [u_1 \quad u_2 \quad u_3 \quad u_4]^T$		
Outputs	$y = [y_{1,DT} \quad y_{2,DT} \quad y_{3,DT}]^T$	$y = [y_{1,DT} \quad y_{2,DT}]^T$	$y = [y_1 \quad y_2 \quad y_3]^T$
d_0	$[0.59 \quad 0.52 \quad 0.47]^T \cdot 10^{-3}$	$[0 \quad 0]^T$	$[0 \quad 0 \quad 0]^T$
d_1	$[768.0 \quad 688.1 \quad 733.1]^T$	$[773.2 \quad 692.9]^T$	$[773.2 \quad 692.9 \quad 737.3]^T$

Figure 2 displays the models' first output (maximum temperature) compared to the measured output data. The figure highlights the modeling result appropriately; M_1 captures the process output dynamics as well as the absolute level of the measurement accurately. M_2 and M_3 are not as accurate as M_1 and do not replicate the degradation trend, but they are sufficiently accurate for control and estimator development purposes, respectively.

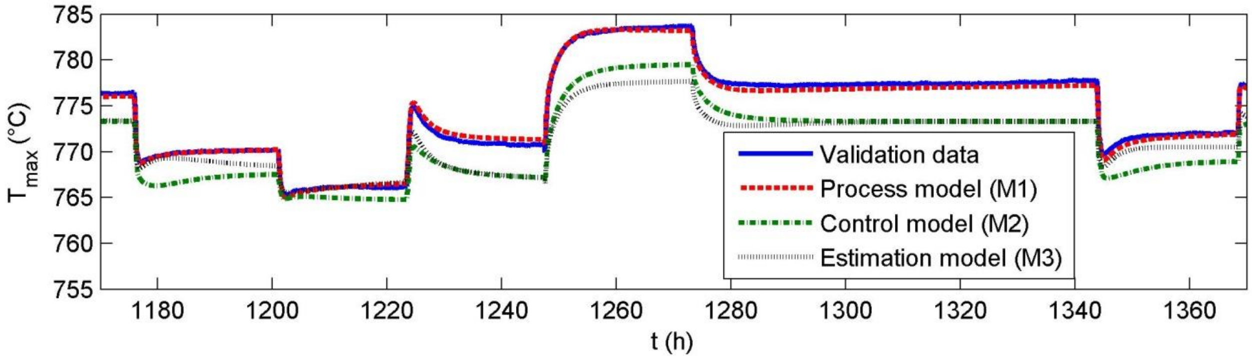

Figure 2 - Measured data compared with the output of $M_{1\ldots3}$.

3 Control

In this section, the generalized predictive control (GPC) method, devised by Clarke et al. [20] is applied to develop model predictive temperature control of the SOFC stack. The MIMO-formulation for the GPC algorithm can be found e.g. in [24].

3.1 Model formulation

In [24] the GPC algorithm is given starting from the CARIMA type model (controlled, autoregressive with integrated moving average) with manipulated inputs u, controlled outputs y, measured disturbances v and a noise signal e as shown in (3).

$$A(q)y(t) = B(q)u(t-1) + D(q)v(t) + \frac{1}{1-q^{-1}}e(t) \tag{3}$$

In [25] it is shown, however, that the GPC algorithm is as such applicable also to the ARX model structure (1), provided that the algorithm parameters are obtained accordingly. Therefore, the ARX model structure is maintained throughout this work. Note also that a model with the CARIMA structure could also be identified from the available data directly, e.g. by differencing the data as shown in [26].

For control purposes, the models obtained by identification in the ARX form (1) are re-grouped so that the controlled inputs u and the not-controlled, but measured, inputs v (i.e. the measured disturbances) are separated. The so obtained ARX model with measured disturbances can be expressed as (4).

$$A(q)y(t) = B(q)u(t) + D(q)v(t) + e(t) \tag{4}$$

In (4) those m inputs of u_{id} in (1) which are considered controlled are included in u, and those $m_{id} - m$ inputs which are considered measured but not controlled are included in v. Similarly, the values for the matrix polynomials B and D are extracted from the identified B_{id} in (1) so that those coefficients of B_{id} corresponding to the measured but not-controlled inputs are grouped into D and the remaining coefficients are left in B.

3.2 Calculating the control parameters for GPC

The GPC method aims to minimize the quadratic cost criterion J, given in (5), over a finite operating horizon given by N_1 and N_2, the minimum and maximum prediction horizon, respectively, and N_3, the control horizon.

$$J(N_1, N_2, N_3) = \sum_{j=N_1}^{N_2} \|\hat{y}(t+j|t) - w(t+j)\|_R^2 + \sum_{j=1}^{N_3} \|u(t+j-1) - u(t+j-2)\|_Q^2 \tag{5}$$

In (5), $\hat{y}(t+j|t)$ is the optimum j-step-ahead prediction of the output y, given the measurement data up to time t. $w(t+j)$ is the set point trajectory and R and Q are positive definite weight matrices that determine the proportional cost effect of the target error and control effort, respectively.

Following the procedure given in [20, 24] for obtaining the GPC parameters, the past (including present) and future terms of the controlled process are separated. By doing so and by adopting matrix notation the cost function (5) can be re-formulated as (6),

$$J = (G\Delta u + f + H\Delta v - w)^T \bar{R}(G\Delta u + f + H\Delta v - w) + \Delta u^T \bar{Q}\Delta u \tag{6}$$

from which the unconstrained optimal control Δu is obtained by differentiating and solving the equation with respect to Δu, resulting in (7).

$$\Delta u = (G\bar{R}G + \bar{Q})^{-1} G\bar{R}(w - f - H\Delta v) \tag{7}$$

Here the matrix $\Delta u = \left[\big(u(t) - u(t-1)\big)^T \quad \cdots \quad \big(u(t+N-1) - u(t+N-2)\big)^T\right]^T$ contains the control increments, $\Delta v = \left[\big(v(t) - v(t-1)\big)^T \quad \cdots \quad \big(v(t+N-1) - v(t+N-2)\big)^T\right]^T$, the disturbance changes and $w = [w(t)^T \quad \cdots \quad w(t+N-1)^T]^T$ the reference signal over the operating horizon N. $\bar{Q}$ and $\bar{R}$ are matrix forms of the weight factors Q and R, i.e. $\bar{Q} = diag(Q, \dots, Q)$ and $\bar{R} = diag(R, \dots, R)$. The operating horizon N is chosen by the operator and typically in practice $N = N_3 = N_2 - N_1 + 1$.

The matrices G, f and H are the control gain matrix, the uncontrolled system response signal and the disturbance gain matrix, respectively, over the operating horizon. These signals are considered to form the system response and therefore, the optimal process output prediction over the operating horizon can be given as (8). This equation must also be considered a basic assumption underlying the control strategy.

$$\hat{y} = G\Delta u + f + H\Delta v \tag{8}$$

G, f and H are obtained after reorganizing the problem into a part dependent only on current and future controls, and a part dependent on the past. Then, the Diophantine equation $I_{n\times n} = E_j(q)\big(A(q) - q^{-1}A(q)\big) + q^{-j}F_j(q)$ is solved for $j = 1 \dots N$ to obtain the intermediate polynomials E_j and F_j of orders $j-1$ and n_a, respectively.

The control gain and disturbance gain matrices, G and H, are obtained by multiplying B and D, respectively, with E_j and then re-grouping the obtained matrix polynomials, as given in (9)-(11).

$$E_j(q)B(q) = \sum_{i=0}^{j-1} G_i q^{-i} + q^{-j} f_{j,u}(q) \tag{9}$$

$$E_j(q)D(q) = \sum_{i=0}^{j-1} H_i q^{-i} + q^{-j} f_{j,v}(q) \tag{10}$$

$$G = \begin{bmatrix} G_0 & 0 & \cdots & 0 \\ G_1 & G_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ G_{N-1} & G_{N-2} & \cdots & G_0 \end{bmatrix}, \quad H = \begin{bmatrix} H_0 & 0 & \cdots & 0 \\ H_1 & H_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ H_{N-1} & H_{N-2} & \cdots & H_0 \end{bmatrix} \tag{11}$$

The elements G_i (all matrices) can be interpreted as step gains of the system with respect to the control signal. More specifically, $(G_i)_{k,j}$ is the value of the response measured from output k when a unit step excitation was fed to controlled input j of the system at instant $t - k$. A similar interpretation, but for the measured disturbances, is valid for H.

The uncontrolled response f is obtained by calculating (12)

$$\begin{aligned} f_j = & f_{j,u}(q)\big(u(t-j) - u(t-j-1)\big) \\ & + f_{j,v}(q)\big(v(t-j) - v(t-j-1)\big) \\ & + F_j(q)y(t) \end{aligned} \tag{12}$$

and collecting the f_j into $f = [f_1 \quad \cdots \quad f_N]^T$.

The calculation procedure is reflected by the result, (8), where it is seen that G and H operate on future, predicted or otherwise known, inputs and disturbances, and f contains the past effects. More specifically, f can be interpreted as the behavior of the system onward from time t if all controls and external disturbances were constant after the instant t.

The procedure is computationally efficient as it can be implemented recursively. A detailed description of the recursive computation algorithm is given in [24, Ch. 4, 6] and [20, Appendix B]. Furthermore, a means to obtain the parameters G and f directly from the model parameters A and B is given in [25].

3.3 Control constraints

One asset, if not the most important one, of model predictive control is the simplicity of introducing both input and output constraints on the control computation. On one hand, such constraints may be dictated by the actuator equipment, their operating range and operating speed, and on the other hand by the operator, who may wish to limit the system operation within pre-defined safety boundaries.

Absolute limits and slew rate limits for the manipulated variables are formulated simply as (13) and (14), respectively.

$$\underline{u} \leq u(t) \leq \overline{u}\,, \quad \forall t \tag{13}$$

$$\underline{\Delta u} \leq u(t) - u(t-1) \leq \overline{\Delta u}\,, \quad \forall t \tag{14}$$

Here $\underline{u}$ and $\overline{u}$ are the lower and upper limits for the inputs, and $\underline{\Delta u}$ and $\overline{\Delta u}$ the limits for the input change rate, respectively. Limits for the process output are given similarly as (15), where $\underline{y}$ and $\overline{y}$ denote the output lower and upper boundaries, respectively.

$$\underline{y} \leq y(t) \leq \overline{y}\,, \quad \forall t \tag{15}$$

In order to bring the constraints (13)-(15) into a form that can be implemented numerically, the output constraints are multiplied by suitable block unit matrices and the system output is estimated by the model output over the operating horizon, i.e., $y(t) \approx \hat{y} = G\Delta u + f$, over $t \dots t + N$. Furthermore, the two-sided constraints are all divided into two single-sided constraints. With these reformulations, all the constraints can be compactly given as a function of the control increments Δu as in (16)

$$S\Delta u \leq c, \tag{16}$$

where

$$S=\begin{bmatrix} T \\ -T \\ I_{N\times N} \\ -I_{N\times N} \\ G \\ -G \end{bmatrix},\quad c=\begin{bmatrix} 1_m\left(\overline{u}-u(t-1)\right) \\ -1_m\left(\underline{u}+u(t-1)\right) \\ 1_m\overline{\Delta u} \\ -1_m\underline{\Delta u} \\ 1_n\overline{y}-\mathrm{f} \\ -1_n\underline{y}+\mathrm{f} \end{bmatrix}. \tag{17}$$

In (17), T is a lower triangular block matrix of $N \times N$ blocks with $I_{m\times m}$ matrices in the non-null entries and $I_{N\times N}$ is an $N \times N$ unit matrix (identity matrix). 1_m and 1_n are $Nm \times m$ and $Nn \times n$ matrices formed by N $m \times m$ or $n \times n$ unit matrices, respectively (placed on top of each other). This formulation of the constraints yields the optimal control finding problem as a constrained convex optimization problem, which can be solved with very efficient routines.

3.4 Case-specific implementation

In our case, the system is considered time-invariant and a non-adaptive control can therefore be applied. Therefore, the process of solving E and F, and calculating G and H, is carried out only once and is done off-line and not at every time-step during the simulation. In practice this means that programming the algorithm into a process controller is simplified since none of the GPC parameter computation, (9)-(12), needs to be implemented there but the parameters can be imported from an external device.

The external disturbances v are considered measureable but not predictable or known in the future. Therefore they are considered constant over the prediction horizon, whereby the future disturbance increments $\Delta v = 0$. Therefore this assumption allows removing the $H\Delta v$ term from the control calculation (7). Note, however, that despite H is not used in the control calculation, it is calculated to obtain the past disturbance effects included in f.

An elementary part of the model predictive control scheme is the receding horizon approach. This means that only the first m rows of the matrix $(G\bar{R}G+\bar{Q})^{-1}G\bar{R}$ are needed to obtain the unconstrained control signal. By denoting these rows with K and substituting in (7) with the $H\Delta v$ term removed, the unconstrained control at time instant t obtains the form of an integrating control, guaranteeing zero steady-state error (18).

$$u(t)=\ K(w-f)+u(t-1) \tag{18}$$

Calculating $u(t)$ according to (18) is a simple matrix multiplication and requires no use of an iterative, numerical solver. Therefore, the approach taken in our implementation is that $u(t)$ is always calculated first by (18), then a constraint violation check is carried out with (16) and the constrained numerical optimization procedure is started only if the unconstrained control violates the constraints.

4 Results and discussion

4.1 Control simulation

In Sections 4.2-4.5, the behavior of the predictive control strategy based on GPC is evaluated by simulating the following four cases.

1) Control of a single output variable (T_{max}) with a single manipulated variable ($T_{air,in}$) when input constraints are present. GPC is compared with standard PID control with integrator anti-windup, and with and without feed-forward compensation. The purpose of this case is to illustrate how GPC handles constraints and that implementing GPC does *not* significantly improve the control result compared to a well-tuned PID controller in a simple, single-variable control case.
2) Control of a single output variable (T_{max}) with two manipulated variables ($T_{air,in}$, $\dot{V}_{air}$) when input constraints are present. The purpose of this case is to illustrate the practical effects of the control action cost parameters. Furthermore, the results are compared with Case 1 results to further analyze the practical aspects of multi-input control with respect to single-input control.
3) Control of two temperature variables (T_{max} and T_{min}) with the aim to reduce the temperature difference ($T_{max} - T_{min}$). No input constraints are used. The purpose is to analyze the feasibility of the control and to discuss implications of the control on system efficiency. In this case, two control approaches are studied:
   3.1) Using two manipulated variables ($T_{air,in}$ and $\dot{V}_{air}$).
   3.2) Using three manipulated variables ($T_{air,in}$, $\dot{V}_{air}$ and $\dot{V}_{NG}$).
4) Same as case 3.1, but now with the Kalman filter –based estimator, adopted from [21], used to estimate the stack temperature, while the cathode outlet temperature measurement is corrupted with noise. The purpose of this case is to demonstrate to functioning of the complete control implementation.

Tuning the model predictive controller is done by adjusting the cost function weights R and Q and the operating horizon parameters $N_{1...3}$.

The operating horizon is set to three hours (i.e. $N = 36$) with $N_2 = N_3 = 36$ and $N_1 = 1$ for all the simulation cases. The minimum prediction horizon ($N_1 = 1$) is determined based on the model dead-time (n_k), which for model 2 is 1 time step. The maximum prediction horizon is chosen by trial and error so that the prediction is considered sufficiently accurate over the whole horizon. The control horizon is set equal to the prediction horizon.

The cost function weights for input manipulation, Q, are set based on the system input-to-output characteristics which can be illustrated with the step response of each output to each input. These are shown for model 1 in Figure 3.

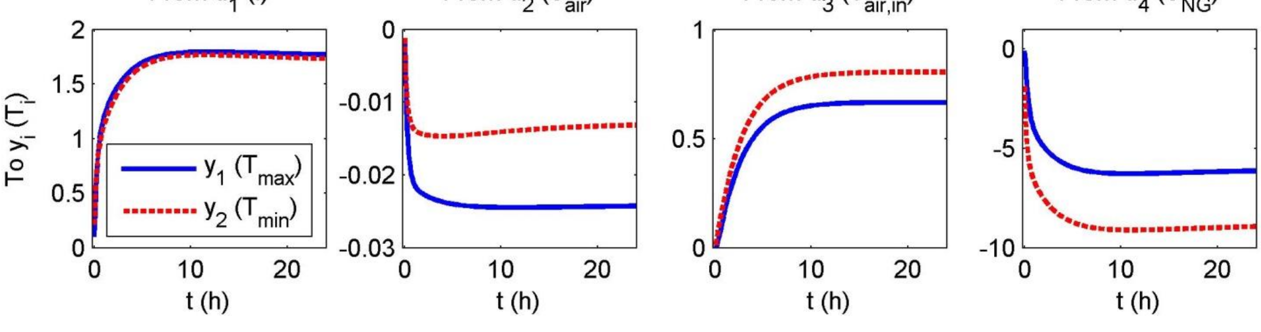

Figure 3 - Response of each output to a unit step change in each input.

Figure 3 enables determining two important system characteristics, namely, (i) what is the steady-state gain and (ii) what is the "response time" of each output to the change in each input. For example, it is seen that a unit change in input u_3 (corresponding to $T_{air,in}$) has a positive effect of ca. 0.75 units on output y_1 (corresponding to T_{max}) in steady-state. For u_2 ($\dot{V}_{air}$), this is ca. 0.025 units but in the opposite direction. Therefore, the figure enables determining whether or not, and to what extent the outputs can be individually controlled with the given input variables. It is also seen that although both outputs y_1 and y_2 respond towards the same direction for a change in a certain input, there are differences in the response magnitudes. These differences enable, to some extent, decoupled control of the individual outputs.

Based on the shape of the step response it is possible to estimate which inputs affect the outputs faster than others. For instance, it appears that a change in the air flow (u_2) alters the stack temperatures faster than a change in the inlet air temperature (u_3).

With this background, the cost function weights Q for the control effort are determined by the input variables' (average) proportional effect on the output variables, and the desired speed of the controlled response.

The target error costs, R, are determined based on the control aim and also the desired speed of the controlled response. In cases 1-2, only the first of the two output temperatures is controlled and thus all other elements of R except $R_{1,1}$ are zero. In case 3 both output temperatures are controlled with equal importance.

All simulation parameters are compiled into Table 3.

Table 3 - System set-up and the GPC algorithm parameters for the simulated test cases.

	Case 1	Case 2	Case 3.1 & Case 4	Case 3.2
Controlled outputs	T_{max}	T_{max}	T_{max} and T_{min}	T_{max} and T_{min}
Manipulated inputs	$T_{air,in}$	$\dot{V}_{air}$ and $T_{air,in}$	$\dot{V}_{air}$ and $T_{air,in}$	$\dot{V}_{air}$, $T_{air,in}$ and $\dot{V}_{NG}$
Measured disturbances	I			
N_1	1			
N_2	36			
N_3	36			
u	u_3	$[u_2 \quad u_3]^T$	$[u_2 \quad u_3]^T$	$[u_2 \quad u_3 \quad u_4]^T$
$\underline{u}$	-5	$[-50 \quad -5]^T$	–	–
$\overline{u}$	5	$[50 \quad 5]^T$	–	–
R	$\begin{bmatrix}1 & 0\\0 & 0\end{bmatrix}$	$\begin{bmatrix}1 & 0\\0 & 0\end{bmatrix}$	$\begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}$	$\begin{bmatrix}0.1 & 0\\0 & 0.1\end{bmatrix}$
Q	$[0.5]$	$\begin{bmatrix}0.03 \dots 0.3 & 0\\0 & 0.7 \dots 0.97\end{bmatrix}$	$\begin{bmatrix}0.015 & 0\\0 & 0.985\end{bmatrix}$	$\begin{bmatrix}0.002 & 0 & 0\\0 & 0.096 & 0\\0 & 0 & 0.902\end{bmatrix}$

All simulations and the control parameter calculation were implemented in the Matlab® and Simulink® environment. Constrained optimization was done with the Matlab® `fmincon` function. On a regular laptop computer the control parameter calculation (obtaining G, f and H) as well as simulation of the unconstrained control case occur instantaneously. The simulation of the constrained control case runs with a speed of ca. 3000 faster than real-time, which allows assuming that the algorithms are as such suited for online implementation in an embedded computation system.

4.2 Case 1: Single-variable GPC comparison with PID control

In the first case, the stack maximum temperature T_{max} is controlled by manipulating the stack module inlet air temperature $T_{air,in}$ (corresponding to the output and input variables y_1 and u_3, respectively). In Figure 3 it is seen that the relationship between this I/O pair is very close to a first-order process without significant dead-time and the process is easily controlled by a well-tuned PID controller.

Typically the inputs of a SOFC system are constrained to some pre-defined range which can have significant effects on the practical control behavior. The model predictive controller handles the input constraints as part of the control calculation, and an integrator anti-windup scheme is typically used in PID controllers to avoid problems resulting from limited actuation.

In the simulation case, the GPC response is compared with the response obtained by PID control with integrator anti-windup, with and without a first-order feedforward compensation for the measureable load disturbance. The PID and FF control algorithms as well as the parameters are given in Appendix A.

The system is excited first by a positive unit step disturbance in stack load current, posed on the system at t = 2 hours (I changes from 160A to 161A, Figure 4 (a)). Then at t = 10 hours a negative unit step change in the T_{max} setpoint is carried out (i.e. the T_{max} setpoint changes from 773.3 °C to 772.3 °C, Figure 4 (c)). Throughout the simulation, the manipulated variable is constrained between ±5°C from the nominal value (i.e. $T_{air,in} \in [730, 740]$ °C) and the two unused inputs, $\dot{V}_{air}$ and $\dot{V}_{NG}$ are kept at their nominal values (Table 1).

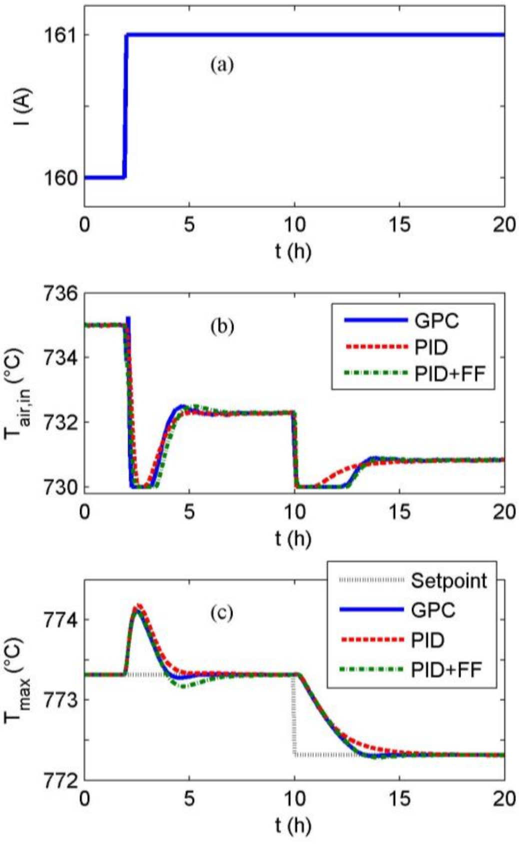

Figure 4 – Case 1, illustration of the GPC algorithm performance compared to a PID controller with and without feedforward (FF) compensation for measured disturbances. The figures show (a) the disturbance signal, (b) the manipulated input and (c) the controlled output and its setpoint.

Figure 4 (c) shows the controlled responses. Both for the disturbance and the setpoint change, the speed of the controlled response is clearly limited by the input constraint. Although the GPC output appears slightly better performing than the PID outputs, the difference is not significant enough to justify the extra complexity of the GPC algorithm. The result illustrates, however, how GPC inherently contains the features which are brought to the PID controller by adding feedforward compensation and integrator anti-windup.

4.3 Case 2: Multivariable control of cell maximum temperature

Using multiple manipulated input variables, instead of only one, to control the output enables improving the control and response characteristics. In particular, as the system response to some manipulated inputs is faster than to others (Figure 3), the settling time of the controlled system response can be affected by selection of the cost function parameters. Similarly, the differences between the input-output pairs' steady state gain can be utilized to adjust the steady-state cost of the control effort. In the following simulations, T_{max} is controlled by manipulating $T_{air,in}$ and $\dot{V}_{air}$ with two different values of Q, i.e. the control effort cost weighting coefficients.

Based on Figure 3 it may be expected that if $\dot{V}_{air}$ is used to control the stack temperature, the controlled response is faster than if $T_{air,in}$ is used. The effect can be examined by varying the proportional cost of these two manipulated variables. The control effort weights used in the following simulations are $Q_1 = \begin{bmatrix} 0.03 & 0 \\ 0 & 0.97 \end{bmatrix}$, $Q_2 = \begin{bmatrix} 0.3 & 0 \\ 0 & 0.7 \end{bmatrix}$. The sum of the weighting factors in both Q_1 and Q_2 is 1, and in Q_1, the weight coefficients for $T_{air,in}$ and $\dot{V}_{air}$ correspond to the steady-state gain of their unit step responses (seen in Figure 3). If using Q_2 the proportional cost of manipulating $\dot{V}_{air}$ is ten-fold compared to using Q_1.

The disturbance and setpoint signals as well as the uncontrolled inputs are identical to those in case 1 (Section 5.2). The constraints for $T_{air,in}$ are the same as in case 1 (i.e. $T_{air,in} \in [730, 740]$ °C) and for $\dot{V}_{air}$ a maximum variation of 50 dm^3 min^{-1} from the nominal value is set (i.e. $\dot{V}_{air} \in [1012, 1112]$ dm^3 min^{-1}). Furthermore, a linear trend of 1°C / 30 hours (33 °C / 1000 h) is added to the process output to simulate the control behavior under degradation effects. The trend is heavily exaggerated for illustrative purposes when comparing to the trend of ca. 5-7 °C / 1000 hours observed in the experiments [21].

The simulation case manipulated inputs and the controlled output are shown in Figure 5. The manipulated inputs and controlled output of the GPC control in case 1 are shown for comparison. Note though, that no degradation trend was present in case 1.

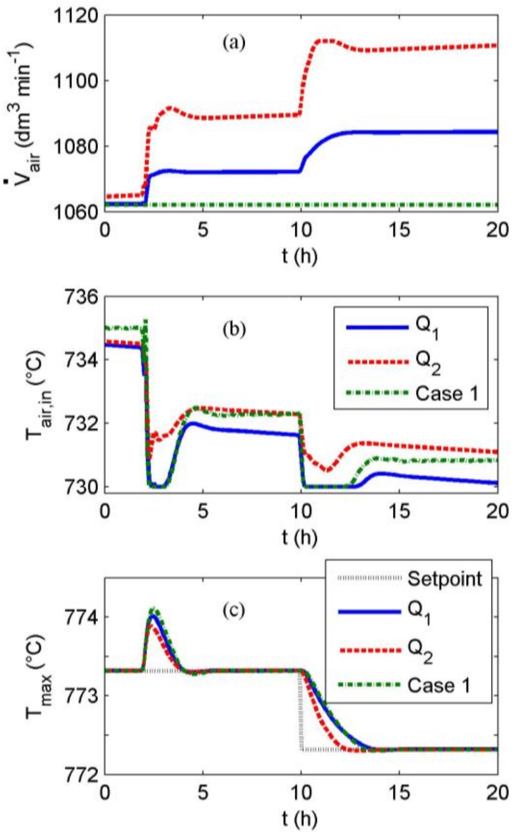

Figure 5 – Case 2, illustration of the GPC algorithm performance when the stack maximum temperature is controlled by manipulating both air flow and air inlet temperature. A linear trend is added to the process output to simulate effects of stack performance degradation. Also the single-input control result (case 1) is shown for comparison.

The primary observation of Figure 5 (c) is that the controlled response is slightly faster with Q_2, when the cost for manipulating $\dot{V}_{air}$ is reduced, compared to Q_1. It is also seen in Figure 5 (c) that the controlled response settles slightly faster and deviates less from the desired value in case 2 than in case 1.

The trend added to the process model output results in a continuously changing control signal – the control adjusts the manipulated variables continuously in order to maintain the controlled variable at its desired value.

The effects of modifying the weights in Q become evident by observing the manipulated variables in Figure 5 (a)-(b). If the cost of manipulating $\dot{V}_{air}$ is high, its magnitude is smaller and $T_{air,in}$ is used proportionally more to obtain the desired output, and vice-versa. Therefore, with Q_1 it is observed that $T_{air,in}$ becomes limited by the input constraint and the same happens for $\dot{V}_{air}$ with Q_2.

Remark – It is clear that modifying $\dot{V}_{air}$ will also affect the parasitic power consumption of the air blower and so the system efficiency. Modifying $T_{air,in}$, however, takes place by changing the opening of an air heat-exchanger by-pass valve which has little or no effects on the system efficiency. With the formulation of the cost function in (5) only the control increments affect the cost and so it is not possible to directly optimize the control with respect to the absolute magnitude of the control effort. For such optimization, (5) should be modified, e.g., to include a cumulative sum of the control actions which cause for system efficiency to decrease, in this case $\dot{V}_{air}$.

4.4 Case 3: Control of cell maximum and minimum temperature

Reducing, the temperature difference over the cells in the stack could be beneficial considering attenuation of thermal stresses in the stack and on the ceramic cell structure. Next, two approaches to control both T_{max} and T_{min} and thereby the difference $T_{max} - T_{min}$ are simulated in the following cases:

3.1) By using two manipulated variables: $T_{air,in}$ and $\dot{V}_{air}$.

3.2) By using three manipulated variables: $T_{air,in}$ $\dot{V}_{air}$ and $\dot{V}_{NG}$.

It is clear that since essentially two variables are controlled, at least two manipulated variables are necessary. Furthermore, when $\dot{V}_{NG}$ and $\dot{V}_{air}$ are manipulated to control the stack temperature, also the electrical efficiency of the system is affected; $\dot{V}_{NG}$ relates to the efficiency directly and $\dot{V}_{air}$ affects the parasitic power consumption of the air blower. However, if a longer system lifetime is obtained by a moderate reduction in efficiency, then it may be worthwhile.

The disturbance signal and the uncontrolled inputs are identical to cases 1-2. Also the trend is included in the system output (as in case 2). The setpoint signal for T_{max} is also identical to cases 1-2 but the setpoint for T_{min} is kept constantly at its nominal value (692.9 °C) in order to simulate reduction of the temperature difference. The cost function weights are given in Table 3 and the input constraints are removed to allow for larger control actions.

Figure 7Figure 6 (a)-(c) shows the behavior of the manipulated variables obtained with the two different control approaches. The resulting outputs are shown in Figure 7 (a)-(b).

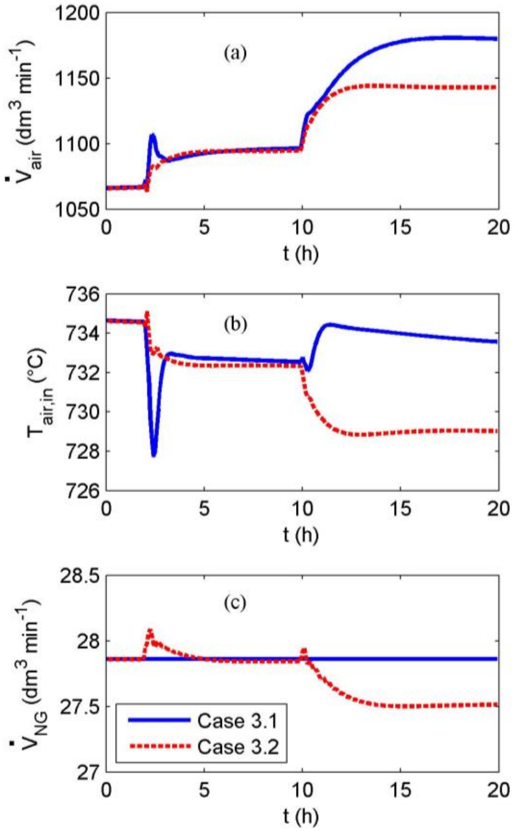

Figure 6 – Cases 3.1-3.2, the manipulated input variables.

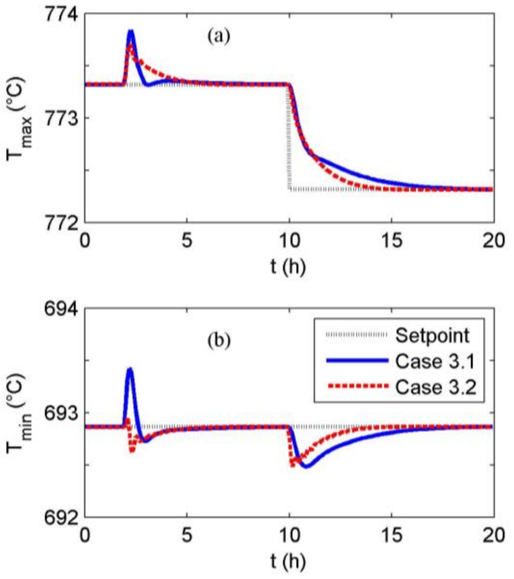

Figure 7 – Cases 3.1-3.2, the controlled response.

The primary observation on Figure 6 and Figure 7 is that it appears to be possible to regulate and to reduce the temperature difference over a cell in the SOFC stack by using multivariable control. If comparing the results with cases 1-2, it is also seen that enforcing both temperature variables to be controlled requires in general for larger control actions to be carried out. However, due to the large gain from the NG feed input to the stack temperature (Figure 3) only a rather modest manipulation of the NG feed is necessary to affect the stack temperature.

An interesting observation is that in case 3.2, a reduction of the temperature difference is obtained with a control action where the NG flow is decreased and, furthermore, so that also the air flow ends at a lower level than in case 3.1, where the NG flow is kept constant. Therefore, if fuel and/or air starvation do not pose practical constraints, the three-input control enables both reducing the temperature differences in the stack and improving the system efficiency at the same time. To interpret this in another way, in order to obtain the desired stack operating temperature and temperature distribution, it is best to operate the system at minimal feasible air and fuel flow and manipulate only the air inlet temperature (instead of manipulating also the air flow).

4.5 Case 4: Simulation with estimator included

To illustrate the functioning of the complete control system, the Kalman filter –based estimator for the stack temperatures was added to the simulation case 3.1. Band-limited white noise with a standard deviation of 0.5 °C and a random drift with a drift velocity of ca. 2 °C/h were added to the cathode outlet temperature measurement. Both noise properties are exaggerated for illustrational purposes. Values observed in the raw data for the temperature measurement are below 0.3 °C for the standard deviation and below 0.1 °C/h for the drift velocity. Also the linear degradation trend of 1°C / 30 h, used in cases 2-3, is applied to simulate exaggerated degradation of the stack performance.

In Figure 8 it is seen that the control is similar to that obtained in case 3.1. Also the controlled outputs shown in Figure 9 are otherwise the same in case 4 as in case 3.1, but effects of the measurement noise are carried along to the control also, although the Kalman filter removes much of the noise (as seen in Figure 10). Although the noise properties were heavily exaggerated in the simulation case, it is apparent that if the application case is significantly noisy, filtering both the measurement and output signals may be necessary.

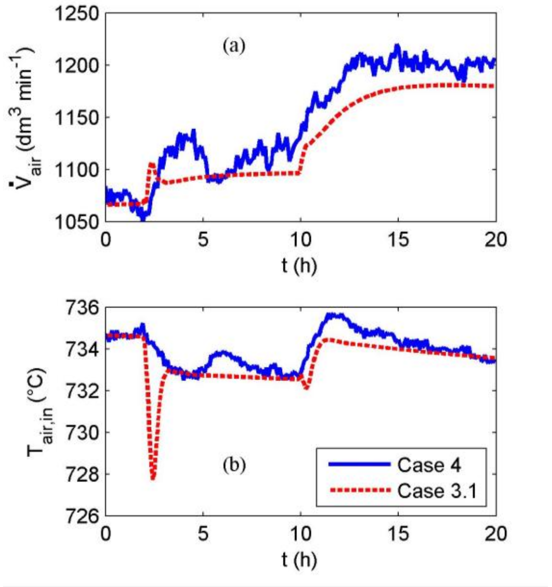

Figure 8 – Cases 4 and 3.1, the manipulated input variables.

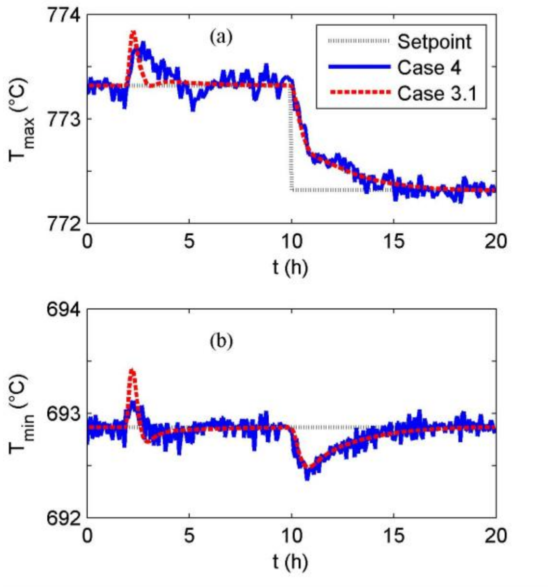

Figure 9 – Cases 4 and 3.1, the controlled output. Note that in case 4 the plotted variable is the estimator output whereas in case 3.1 it is the model output.

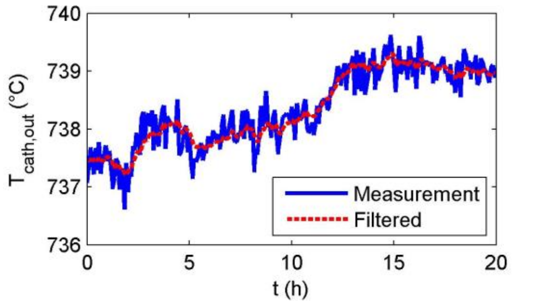

Figure 10 – Case 4, the cathode outlet temperature measurement before and after filtering.

4.6 Discussion

In this work the GPC parameters were not tuned in any particular manner as long as a stable output was obtained. Tuning of model predictive controllers, i.e. setting the operating horizon N and the cost function weights R and Q, is in general not straightforward and may require several iterations by trial and error. An analysis of the process input/output relations and the corresponding steady-state gains, as given with Figure 3, can be a useful starting point as this way, proportionally equal weights for the control cost function are obtained easily. From this starting point, the practical implications of each input and output on the real system operating cost, such as its efficiency or lifetime, can be used to further tune the cost weights. Once suitable parameters for some reference case are found, it should be confirmed that the system is stable with these parameters over the full desired operating range. To this end, a theoretical analysis of the model predictive controller, in addition to extensive simulation, should be carried out. After such an analysis, MPC tuning rules reported in the literature may be applicable [27].

Setpoint prediction or prediction of measureable disturbances in the GPC algorithm was also left unutilized in this work. Although it is unrealistic to assume prediction of disturbances, however measureable, the setpoints may well be made predictable by introducing a lag to setpoint changes. Furthermore, considering e.g. residential SOFC-based CHP systems, where system operation is not ultimately time-critical or can be configured to follow a pre-determined daily operating cycle, setpoint prediction as well as prediction (or scheduling) of the measureable disturbances could be reasonably implemented.

Considering the multivariable control in cases 2-4, system efficiency should be taken into account when designing the control so that the desired tradeoff between system efficiency and output controllability is obtained. Clearly, the cost function weights for the control parameters and plain input constraints provide a tool for obtaining such a control which is both technically applicable and economically viable. Additionally, in case 2, when only T_{max} is controlled, and in case 3, when approach 2 is applied, there are more manipulated variables than controlled variables and the system is *overdefined*. In these cases it should be examined by input-output pairing whether or not it is really worthwhile to utilize all the available manipulated variables for control.

The simulations in this work were based on a model identified from experimental data from a real SOFC system which allows assuming that GPC can also be utilized for SOFC control in practice. Although this work focused on a single-stack system, the need for a centralized control method, such as GPC, increases especially when large SOFC systems containing several stacks are built. In such multi-stack systems the differences in performance between the stacks have to be compensated for somehow in order to maximize system lifetime and/or efficiency [28]. Managing the temperature of the stacks becomes especially important even during steady state operation as the stacks' temperature affects also the fuel and air distribution between the stacks when they share common gas flow manifolds. Considering e.g. a compensation strategy by stack-wise load modulation or flow adjustment, it is practically impossible to implement an optimal compensation by giving setpoints to individual stack load controllers, while a centralized control does this automatically.

5 Conclusions

A simple example of model predictive control, based on the generalized predictive control algorithm for the temperature control of a SOFC stack was presented. The models utilized for control development were identified from data obtained with a series of experiments carried out on a complete 10 kW SOFC system.

The simulations illustrate that with the MIMO GPC algorithm it is possible to regulate and to control the temperature difference over a cell in a SOFC stack. This enables the reduction of thermal stresses over the cell which may be expected to have a positive effect on stack lifetime.

The GPC algorithm was compared with a standard PID controller for controlling a single stack temperature variable (the stack maximum temperature) with one and two manipulated input variables. Although the controlled response can be made slightly faster by using GPC and multiple manipulated inputs, the PID controller generally provides a comparable performance to the GPC algorithm when only a single temperature is controlled. Therefore, despite its flexibility regarding constraint handling, the justification for using GPC which is significantly more complex than PID control is questionable in a simple control case.

The operation of a complete MIMO GPC control algorithm with a Kalman filter –based estimator for the controlled, non-measureable variable was studied under noise effects. The computational performance of the algorithm was found viable for embedded online implementation.

Future work includes validating the approach in a complete SOFC system and including temperature control for startup and shutdown procedures, occurring outside the nominal operating envelope.

Acknowledgements

This work was carried out under the RealDemo project for which the funding was obtained from the Finnish Agency of Technology and Innovation TEKES, VTT and the participating industrial companies. Their contribution is gratefully acknowledged.

Glossary

ARX	Autoregressive (model) with extra input
CARIMA	Controlled and autoregressive (model) with integrated moving average (noise term)
FF	Feedforward (compensation)
GPC	Generalized predictive control
I/O	Input-output
LTI	Linear and time-invariant
MIMO	Multiple-input-multiple-output
MPC	Model predictive control
NG	Natural gas
NOC	Nominal operating conditions
PID	Proportional-integral-derivative
SOFC	Solid oxide fuel cell

References

[1] R. Knibbe, A. Hauch, J. Hjelm, S. Ebbesen and M. Mogensen, Durability of Solid Oxide Cells, *Green*, **1**, 141-169, 2011

[2] A. Nakajo, Z. Wuillemin, J. Van herle and D. Favrat, Simulation of thermal stresses in anode-supported solid oxide fuel cell stacks. Part I: Probability of failure of the cells, *J. Power Sources*, **193**, 203-215, 2009

[3] Z. Wuillemina, A. Nakajo, A. Müller, A. J. Schuler and S. Diethelm, Locally-Resolved Study of Degradation in a SOFC Repeat-Element, *ECS Trans.*, **25**(2), 457-466, 2009

[4] R.S. Gemmen, Dynamic modeling of fuel cells, Ch. 9 in Modeling Solid Oxide Fuel Cells, R. Bove and S. Ubertini (Eds.), Springer, 2008

[5] D. Bhattacharyya and R. Rengaswamy, A review of solid oxide fuel cell (SOFC) dynamic models, *Ind. & Eng. Chem. Res.*, **48**, 6068-6086, 2009

[6] F. Mueller, F. Jabbari, J. Brouwer, On the intrinsic transient capability and limitations of solid oxide fuel cell systems, *J. Power Sources*, **187**, 452-460, 2009

[7] Y.H. Li, S.S. Choi, S. Rajakaruna, An analysis of the control and operation of a solid oxide fuel-cell power plant in an isolated system, *IEEE Trans. Energy Conversion*, **20**(2), 381-387, 2005

[8] X. Wang, B. Huang, T. Chen, Data-driven predictive control for solid oxide fuel cells, *J. Process Control*, **17**, 103—114, 2007

[9] H.-B. Huo, X.-J. Zhu, W.-Q. Hu, H.-Y. Tu, J. Li, J. Yang, Nonlinear model predictive control of SOFC based on a Hammerstein model, *J. Power Sources*, **185**, 338-344, 2008

[10] T.L. Vincent, B. Sanandaji, A.M Colclasure, H. Zhu, R.J. Kee, Physically Based Model-Predictive Control for SOFC Stacks and Systems, *ECS Trans.*, **25**(2), 1175-1184, 2009

[11] T. Allag, T. Das, Robust control of solid oxide fuel cell ultracapacitor hybrid system, *IEEE Trans. Control System Technology*, **20**(1), 2012

[12] B.J. Spivey, T.F. Edgar, Dynamic modeling, simulation and MIMO predictive control of a tubular solid oxide fuel cell, *J. Proc. Control*, **22**, 1502-1520, 2012

[13] Y. Komatsu, S. Kimijima, J.S. Szmyd, Numerical analysis on dynamic behavior of solid oxide fuel cell with power output control scheme, *J. Power Sources*, **223**, 232-245, 2013

[14] P. Aguiar, C.S. Adjiman, N.P. Brandon, Anode-supported intermediate-temperature direct internal reformin solid oxide fuel cell II. Model-based dynamic performance and control, *J. Power Sources*, **147**, 136-147, 2005

[15] H.-B. Huo, Y.-X. Wu, Y.-Q. Liu, S.-H. Gan, X.-H. Kuang, Control-oriented nonlinear modeling and temperature control for solid oxide fuel cell, *J. Fuel Cell Sci. Tech.*, **7**, 2010

[16] M. Fardadi, F. Mueller, F. Jabbari, Feedback control of solid oxide fuel cell spatial temperature variation, *J. Power Sources*, **195**, 4222-4233, 2010

[17] M. Sorrentino, C. Pianese, Model-based development of low-level control strategies for transient operation of solid oxide fuel cell systems, *J. Power Sources*, **196**, 9036-9045, 2011

[18] M. Fardadi, D.F. McLarty, F. Jabbari, Actuator limitations in spatial temperature control of SOFC, *J. Fuel Cell Sci. Tech.*, **10**, 2013

[19] M.J. Kupilik, T.L. Vincent, Control of a solid oxide fuel cell system with sensitivity to carbon formation, *J. Power Sources*, **222**, 267-276, 2013

[20] D.W. Clarke, C. Mohtadi, P.S. Tuffs, Generalized predictive control – Part I. The basic algorithm, *Automatica*, **23**(2), 137-148, 1987

[21] A. Pohjoranta, M. Halinen, J. Pennanen, J. Kiviaho, SOFC stack temperature estimation with data-based models –Designed experiments and parameter identification, *J. Power Sources.*, online 2014-09-16, 2014-08-11, DOI: 10.1016/j.jpowsour.2014.08.130

[22] M. Halinen, M. Rautanen, J. Saarinen, J. Pennanen, A. Pohjoranta, J. Kiviaho, M. Pastula, B. Nuttall, C. Rankin and B. Borglum, Performance of a 10 kW SOFC Demonstration Unit Stacks and Systems, *ECS Trans.*, **35**(1), 113-120, 2011

[23] B. Borglum, E. Tang, M. Pastula, Development of solid oxide fuel cells at Versa Power Systems, *ECS Trans.*, **35**(1), 63-69, 2011

[24] E.F. Camacho, C. Bordons, Model predictive control, 2nd ed., Springer, 2007

[25] Q. Song, F. Liu, The direct approach to unified GPC based on ARMAX/CARIMA/CARMA model and application for pneumatic actuator control, *Proc. ICICIC'06, IEEE*, 2006

[26] L. Ljung, System identification - Theory for the user, 2nd ed., Prentice Hall PTR, 1999

[27] J.L. Garriga, M. Soroush, Model predictive control tuning methods: a review, *Ind. Eng. Chem. Res.*, **49**, 3505-3515, 2010

[28] M. Noponen, T. Korhonen, Effects of multiple stacks with varying performances in SOFC system, *Proc. 10th European SOFC Forum*, Lucerne (CH), A13 – 17-26, 2012

[29] K.J. Åström, T. Hägglund, Advanced PID Control, ISA – Instrumentation, Systems, and Automation Society, 2006

Appendix A – The discrete PID controller and FF compensator algorithms

The discrete-time PID controller algorithm used in the Section 3 simulations is given below. The algorithm is the common discrete-time PID approximation obtained with backward differencing and the integrator anti-windup is implemented with the back-calculation method, both found e.g. in [29]. In (A.1)-(A.4), $e(t)$ is the error signal $(w(t) - y(t))$, K_P, K_I and K_D are the control parameters, i.e. the gains for the proportional, integral and derivative parts of the control signal, respectively. N_f is a filtering coefficient (integer) for the derivative part. The given notation assumes a discretization time-step of 1.

In all simulations the parameters have the following values: K_P = 2.5, K_I = 0.25, K_D = 1000 and N_f =1. The parameter values were found by trial and error.

$$u_{PID}(t) = u_P(t) + u_I(t) + u_D(t) \tag{A.1}$$

$$u_P(t) = K_P e(t) \tag{A.2}$$

$$u_I(t) = u_I(t-1) + K_I e(t) + \left(\max\left(\underline{u}, \min\left(\overline{u}, u_{PID}(t-1)\right)\right) - u_{PID}(t-1)\right) \tag{A.3}$$

$$u_D(t) = \frac{K_D}{K_D + K_P N_f} u_D(t-1) + \frac{K_D K_P N_f}{K_D + K_P N_f}\left(e(t) - e(t-1)\right) \tag{A.4}$$

The feedforward compensation for load disturbance is implemented using a first-order compensator (A.5) whose parameters are identified from the measurement data. It is seen from the compensator parameters that not much disturbance rejection can be obtained by adding the compensator which is apparent also based on the simulation results.

$$u_{FF}(t) = 0.9973 u_{FF}(t-1) + u_1(t) - 0.9981 u_1(t-1) \tag{A.5}$$