10.2478/acsc-2014-0005

Archives of Control Sciences
Volume 24(LX), 2014
No. 1*, pages 67–84*

Application of adaptive multivariable Generalized Predictive Control to a HVAC system in real time

MARTIN GULAN, MICHAL SALAJ and BORIS ROHAL'-ILKIV

This paper presents the application of a Multivariable Generalized Predictive Controller (MGPC) for simultaneous temperature and humidity control in a Heating, Ventilating and Air-Conditioning (HVAC) system. The multivariable controlled process dynamics is modeled using a set of MISO models on-line identified from measured input-output process data. The controller synthesis is based on direct optimization of selected quadratic cost function with respect to amplitude and rate input constraints. Efficacy of the proposed adaptive MGPC algorithm is experimentally demonstrated on a laboratory-scale model of HVAC system. To control the air-conditioning part of system the designed multivariable predictive controller is considered in a cascade dual-rate control scheme with PID auxiliary controllers.

Key words: predictive control, multivariable systems, constrained optimization, HVAC system, cascade control

1. Introduction

With regard to the world-wide trend in energy saving it is of a great importance to study and develop new, modern and intelligent methods of building control. A very common control problem forms creating of a thermal comfort in heated and air-conditioned rooms with a minimal energy input. This task brings many problems such as variableness and non-linearity of the controlled process, and presence of disturbances, mainly of random character. In this case classic methods cannot ensure optimal control. An employment of modern control algorithms is one alternative.

Previous research projects have established a foundation of process models and control strategies suitable for building HVAC systems. Good results were obtained with application of PID auto-tuning controllers [19], neural network techniques [3], and even

The Authors are with Institute of Automation, Measurement and Applied Informatics, Faculty of Mechanical Engineering, Slovak University of Technology in Bratislava, Nám. slobody 17, 812 31 Bratislava, Slovakia. Email: martin.gulan@stuba.sk

The paper was created with the support of the Ministry of Education, Science, Research and Sport of the Slovak Republic within the Research and Development Operational Programme for the project "University Science Park of STU Bratislava", ITMS 26240220084, co-funded by the European Regional Development Fund.

Received 20.12.2013. Revised 3.03.2014.

genetic algorithms [13]. In [15] an application of SISO-MISO self-tuning control of environmental conditions is proposed. The possible use of the control algorithm was tested using the data obtained from the non-linear mathematical model in which interaction between the variables of relative humidity and temperature is considered.

Pioneered by [12], GPC belongs to a wide class of Model Predictive Control (MPC) methods. It incorporates all major features of the predictive controllers in an unified framework and offers advantages such as ability to stabilize and control nonminimum-phase, variable or unknown dead-time, open-loop unstable processes and also plants with unknown orders. Over the years, GPC has proven to be very effective when requirements on the robustness and performance are hard to achieve with traditional designs [17].

This paper deals mainly with the extension of the GPC design to the multivariable case which is the nature of most industrial processes. Multivariable or multiloop GPC takes into consideration interactions between loops what improves control performance [2, 5], unlike e.g. the use of multiple single loop controllers, mostly PID, which is the simplest, however often not satisfactory solution. Hence to simplify the procedure of deriving of prediction equations a strategy of MIMO model decomposition to a set of multi-input single-output (MISO) CARIMA input-output models is proposed.

The case study developed in this work focuses on a possible application of predictive control methods, namely GPC, to control of air-conditioning processes. The implementation of the proposed MGPC controller into the process is realized by a cascade scheme. The inner loops of the scheme governed by PSD controllers, so-called servo loops, ensure desired settings of regulated variables (e.g. flow of the vapour from humidifier, or temperature of the air at the heater output). The control algorithm makes use of an incremental CARIMA model in which interaction between the variables of relative humidity and temperature is considered. The properties of the tested control algorithm were supported by a robust recursive parameter estimation scheme used for the on-line identification of the plant model. The resulting system stability is guaranteed via choice of a reasonably long prediction horizon. Experimental verification of the proposed control algorithm was executed on a HVAC plant constructed at the authors' department.

2. Multivariable Generalized Predictive Control

The objective of the predictive control law is to drive future plant outputs as close to the reference value as possible. Thus, a predictive controller is based on the following concepts [18]:

- using a representative mathematical model of the plant dynamics obtained by physical modelling or a system identification procedure,
- predicting future plant dynamics,
- expressing the process optimality by an appropriate cost function of the future errors and controls,

- predicting cost of future plant dynamics,
- minimizing the cost function (optimal control),
- and finally applying a receding horizon control strategy.

The main idea of a generalized predictive control algorithm based on an on-line identification procedure is schematically illustrated in Fig. 1. As shown, GPC can be thus viewed as a form of a feedback control algorithm, where instead of a fixed feedback law a dynamic optimization process determines inputs based on actual measurements [18].

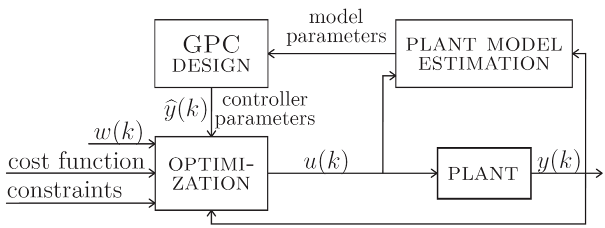

Figure 1. Block diagram of an indirect adaptive GPC controller.

In the following subsections an approach to the design of an adaptive multivariable GPC controller is presented.

2.1. Multivariable CARIMA model and objective function

Let the controlled process to be a m-input, n-output linear system which can be described by a discrete MIMO model. To satisfy the control objective in GPC, a multivariable CARIMA (Controlled Auto-Regressive Integrated Moving Average) process model is utilized to represent the plant dynamics. To ensure effective integral action in a predictive control law, its incremental form

$$\mathcal{A}(q^{-1})\Delta\mathbf{y}_k = q^{-d}\mathcal{B}(q^{-1})\Delta\mathbf{u}_{k-1} + \mathcal{C}(q^{-1})\boldsymbol{\xi}_k \tag{1}$$

is used, as it allows offset free predictions in the steady state [16]. This can be subsequently rewritten as

$$\mathcal{A}(q^{-1})\mathbf{y}_k = q^{-d}\mathcal{B}(q^{-1})\mathbf{u}_{k-1} + \frac{\mathcal{C}(q^{-1})}{\Delta}\boldsymbol{\xi}_k, \tag{2}$$

where quantities $\mathbf{y}_k = [y_1(k),\ldots,y_n(k)]^T \in \mathbb{R}^n$, $\mathbf{u}_k = [u_1(k),\ldots,u_m(k)]^T \in \mathbb{R}^m$ and $\boldsymbol{\xi}_k = [\xi_1(k),\ldots,\xi_n(k)]^T \in \mathbb{R}^n$ are the output, input and zero-mean white noise vectors, respectively, $\Delta = 1 - q^{-1}$ is the difference operator and d is the dead time[1] of the process

[1]For simplicity in the development, d is chosen to be zero.

dynamics. $\mathcal{A} \in \mathbb{R}^{n\times n}$, $\mathcal{B} \in \mathbb{R}^{n\times m}$ and $\mathcal{C} \in \mathbb{R}^{n\times n}$ are adequate polynomial matrices in backward shift operator q^{-1}, whose elements are polynomials, i.e.:

$$\begin{aligned} A_{ii}(q^{-1}) &= 1 + (a_1)_{ii}q^{-1} + \cdots + (a_{n_{A_{ii}}})_{ii}q^{-n_{A_{ii}}} \\ B_{il}(q^{-1}) &= (b_0)_{il} + (b_1)_{il}q^{-1} + \cdots + (b_{n_{B_{il}}})_{il}q^{-n_{B_{il}}} \\ C_{ii}(q^{-1}) &= 1 + (c_1)_{ii}q^{-1} + \cdots + (c_{n_{C_{ii}}})_{ii}q^{-n_{C_{ii}}} \quad i = 1,2,\ldots,n, \quad l = 1,2,\ldots,m \end{aligned} \tag{3}$$

with $n_{A_{ii}} = \delta(A_{ii})$, $n_{B_{il}}$, $n_{C_{ii}}$ indicating their respective order. In the following, the argument q^{-1} is dropped for brevity.

As it will be demonstrated later, by applying MGPC such a control sequence can be obtained that drives the future outputs as close as possible to a reference within defined horizon. This is attained by minimization of a multistage cost function of the following form:

$$J(k) = \sum_{j=N_{1_i}}^{N_{2_i}} \sum_{i=1}^{n} \delta_i(j)\left[\widehat{y}_i(k+j \mid k) - w_i(k+j)\right]^2 + \sum_{j=1}^{N_{u_l}} \sum_{l=1}^{m} \lambda_l(j)\left[\Delta u_l(k+j-1)\right]^2 \tag{4}$$

The first term in (4) corresponds to the difference between an optimal predicted sequence of the system output $\widehat{y}(k+j \mid k)$ conditioned on data known up to a discrete time (k) and a future set-point sequence $w(k+j)$ for controlled variables. The second term consists of a weighted sequence of the future control input increments $\Delta u(k+j-1)$ obtained from the minimization of the finite horizon quadratic criterion (4). N_{1_i} and N_{2_i} are the minimum and maximum prediction horizons for the scalar output y_j, and N_{u_l} is the control horizon over all future inputs u_j, which affect the outputs included in J. To penalize future tracking errors and control increments along the horizons, appropriate weighting sequences $\delta_i(j)$ and $\lambda_l(j)$ are defined as well. Eq. (4) can be equivalently[2] expressed as:

$$J(k) = \sum_{j=N_1}^{N_2} \|\widehat{\mathbf{y}}_{k+j|k} - \mathbf{w}_{k+j}\|^2_{\mathbf{Q}_\delta(j)} + \sum_{j=1}^{N_u} \|\Delta\mathbf{u}_{k+j-1}\|^2_{\mathbf{Q}_\lambda(j)} \tag{5}$$

or in a more condensed vector/matrix form

$$J = \|\, \underset{\rightarrow}{\widehat{\mathbf{y}}} - \underset{\rightarrow}{\mathbf{w}}\|^2_{\mathbf{Q}_\delta} + \|\Delta\, \underset{\rightarrow}{\mathbf{u}}\|^2_{\mathbf{Q}_\lambda}, \tag{6}$$

[2]Let us assume that prediction/control horizons for particular outputs/inputs are equal, i.e. $N_2 = N_{2_i}$, $N_1 = N_{1_i}$, $N_u = N_{u_l}$, $\forall i \in [1,n], \forall l \in [1,m]$.

where the augmented vectors[3]

$$
\begin{aligned}
\underset{\rightarrow}{\widehat{\mathbf{y}}} &= \left[\widehat{\mathbf{y}}_{k+N_1}^T \ \dots \ \widehat{\mathbf{y}}_{k+N_2}^T\right]^T \in \mathbb{R}^{n(N_2-N_1+1)} \\
\underset{\rightarrow}{\mathbf{w}} &= \left[\mathbf{w}_{k+N_1}^T \ \dots \ \mathbf{w}_{k+N_2}^T\right]^T \in \mathbb{R}^{n(N_2-N_1+1)} \\
\Delta \underset{\rightarrow}{\mathbf{u}} &= \left[\Delta\mathbf{u}_k^T \ \dots \ \Delta\mathbf{u}_{k+N_u-1}^T\right]^T \in \mathbb{R}^{mN_u}
\end{aligned}
\tag{7}
$$

can be obtained by stacking the sub-vectors

$$
\begin{aligned}
\widehat{\mathbf{y}}_{k+j|k} &= \left[\widehat{y}_1(k+j \mid k) \ \dots \ \widehat{y}_n(k+j \mid k)\right]^T \in \mathbb{R}^n \\
\mathbf{w}_{k+j} &= \left[w_1(k+j) \ \dots \ w_n(k+j)\right]^T \in \mathbb{R}^n \\
\Delta\mathbf{u}_{k+j-1} &= \left[\Delta u_1(k+j-1) \ \dots \ \Delta u_m(k+j-1)\right]^T \in \mathbb{R}^m.
\end{aligned}
\tag{8}
$$

$\mathbf{Q}_\delta$ and $\mathbf{Q}_\lambda$ are appropriate symmetric positive-(semi)definite weighting matrices for sequences of predicted control errors and future control input increments, respectively, given by

$$
\begin{aligned}
\mathbf{Q}_\delta &= \mathbf{diag}\{\mathbf{Q}_\delta(N_1), \dots, \mathbf{Q}_\delta(N_2)\} \in \mathbb{S}_+^{n(N_2-N_1+1)} \\
\mathbf{Q}_\lambda &= \mathbf{diag}\{\mathbf{Q}_\lambda(1), \dots, \mathbf{Q}_\lambda(N_u)\} \in \mathbb{S}_{++}^{mN_u}
\end{aligned}
\tag{9}
$$

where $\mathbf{Q}_\delta(j) = \mathbf{diag}\{\delta_1(j), \dots, \delta_n(j)\}$ and $\mathbf{Q}_\lambda(j) = \mathbf{diag}\{\lambda_1(j), \dots, \lambda_m(j)\}$.

Since the MIMO model (1) leads to solving matrix Diophantine equations, it is not convenient concerning computational difficulties with deriving j-step-ahead predictor. In the next subsection a technique is shown, which rejects this problem and hence simplifies derivation of prediction equations.

2.2. MIMO model decomposition

In this work, from this point forward, square $m \times m$ systems are considered and a diagonal structure of polynomial matrices $\mathcal{A}$ and $\mathcal{C}$ is assumed, i.e. $\mathcal{A} = \mathbf{diag}\{A_{ii}\}$, $\mathcal{C} = \mathbf{diag}\{C_{ii}\}$, $\forall i \in [1, m]$. As a result of this assumption, model (2) can be considered as a set of m MISO processes (see Fig. 2) of form:

$$
A_{ii} y_i(k) = \mathbf{B}_i \mathbf{u}_{k-1} + \frac{C_{ii}}{\Delta} \xi_i(k), \quad i = 1, \dots, m
\tag{10}
$$

where $\mathbf{B}_i$ is the i-th row of $\mathcal{B}$, i.e. $\mathbf{B}_i = [B_{i1} \dots B_{im}]$.

[3]The notations of arrows pointing right/left is used for a vector of future/past values, possibly including current value according to the context. Notice also that throughout the paper bold lower case is used to denote vectors and non-bold lower case implies a scalar quantity.

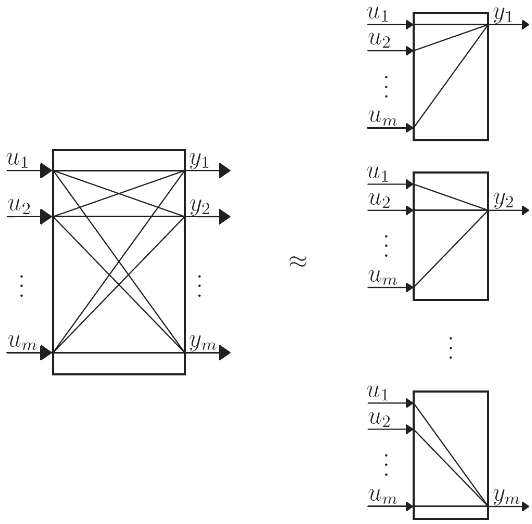

Figure 2. Basic idea of a $m \times m$ MIMO model decomposition. Clearly, the same strategy can be applied to a $m \times n$ process.

Although the proposed model is not of general form, it is suitable for adequate representation of most real processes [4]. Deterministic parts of both MIMO and MISO descriptions[4] are equivalent, therefore the transformation depends on noise properties [8]. Main benefits of the above introduced form lie in the following:

- output predictions based on MISO models are possible to perform in a way similar to the SISO case [1],
- number of estimated parameters of MISO model (10) is smaller than in MIMO case (2) with full $\mathcal{A}$ and $\mathcal{C}$, which is most desirable in adaptive applications when on-line estimators are used [4],
- proposed structure considerably simplifies MIMO controller implementation.

From the previous it follows that for a 2×2 process one gets:

$$\underbrace{\begin{bmatrix} A_{11} & 0 \\ 0 & A_{22} \end{bmatrix}}_{\mathcal{A}} \underbrace{\begin{bmatrix} y_1(k) \\ y_2(k) \end{bmatrix}}_{\mathbf{y}_k} = \underbrace{\begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}}_{\mathcal{B}} \underbrace{\begin{bmatrix} u_1(k-1) \\ u_2(k-1) \end{bmatrix}}_{\mathbf{u}_{k-1}} + \frac{1}{\Delta} \underbrace{\begin{bmatrix} C_{11} & 0 \\ 0 & C_{22} \end{bmatrix}}_{\mathcal{C}} \underbrace{\begin{bmatrix} \xi_1(k) \\ \xi_2(k) \end{bmatrix}}_{\boldsymbol{\xi}_k} \tag{11}$$

where

$$\mathcal{A} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} (a_1)_{11} & 0 \\ 0 & (a_1)_{22} \end{bmatrix} q^{-1} + \cdots + \begin{bmatrix} (a_{n_A})_{11} & 0 \\ 0 & (a_{n_A})_{22} \end{bmatrix} q^{-n_A},$$

[4]Terms such as "full" multivariable and "distributed" GPC can be also found in the literature, see e.g. [11].

$$\mathcal{B} = \begin{bmatrix} (b_0)_{11} & (b_0)_{12} \\ (b_0)_{21} & (b_0)_{22} \end{bmatrix} + \begin{bmatrix} (b_1)_{11} & (b_1)_{12} \\ (b_1)_{21} & (b_1)_{22} \end{bmatrix} q^{-1} + \dots \begin{bmatrix} (b_{n_B})_{11} & (b_{n_B})_{12} \\ (b_{n_B})_{21} & (b_{n_B})_{22} \end{bmatrix} q^{-n_B}.$$

In general, the monic polynomial matrix $\mathcal{C}$ is chosen equal to the identity matrix because of the difficulties in identifying the characteristics of noise [14]. Its identification is therefore assumed not necessary here.

Notice also that $n_A = n_{A_{ii}}$, $n_B = n_{B_{il}}$ is assumed.

2.3. On-line identification algorithm

The next stage in the application of GPC adaptive control strategy is the on-line identification of process model parameters (coefficients of A_{ii} and B_{il} polynomials). This is the so-called “self-tuning” phase that distinguishes adaptive control from controllers based on off-line analysis and design. Here, adaptation is achieved by using two MISO *recursive least squares* (RLS) estimators modified for a practical application by introducing *forgetting* of the non-informative data. Methods based on this principle can be viewed as a special case of the following general forgetting algorithm which can be divided into two independent parts [10]:

- *data-update* iteration

$$\widehat{\boldsymbol{\theta}}_{k|k} = \widehat{\boldsymbol{\theta}}_{k|k-1} + \frac{\mathbf{P}_{k|k-1}\boldsymbol{\phi}_k}{\sigma^2 + \boldsymbol{\phi}_k^T \mathbf{P}_{k|k-1}\boldsymbol{\phi}_k} \widehat{e}_{k|k-1} \tag{12a}$$

$$\mathbf{P}_{k|k}^{-1} = \mathbf{P}_{k|k-1}^{-1} + \frac{\boldsymbol{\phi}_k \boldsymbol{\phi}_k^T}{\sigma^2} \tag{12b}$$

- *time-update* iteration

$$\widehat{\boldsymbol{\theta}}_{k+1|k} = \widehat{\boldsymbol{\theta}}_{k|k} \tag{13a}$$

$$\mathbf{P}_{k+1|k} = F\{\mathbf{P}_{k|k}\}, (\geqslant \mathbf{P}_{k|k}) \tag{13b}$$

where $\widehat{\boldsymbol{\theta}}_k$ is the parameter estimates vector, $\boldsymbol{\phi}_k$ is regression vector, $\mathbf{P}_k$ denotes covariance matrix of the estimates, and

$$\widehat{e}_{k|k-1} = y_k - \widehat{\boldsymbol{\theta}}_{k|k-1}^T \boldsymbol{\phi}_k \tag{14}$$

is the a priori estimation error.

Forgetting itself is introduced via a reasonable choice of forgetting operator $F\{.\}$, i.e. in the time update of the covariance matrix $\mathbf{P}$. Since the plant is time-varying, the use of a forgetting factor prevents the eigenvalues of $\mathbf{P}$ tending to zero and hence avoids a possible numerical destabilization of the algorithm. The RLS estimation scheme utilized in this work assumes so-called “directional” forgetting proposed in [9].

2.4. Design of a MIMO predictor

Output predictions based on CARIMA models are carried out in a way similar to the SISO case, by the use of recursive Diophantine equations or direct iteration[5] of

$$\mathcal{C}^{-1}\mathcal{A}(q^{-1})\Delta\mathbf{y}_k = \mathcal{C}^{-1}\mathcal{B}(q^{-1})\Delta\mathbf{u}_{k-1} \tag{15}$$

under the assumption that the future white noise is zero [4].

As the matrix $\mathcal{A}$ is diagonal the Diophantine equations that may be used to obtain the optimal prediction of each output can also be solved independently for each output, thus the one-output m-input CARIMA model is used [14]. Substituting $(k+j)$ for (k) in (10) and multiplying by $E_{i,j}$ yields:

$$E_{i,j}A_{ii}\Delta y_i(k+j\,|\,k) = E_{i,j}\mathbf{B}_i\Delta\mathbf{u}_{k+j-1} + E_{i,j}\xi_i(k+j) \tag{16}$$

Using the solution of the first Diophantine equation,

$$1 = E_{i,j}A_{ii}\Delta + q^{-j}F_{i,j} \tag{17}$$

in (18) leads to

$$y_i(k+j\,|\,k) = F_{i,j}y_i(k) + \sum_{l=1}^{m} E_{i,j}B_{il}\Delta u_l(k+j-1) + E_{i,j}\xi_i(k+j) \tag{18}$$

where $E_{i,j}(q^{-1})$ and $F_{i,j}(q^{-1})$ are polynomials of order $j-1$ and n_A-1, respectively.

In order to split up the j-step-ahead predictor (18) in parts that are known at time (k) and future signals, the second Diophantine equation,

$$E_{i,j}B_{il} = G_{il,j} + q^{-j}G_{p_{il,j}} \tag{19}$$

is introduced, resulting in

$$\widehat{y}_i(k+j\,|\,k) = \sum_{l=1}^{m} G_{il,j}\Delta u_l(k+j-1) + \sum_{l=1}^{m} G_{p_{il,j}}\Delta u_l(k-1) \; + \; F_{i,j}y_i(k) \tag{20}$$

where the term $\mathbf{G}_{i,j}\Delta\mathbf{u}_{k+j-1}$ represents the *forced* response and the last two terms denoted as $f_i(k+j) = \mathbf{G}_{p_{i,j}}\Delta\mathbf{u}_{k-1} + F_{i,j}y_i(k)$ are the *free* response of the process. Notice also that the last term in (18) involving future noise $\xi_i(k+j)$ is omitted in (20) since it is assumed to be a discrete zero-mean white noise. Thus $\widehat{y}_i(k+j\,|\,k)$ denotes an estimate of $y_i(k+j\,|\,k)$.

For further use, it is preferable to collect all the j-step-ahead predictors (for $j = N_1 \ldots N_2$) for all m MISO processes in the following vector/matrix notation:

$$\underset{\rightarrow}{\widehat{\mathbf{y}}} = \mathbf{G}\Delta\underset{\rightarrow}{\mathbf{u}} + \underset{\rightarrow}{\mathbf{f}} \tag{21}$$

[5] In an adaptive context, direct iterating forward in time results in a significant computational saving compared to the Diophantine equation approach and from the coding point of view can be effectively applied by the use of Toeplitz/Hankel matrix algebra as shown e.g. in [16, 7].

where the free response sequence $\underset{\rightarrow}{\mathbf{f}}$ can be obtained as

$$\underset{\rightarrow}{\mathbf{f}} = \mathbf{G}_p \Delta \mathbf{u}_{k-1} + \mathbf{F}\mathbf{y}_k,$$

and which considers the following quantities:

$$\mathbf{G} = \begin{bmatrix} \mathbf{G}(N_1) & \mathbf{G}(N_1-1) & \cdots & \mathbf{0}_{m\times m} \\ \mathbf{G}(N_1+1) & \mathbf{G}(N_1) & \cdots & \mathbf{0}_{m\times m} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{G}(N_2) & \mathbf{G}(N_2-1) & \ldots & \mathbf{G}(N_2-N_u+1) \end{bmatrix} \in \mathbb{R}^{m(N_2-N_1+1)\times mN_u}$$

$$\mathbf{G}(j) = \begin{bmatrix} g_{11}(j) & \cdots & g_{1m}(j) \\ \vdots & \ddots & \vdots \\ g_{m1}(j) & \cdots & g_{mm}(j) \end{bmatrix} \in \mathbb{R}^{m\times m},$$

$$\underset{\rightarrow}{\mathbf{f}} = \begin{bmatrix} \mathbf{f}^T_{k+N_1} & \ldots & \mathbf{f}^T_{k+N_2} \end{bmatrix}^T \in \mathbb{R}^{m(N_2-N_1+1)}$$

$$\mathbf{f}_{k+j|k} = \begin{bmatrix} f_1(k+j) & \ldots & f_m(k+j) \end{bmatrix}^T \in \mathbb{R}^m,$$

where $g_{il}(j)$ is the j-th step response coefficient of the transfer function relating the i-th output to the l-th input.

Similarly, $\mathbf{G}_p, \mathbf{F} \in \mathbb{R}^{m(N_2-N_1+1)\times m}$ are polynomial matrices consisting of polynomials $G_{p_{il,j}}(q^{-1})$ and $F_{i,j}(q^{-1})$. Orders of $G_{il,j}$ and $G_{p_{il,j}}$ are $j-1$ and n_B-1, respectively.

2.5. MGPC control law

The system output prediction sequence, $\underset{\rightarrow}{\widehat{\mathbf{y}}}$, given by (21) can be used in the cost function (6), which subsequently yields

$$J = \Delta \underset{\rightarrow}{\mathbf{u}}^T \mathbf{H} \Delta \underset{\rightarrow}{\mathbf{u}} + 2\mathbf{g}^T \Delta \underset{\rightarrow}{\mathbf{u}} + \mathbf{f}_0 \tag{22}$$

where

$$\mathbf{H} = \mathbf{G}^T \mathbf{Q}_\delta \mathbf{G} + \mathbf{Q}_\lambda$$
$$\mathbf{g} = \mathbf{G}^T \mathbf{Q}_\delta (\underset{\rightarrow}{\mathbf{f}} - \underset{\rightarrow}{\mathbf{w}})$$
$$\mathbf{f}_0 = (\underset{\rightarrow}{\mathbf{f}} - \underset{\rightarrow}{\mathbf{w}})^T (\underset{\rightarrow}{\mathbf{f}} - \underset{\rightarrow}{\mathbf{w}}).$$

2.5.1. Unconstrained formulation

The necessary condition for finding the unconstrained solution of the MGPC controller,

$$\Delta \underset{\rightarrow}{\mathbf{u}}^* = arg \min_{\Delta \underset{\rightarrow}{\mathbf{u}}} J \tag{23}$$

is obtained by setting the gradient of (22) to zero with respect to $\Delta \underset{\rightarrow}{\mathbf{u}}$, that is

$$\frac{\partial J}{\partial \Delta \underset{\rightarrow}{\mathbf{u}}} = \mathbf{H}\Delta \underset{\rightarrow}{\mathbf{u}} + \mathbf{g} = \mathbf{0} \tag{24}$$

which results in a closed form control law

$$\Delta\underset{\rightarrow}{\mathbf{u}}^* = -\mathbf{H}^{-1}\mathbf{g} = \left[\mathbf{G}^T\mathbf{Q}_\delta\mathbf{G} + \mathbf{Q}_\lambda\right]^{-1}\mathbf{G}^T\mathbf{Q}_\delta(\underset{\rightarrow}{\mathbf{w}} - \underset{\rightarrow}{\mathbf{f}}) \tag{25}$$

Finally, in accordance with the receding horizon strategy, at the current sample time (k) only the first part of this solution, i.e. the first block element[6] of the optimal sequence of future input moves, $\Delta\underset{\rightarrow}{\mathbf{u}}^*$,

$$\Delta\mathbf{u}_k^* = \big[\mathbf{I}_m\ \underbrace{\mathbf{0}_{m\times m}\ \ldots\ \mathbf{0}_{m\times m}}_{(N_u-1)\text{times}}\big]\Delta\underset{\rightarrow}{\mathbf{u}}^* \tag{26}$$

is utilized and effectively applied to the plant. At the next step $(k+1)$ the whole process is repeated, shifting the horizon one step further.

2.5.2. Constrained formulation

When dealing with input constraints, it is no longer possible to express the solution explicitly in a closed form as before. Instead, an optimization procedure has to be performed at each sampling instant (k), repeatedly solving a constrained minimization problem in the form:

$$\min_{\Delta \underset{\rightarrow}{\mathbf{u}} \in \mathcal{U}_{ad}} \{J(\Delta \underset{\rightarrow}{\mathbf{u}})\} = \min_{\Delta \underset{\rightarrow}{\mathbf{u}} \in \mathcal{U}_{ad}} \{\Delta \underset{\rightarrow}{\mathbf{u}}^T\mathbf{H}\Delta \underset{\rightarrow}{\mathbf{u}} + 2\mathbf{g}^T\Delta \underset{\rightarrow}{\mathbf{u}} + \mathbf{f}_0\} \tag{27}$$

where $\mathcal{U}_{ad} \subset \mathbb{R}^m$ is a compact set of admissible control signal increments that imply satisfying the following constraints:

[6]Recall that in our multiple-input case this is given by a vector of two control inputs $\Delta\mathbf{u}_k = \left[\Delta u_1(k)\ \ \Delta u_2(k)\right]^T$ which equals the first m wide block row of $-\mathbf{H}^{-1}\mathbf{g}$.

$$\underbrace{\begin{bmatrix}\underline{\Delta\mathbf{u}}\\ \underline{\Delta\mathbf{u}}\\ \vdots\\ \underline{\Delta\mathbf{u}}\end{bmatrix}}_{\underline{\Delta\mathbf{U}}} \leqslant \underbrace{\begin{bmatrix}\Delta\mathbf{u}_{k|k}\\ \Delta\mathbf{u}_{k+1|k}\\ \vdots\\ \Delta\mathbf{u}_{k+N_u-1|k}\end{bmatrix}}_{\Delta\underset{\rightarrow}{\mathbf{u}}} \leqslant \underbrace{\begin{bmatrix}\overline{\Delta\mathbf{u}}\\ \overline{\Delta\mathbf{u}}\\ \vdots\\ \overline{\Delta\mathbf{u}}\end{bmatrix}}_{\overline{\Delta\mathbf{U}}}$$
$$\underbrace{\begin{bmatrix}\underline{\mathbf{u}}\\ \underline{\mathbf{u}}\\ \vdots\\ \underline{\mathbf{u}}\end{bmatrix}}_{\underline{\mathbf{U}}} \leqslant \underbrace{\begin{bmatrix}\mathbf{u}_{k|k}\\ \mathbf{u}_{k+1|k}\\ \vdots\\ \mathbf{u}_{k+N_u-1|k}\end{bmatrix}}_{\underset{\rightarrow}{\mathbf{u}}} \leqslant \underbrace{\begin{bmatrix}\overline{\mathbf{u}}\\ \overline{\mathbf{u}}\\ \vdots\\ \overline{\mathbf{u}}\end{bmatrix}}_{\overline{\mathbf{U}}} \tag{28}$$
$$\underbrace{\begin{bmatrix}\underline{\mathbf{y}}\\ \underline{\mathbf{y}}\\ \vdots\\ \underline{\mathbf{y}}\end{bmatrix}}_{\underline{\mathbf{Y}}} \leqslant \underbrace{\begin{bmatrix}\widehat{\mathbf{y}}_{k+N_1|k}\\ \widehat{\mathbf{y}}_{k+N_1+1|k}\\ \vdots\\ \widehat{\mathbf{y}}_{k+N_2|k}\end{bmatrix}}_{\underset{\rightarrow}{\widehat{\mathbf{y}}}} \leqslant \underbrace{\begin{bmatrix}\overline{\mathbf{y}}\\ \overline{\mathbf{y}}\\ \vdots\\ \overline{\mathbf{y}}\end{bmatrix}}_{\overline{\mathbf{Y}}}$$

where $\underline{\Delta\mathbf{u}}, \overline{\Delta\mathbf{u}}$, $\underline{\mathbf{u}}, \overline{\mathbf{u}}$ and $\underline{\mathbf{y}}, \overline{\mathbf{y}}$ are the lower/upper rate and amplitude control input, and output bounds, respectively. It is easy to see that

$$\underset{\rightarrow}{\mathbf{u}} = \underbrace{\begin{bmatrix}\mathbf{I}_m & \dots & \dots & \mathbf{0}_{m\times m}\\ \mathbf{I}_m & \mathbf{I}_m & & \vdots\\ \vdots & \vdots & \ddots & \vdots\\ \mathbf{I}_m & \mathbf{I}_m & \dots & \mathbf{I}_m\end{bmatrix}}_{\mathbf{C}_{\mathbf{I}/\Delta}\in\mathbb{R}^{mN_u\times mN_u}} \Delta\underset{\rightarrow}{\mathbf{u}} + \underbrace{\begin{bmatrix}\mathbf{I}_m\\ \mathbf{I}_m\\ \vdots\\ \mathbf{I}_m\end{bmatrix}}_{\mathbf{L}\in\mathbb{R}^{mN_u\times m}} \mathbf{u}_{k-1} \tag{29}$$

where $\mathbf{I}_m$ denotes an identity matrix and $\mathbf{0}_{m\times m}$ a zero square matrix of size equivalent to the number of inputs m.

Numerical solution of (27) yields a quadratic programming problem of the following form:

$$\underset{\Delta\underset{\rightarrow}{\mathbf{u}}}{\text{minimize}} \quad \frac{1}{2}\Delta\underset{\rightarrow}{\mathbf{u}}^T\mathbf{H}\Delta\underset{\rightarrow}{\mathbf{u}} + \mathbf{g}^T\Delta\underset{\rightarrow}{\mathbf{u}} \tag{30a}$$

$$\text{subject to} \quad \mathbf{C}\Delta\underset{\rightarrow}{\mathbf{u}} \preceq \mathbf{d}_k \tag{30b}$$

where

$$\mathbf{C} = \begin{bmatrix} \mathbf{I_I} \\ -\mathbf{I_I} \\ \mathbf{C}_{\mathbf{I}/\Delta} \\ -\mathbf{C}_{\mathbf{I}/\Delta} \\ \mathbf{G} \\ -\mathbf{G} \end{bmatrix}; \quad \mathbf{d}_k = \begin{bmatrix} \overline{\Delta\mathbf{U}} \\ -\underline{\Delta\mathbf{U}} \\ \overline{\mathbf{U}} - \mathbf{L}\mathbf{u}_{k-1} \\ -\underline{\mathbf{U}} + \mathbf{L}\mathbf{u}_{k-1} \\ \overline{\mathbf{Y}} - \mathbf{G}_p\Delta\mathbf{u}_{k-1} - \mathbf{F}\mathbf{y}_k \\ -\underline{\mathbf{Y}} + \mathbf{G}_p\Delta\mathbf{u}_{k-1} + \mathbf{F}\mathbf{y}_k \end{bmatrix}. \tag{31}$$

In (30a), $\mathbf{H} \in \mathbb{S}_{++}^{mN_u}$ is the positive-definite Hessian matrix and $\mathbf{g} \in \mathbb{R}^{mN_u}$ is the gradient vector. $\mathbf{I_I} = \mathbf{diag}(\mathbf{I}_m \dots \mathbf{I}_m) \in \mathbb{S}^{mN_u}$ denotes a block diagonal matrix defined with N_u-times repeated matrix $\mathbf{I}_m$. From the previous it is clear that constraints (30b) together give a set of $2m(2N_u + N)$ linear inequalities, where $N = N_2 - N_1 + 1$. It is also evident that $\mathbf{d}_k$ depends upon past input information and $\mathbf{C}$ is time invariant.

To evaluate this constrained minimization quadratic problem (QP) the qpOASES on-line active set solver developed by Ferreau et al. [6] was utilized. Since the MGPC formulation uses variable model parameters identified on-line at each time step (k), the QP matrix $\mathbf{H}$ and vector $\mathbf{g}$ are varying from one QP to the next as well. Hence, the on-line optimization task can be effectively solved by using the sequential module of qpOASES, as shown e.g. in [7].

The resulting optimizing control sequence $\Delta\underset{\rightarrow}{\mathbf{u}}$ is subjected to the receding horizon control strategy in a similar fashion as in the unconstrained case, i.e. by selecting only the control inputs, $\Delta\mathbf{u}_k^*$, corresponding to current time instant (k). Notice also that in the control task presented further the absolute value of control inputs is needed, and can be determined as

$$\mathbf{u}_{k|k} = \Delta\mathbf{u}_{k|k} + \mathbf{u}_{k-1} \tag{32}$$

The experimental results presented in Section 3 consider the above derived constrained MGPC algorithm.

3. Case study: HVAC system control

This section presents experimental results of the proposed adaptive MGPC strategy, applied on a working real-life model of the HVAC system, in order to demonstrate its performance in a temperature and humidity tracking application by applying appropriate control actions determined by the GPC algorithm.

3.1. Laboratory plant description

The system itself is schematically illustrated in Fig. 3. It consists of a scaled room in which the air can be prepared by heating or air-conditioning. The reference room is placed in another box, and thus creating the so-called interspace which is used for simulating outside temperature changes and other effects such as sunlight or shifting wind.

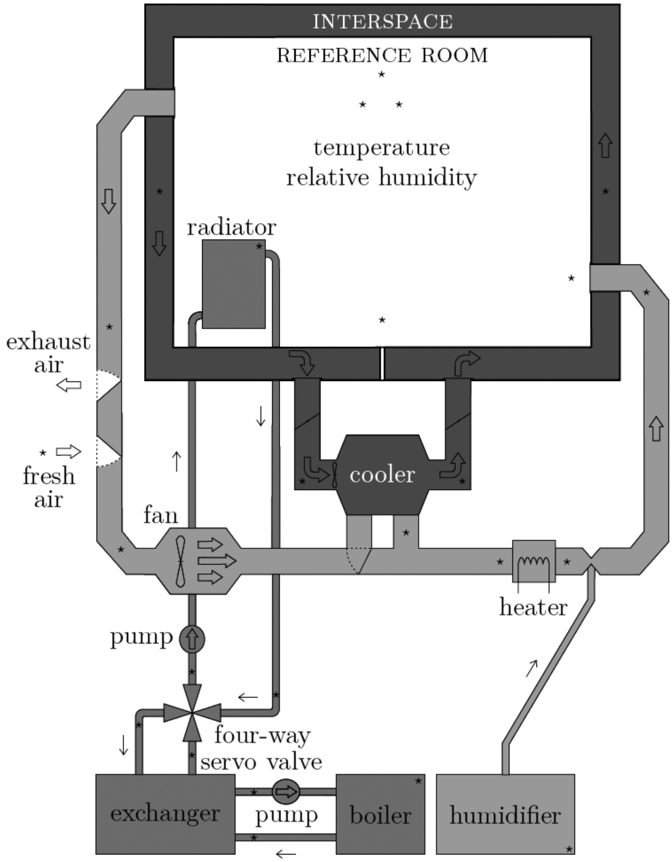

Figure 3. Simplified scheme of the laboratory HVAC system. The *heating* part and *air-conditioning* part of the system are depicted in red and cyan colour, respectively. The star marks ($\star$) refer to the position of sensors.

The red and cyan colour are used to distinguish its heating and air-conditioning part, respectively. The control of these two independent processes represents two particular control tasks being usually practised on the HVAC systems. Please notice that the explanations given further are valid for the latter since it is as a MIMO process in scope of this work. The measurement setup consists of 22 thermal sensors and two sensors of relative humidity situated in the reference room.

The air-conditioning problem was considered as a two-input two-output process, where water temperature in the humidifier, $u_1(k)$, and air flow temperature behind the heating spiral, $u_2(k)$, are the input variables and relative humidity, $y_1(k)$, and temperature, $y_2(k)$, inside the reference room are the output variables. The temperature in reference room can be manipulated with two independent variables, temperature behind the spiral in heater and rate of airflow. However, for experimental purposes only the former with given physical constraints was considered while the latter was kept constant. The

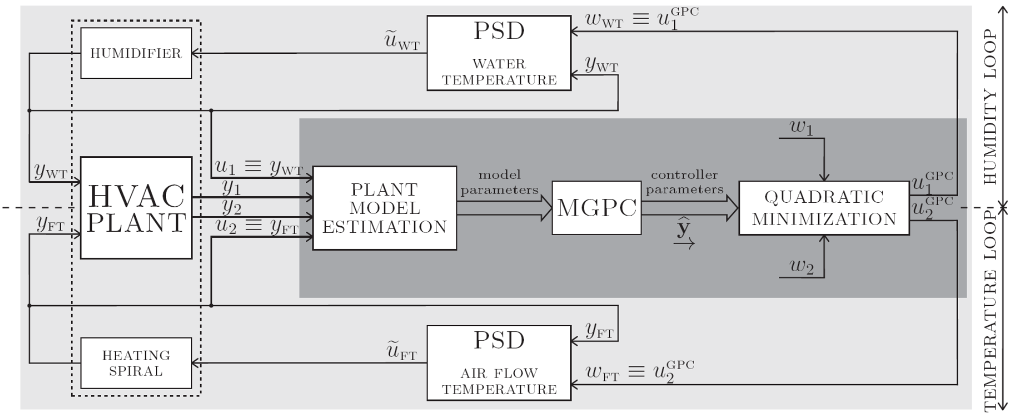

Figure 4. Dual-rate block scheme interpreting the cascade control of the *air-conditioning* part of HVAC laboratory plant. The light and dark gray colour denote the part of control algorithm with higher and lower sampling rate, respectively.

relative humidity in reference room depends on temperature of water in humidifier and also on temperature behind the heating spiral. Since the process of heating and cooling water in humidifier is strongly non-linear, what could introduce many problems in system identification, a solution through injection of cold water into humidifier is proposed.

3.2. Cascade control scheme

The practical implementation of the designed adaptive multivariable GPC algorithm leads to a specific dual-rate control system operating under two frequencies - a so-called cascade control scheme. It consists of a main loop governed by the proposed multivariable predictive controller with sampling time kT, which calculates the desired values for two inner loops with sampling time T. Each of these two is controlled by means of a fixed digital PID (PSD) controller. An incremental (also referred to as a velocity) algorithm is used to calculate the controller output value $\widetilde{u}(k)$ from a previously recorded value $\widetilde{u}(k-1)$ plus correction increment $\Delta\widetilde{u}(k)$ as follows:

$$\widetilde{u}_{\mathrm{WT}}(k) = \Delta\widetilde{u}_{\mathrm{WT}}(k)+\widetilde{u}_{\mathrm{WT}}(k-1) \tag{33a}$$

$$\widetilde{u}_{\mathrm{WT}}(k) = K_P\left\{e_{\mathrm{WT}}(k)-e_{\mathrm{WT}}(k-1)+\frac{T}{T_I}e_{\mathrm{WT}}(k) + \frac{T_D}{T}\left[e_{\mathrm{WT}}(k)-2e_{\mathrm{WT}}(k-1)+e_{\mathrm{WT}}(k-2)\right]\right\}+\widetilde{u}_{\mathrm{WT}}(k-1) \tag{33b}$$

where K_P, T_I and T_D are the proportional gain, the integral time and the derivative time, respectively. The control error is given as

$$e_{\mathrm{WT}}(k) = w_{\mathrm{WT}}(k)-y_{\mathrm{WT}}(k) \tag{34}$$

where $y_{\mathrm{WT}}(k)$ is the measured water temperature in the humidifier and $w_{\mathrm{WT}}(k)$ is its desired value calculated previously by the constrained GPC algorithm, i.e. $w_{\mathrm{WT}} \equiv u_1^{\mathrm{GPC}}$.

Notice that Eq. (33)-(34) are valid for the water temperature (WT) loop. Clearly, a control law for the air flow temperature (FT) loop can be derived in a similar way. The cascade control scheme of the HVAC plant is depicted in Fig. 4.

Control signals $\widetilde{u}_{\mathrm{WT}}$ and $\widetilde{u}_{\mathrm{FT}}$ are subsequently saturated (with adequate amplitude and increment bounds) and applied to the spiral in humidifier and heater, respectively.

3.3. Experimental results

The following experiments represent changes of humidity and temperature set points in the reference room. In general, each experimental run can be divided into three consecutive phases. During the *first*, start-up phase the laboratory plant is achieving steady state around the operation points. The *second* one is a properly chosen learning phase using acquired data, which gives us a preliminary estimation of CARIMA model parameters. For this purpose pseudo-random sequences are applied to both inputs[7], with levels appropriately chosen to excite all relevant modes of the system and such that the process outputs vary within the normal operating region of the process. This procedure makes the start of the controller more reliable and bumpless. The *third* phase finally shows the results of the tuned multivariable adaptive constrained GPC controller in a cascade scheme with PSD controllers.

Number of real-time experiments[8] had been carried out in order to select parameters that would provide the desired controller performance. Finally, the following values of design parameters were set: sampling times $T = 3.5s$ (inner loops), $T_c = 17T$ (main loop), horizons $N_1 = 1$, $N_2 = 15$, $N_u = 3$, dead time $d = 0$, and weights $[\delta_1, \delta_2] = [1, 1]$, $[\lambda_1, \lambda_2] = [1, 10]$. PSD controller parameters were set as $K_P = 50$, $T_I = 120$ and $T_D = 0$. Orders of the model polynomials are $n_A = 5$, $n_B = 5$. Bounds used in minimization of the cost function are $\underline{\mathbf{u}} = [50\ 26]^T$, $\overline{\mathbf{u}} = [95\ 45]^T$, $\underline{\Delta\mathbf{u}} = [-5.5\ -2.5]^T$, $\overline{\Delta\mathbf{u}} = [5.5\ 3]^T$, $\underline{\mathbf{y}} = [50\ 20]^T$ and $\overline{\mathbf{y}} = [65\ 30]^T$.

The experimental results for the air-conditioning part of the HVAC system are depicted in Fig. 5. The start-up *phase I* (first 60 samples) is followed by the learning *phase II* (61 − 560), and finally the *phase III* (561 − 8000), which demonstrates the good performance and robustness of the proposed controller. It also emphasizes the necessity of explicitly taking into account the natural constraints of the actuators. In this case stability is guaranteed via choice of a reasonably long prediction horizon.

The adaptation loop updates on-line an input-output process model, which allows a good behaviour of the control algorithm for different setpoints ($w_1 = 55-60\%$, $w_2 = 24-28°C$).

[7] More precisely, set as desired values ($w_{\mathrm{WT}}, w_{\mathrm{FT}}$) for the PSD controllers.

[8] Duration of one experimental run such as depicted in Fig. 5 was approximately 7.5 hours. The interface between MATLAB/Simulink model and I/O boards was performed under xPC Target rapid prototyping system.

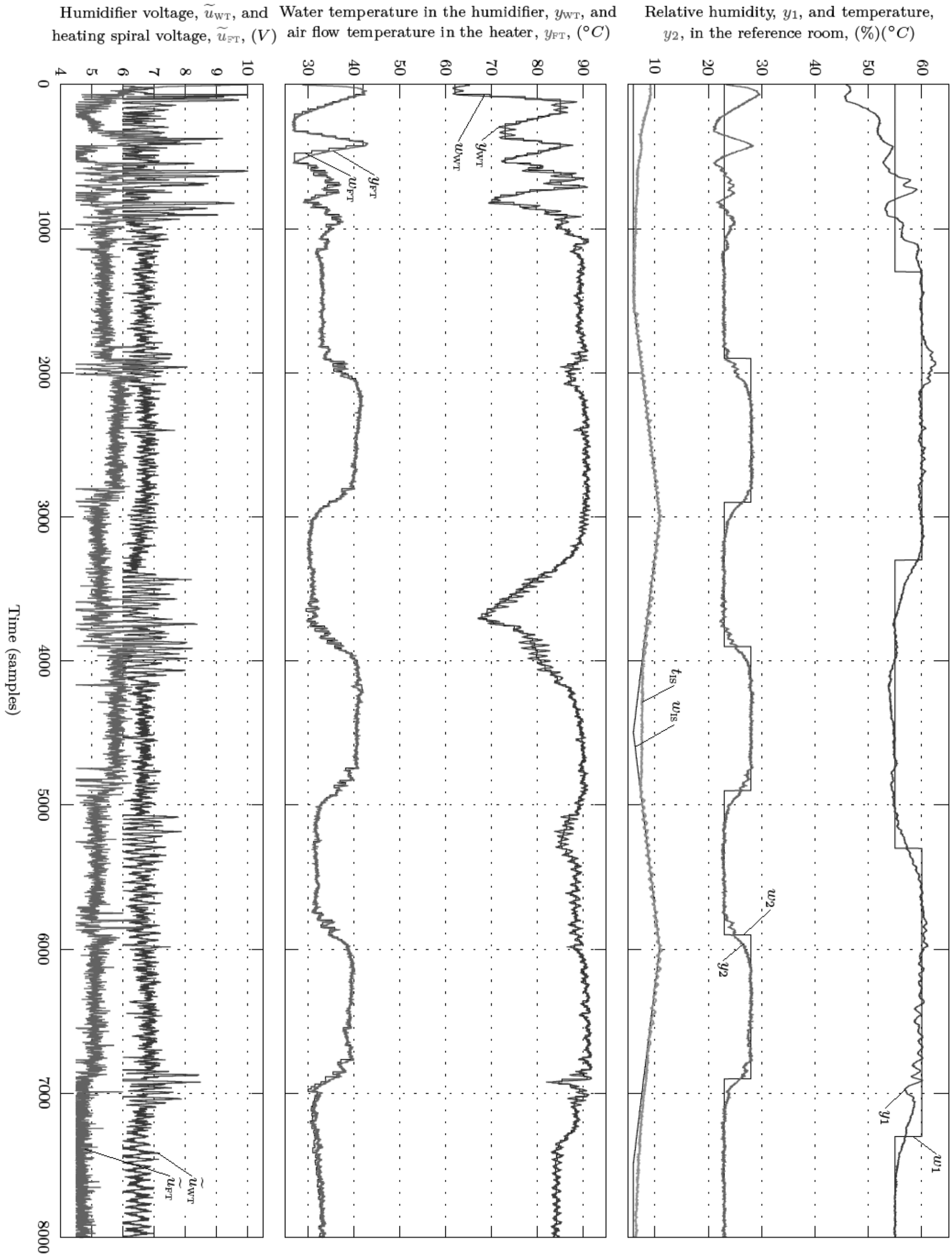

Figure 5. Experimental results for the *air-conditioning* part of the laboratory HVAC system.

Notice also that in the first phase temperature in interspace, t_{IS}, drops under $10°C$, which represents temperature losses of the reference room and thus acts as a disturbance. During the other two phases it changes with respect to its reference by means of on-off control. The outside temperature, t_{OUT}, is shown as well.

4. Conclusion

In this paper, we have developed a multivariable adaptive control algorithm based on generalized predictive control that explicitly takes into account constraints on the actuators.

We have emphasized the practical aspects of the proposed control algorithm through a case study concerning temperature and humidity control of a laboratory HVAC system. This process has some interesting features, at least from the control point of view: multi-input multi-output system with strong couplings, non-linearities, saturations on the actuators, dynamics varying with the setpoint and disturbances.

The proposed control algorithm is shown to perform very satisfactorily, which is due to the adaptation and appropriateness of the algorithm to this particular process.

References

[1] F. BEN AICHA, F. BOUANI and M. KSOURI: A multivariable multiobjective predictive controller. *Int. J. of Applied Mathematics and Computer Science*, **23**(1), (2013), 35-45.

[2] O. BEGO, N. PERIC and I. PETROVIC: Decoupling multivariable GPC with reference observation. In *10th Mediterranean Electrotechnical Conf.*, **2** (2000), 819-822,

[3] Y. CHEN and Z. CHEN: A neural-network-based experimental technique for determining $\mathcal{Z}$-transfer function coefficients of a building envelope. *Building and Environment*, **35**(3), (2000), 181-189.

[4] C.-M. CHOW, T.-W. YOON and D.W. CLARKE: An approach to multivariable predictive control with stability guarantees. In *UKACC Int. Conf. on Control'96*, **2**, (1996), 1350-1355.

[5] E. FEKI, M. AYMEN ZERMANI and A. MAMI: Decoupling control approach for neonate incubator system. *Int. J. of Computer Applications*, **47**(2), (2012), 49-57.

[6] H.J. FERREAU: qpOASES user's manual. http://www.qpOASES.org, 2012.

[7] M. GULAN, M. SALAJ and B. ROHAL'-ILKIV: Real-time implementation of an adaptive feedback and feedforward generalized predictive controller. In *Proc. of the 19th Int. Conf. on Process Control*, Štrbské Pleso, Slovakia, (2013), 383-388.

[8] M. KÁRNÝ, A. HALOUSKOVÁ and J. BÖHM: Mimo - a set of SISO? Multivariate system adaptively controlled as a set of single-input single-output models. In *IFAC Workshop MICC '92.*, Prague, Czech Republic, (1992).

[9] R. KULHAVÝ and M. KÁRNÝ: Tracking of slowly varying parameters by directional forgetting. In *Proc. of the 9th IFAC World Congress*, Budapest, Hungary, **10** (1984), 78-83.

[10] R. KULHAVÝ and M.B. ZARROP: On a general concept of forgetting. *Int. J. of Control*, **58**(4), (1993), 905-924.

[11] E.P. LAMBERT: Process control applications of long-range prediction. Ph.D. dissertation, University of Oxford, 1987.

[12] C. MOHTADI, D.W. CLARKE and P.S. TUFFS: Generalized predictive control - part I. The basic algorithm. *Automatica*, **23**(2), (1987), 137-148.

[13] N. NASSIF, S. KAJL and R. SABOURIN: Evolutionary algorithms for multi-objective optimization in HVAC system control strategy. In *IEEE Annual Meeting of the North American Fuzzy Information Processing Society - NAFIPS, 2004.*, **1** (2004), 51-56.

[14] J.E. NORMEY-RICO and E.F. CAMACHO: Multivariable generalised predictive controller based on the smith predictor. *IEE Proceedings - Control Theory and Applications*, **147**(5), (2000), 538-546.

[15] C. ÖZSOY: Self-tuning control of a heating, ventilating and air-conditioning system. In *Proc. of the Institution of Mechanical Engineers, Part I: J. of Systems and Control Engineering*, **207** (1993), 243-251.

[16] J.A. ROSSITER: Model-Based Predictive Control: A Practical Approach. CRC Press, Boca Raton, 2003.

[17] D. SOLOWAY, J. SHI and A. KELKAR: GPC-based stable reconfigurable control. NASA/TP-2004-212823, (2004).

[18] G. TAKÁCS and B. ROHAL'-ILKIV: Model Predictive Vibration Control: Efficient Constrained MPC Vibration Control for Lightly Damped Mechanical Structures. Springer-Verlag, London, 2012.

[19] Q.-G. WANG, C.-C. HANG, Y. ZHANG and Q. BI: Multivariable controller autotuning with its application in HVAC systems. In *Proc. of the American Control Conference, 1999*, San Diego, California, USA, **6** (1999), 4353-4357.