STABILITY ANALYSIS OF THE RBF-ARX MODEL BASED NONLINEAR PREDICTIVE CONTROL

H. Peng[*], T. Ozaki[**], K. Nakano[†], V. Haggan-Ozaki[‡], Y. Toyoda[⋆]

** College of Information Science & Engineering, Central South University, Changsha, 410083, China. Currently a visiting researcher at the Institute of Statistical Mathematics, 4-6-7 Minami Azabu, Minato-ku, Tokyo 106-8569, Japan. Fax: +81-3-5421-8796, E-mail: peng@ism.ac.jp*

*** The Institute of Statistical Mathematics, 4-6-7 Minami Azabu, Minato-ku, Tokyo 106-8569, Japan.*

† The University of Electro-Communications, 1-5-1 Chofu-ga-oka, Chofu, Tokyo 182-8585, Japan

‡ Sophia University, 4, Yonbancho, Chiyoda-ku, Tokyo 102-0081, Japan

⋆ Niihama National College of Technology, 7-1 Yagumo-cho Niihama, Ehime 792-0805, Japan

Keywords: Nonlinear systems, model predictive control, stability, radial basis function neural networks, ARX model.

Abstract

This paper gives stability analysis of the nonlinear predictive control strategy based on the off-line identified RBF-ARX model which is a pseudo-linear time-varying ARX model with system working-point dependent Gaussian RBF neural network style coefficients. The predictive controller doesn't require on-line parameter estimation; it may be applied to a class of smooth nonlinear processes whose working-point varies over a wide range. Stability analysis of the nonlinear predictive controller is given both in unconstrained case and in case of *a posterior* input constraint. An industrial experiment result of the predictive control design is also revealed for illustrating its effectiveness and feasibility.

1. Introduction

Based on the model-based predictive control (MPC) framework [2], a large concern for MPC is now to be focused on the nonlinear predictive control on the basis of a nonlinear model describing the behavior of system to be controlled. There have been some reports and applications no nonlinear MPC, in which some control schemes (see *e.g.* [6]) were based on the direct use of nonlinear models, but there resulted in on-line solving a higher order nonlinear optimization problem, which is computationally expensive and may get stuck in a local minimum. Some methods (see *e.g.* [1]) used the piecewise linearization technique to describe the nonlinear behavior of a system, so the model was linearized at each sampling interval, which resulted in the solution of a (or a set of) quadratic programming problem at each such interval, as in the case of linear MPC. However, the estimation of many linear models being valid only in each small region is not easy in practice.

In general, the success of MPC is highly dependent on a reliable system model. In fact, a large number of nonlinear processes may be regarded as this kind of system whose working-point varies with time, and which can be locally linearized at any fixed working-point. Therefore, in some cases an alterative suitable for nonlinear MPC is to use a linear time-varying autoregressive model as internal model of MPC, but on-line quickly and accurately estimating its time-varying coefficients is usually difficult. To avoid on-line estimating time-varying parameters of internal model in nonlinear MPC, Peng *et al.* [8] proposed a hybrid pseudo-linear time-varying RBF-ARX model built on the basis of the Gaussian RBF networks and linear ARX model structure to implement MPC for a class of smooth nonlinear systems with working-point dependent dynamics. In the MPC proposed, all the model parameters were estimated off-line by using a quickly-convergent structured nonlinear parameter optimization method (SNPOM) proposed by Peng *et al.* [9]. The RBF-ARX model at any working-point may be treated as a linear ARX model; therefore the quadratic programming routines could be used to solve the optimal control problem at each sampling interval. In this paper, under the assumption that the modeling error of RBF-ARX model is relatively bounded, the stability problems of the RBF-ARX model-based nonlinear predictive controller are discussed without constraint or with *a posterior* input constraint. Finally, the effectiveness and feasibility of the MPC design presented are illustrated by an industrial experiment result to a nonlinear chemical process.

2. The RBF-ARX model based MPC

Consider the nonlinear SISO system with working-point dependent dynamics, which can be described by the following discrete time NARX (nonlinear ARX) model, *i.e.*

$$y(t) = f(y(t-1),\cdots,y(t-n_y),u(t-1),\cdots,u(t-n_u), v(t-1),\cdots,v(t-n_v)) + \zeta(t) \quad (1)$$

where $y(t)$ is the output, $u(t)$ is the input, $v(t)$ is the measurable disturbance, and $\zeta(t)$ is the modeling error generally assumed as a white noise. We use the RBF-ARX model (2) below [8], which is a ARX model with RBF neural network-type coefficients, to approximate system (1) as follows

$$
\begin{cases}
A_t(q^{-1})y(t) = a_{0,t} + B_t(q^{-1})u(t-1) + D_t(q^{-1})v(t-1) + \xi(t) \\
a_{0,t} = c_0^0 + \sum_{k=1}^{m} c_k^0 \exp[-\lambda_k^y \left\| \mathbf{W}(t-1) - \mathbf{Z}_k^y \right\|_2^2] \\
A_t(q^{-1}) = 1 + \sum_{i=1}^{n_y} \{c_{i,0}^y + \sum_{k=1}^{m} c_{i,k}^y \exp[-\lambda_k^y \left\| \mathbf{W}(t-1) - \mathbf{Z}_k^y \right\|_2^2]\} q^{-i} \\
B_t(q^{-1}) = \sum_{i=1}^{n_u} \{c_{i,0}^u + \sum_{k=1}^{m} c_{i,k}^u \exp[-\lambda_k^u \left\| \mathbf{W}(t-1) - \mathbf{Z}_k^u \right\|_2^2]\} q^{-i+1} \\
D_t(q^{-1}) = \sum_{i=1}^{n_v} \{c_{i,0}^v + \sum_{k=1}^{m} c_{i,k}^v \exp[-\lambda_k^v \left\| \mathbf{W}(t-1) - \mathbf{Z}_k^v \right\|_2^2]\} q^{-i+1} \\
\mathbf{W}(t-1) = [w(t-1), w(t-2), \cdots, w(t-n_w)]^{\mathrm{T}} \\
\mathbf{Z}_k^j = \left(z_{k,1}^j, z_{k,2}^j, \cdots, z_{k,n_w}^j \right)^{\mathrm{T}},\ j = y, u, v
\end{cases} \tag{2}
$$

where $\mathbf{Z}_k^j$ are the centers of RBF networks; λ_k are the scaling parameters; $c_{i,k}^j$, c_k^0 are the scalar weighting coefficients; $\|\bullet\|_2$ denotes the vector 2-norm, and q^{-1} is the unit delay operator. The state variable $\mathbf{W}(t-1)$ in model (2), which makes the model coefficients vary with working-point, may be the output signal, the input signal, an external signal, or a composition of the above two or more signals. In fact, a signal which governs system's working-point is more suitable to be used as $\mathbf{W}(t-1)$ in model (2).

Model (2) has an autoregressive structure similar to a linear ARX model, and its state-dependent coefficients make the model dynamics change with system working-point. It is clear that the local linearization of model (2) is a linear ARX model at any working-point by fixing $\mathbf{W}(t-1)$ in (2). In general, model (2) does not require too many RBF centers compared with a single RBF network model, because the complexity of the model is dispersed into the lags of the autoregressive parts of the model. In this paper, the RBF-ARX model (2) is used as the internal model of the predictive controller given in this section. In order to avoid some potential problems caused by parameter estimation online, such as parameter divergence failure, all the parameters of model (2) are identified off-line by the structured nonlinear parameter optimization method (SNPOM) [9], which could largely accelerate the computational convergence of parameter optimization process, especially for the RBF-ARX model with more linear weights and less nonlinear parameters.

Based on model (2) at time t, the $j(j=1,2,\cdots,N)$ step ahead optimal predictive output is given in [8] by

$$
\hat{y}(t+j\,|\,t) = G_{t,j}^{'}(q^{-1})u(t+j-1) + y_0(t+j\,|\,t) \tag{3}
$$

$$
\begin{aligned}
y_0(t+j\,|\,t) = {} & E_{t,j}^{'}(1)a_{0,t} + F_{t,j}(q^{-1})y(t) + H_{t,j}(q^{-1})u(t-1) \\
& + E_{t,j}^{'}(q^{-1})D_t(q^{-1})v(t+j-1)
\end{aligned} \tag{4}
$$

where the polynomials $E_{t,j}^{'}(q^{-1})$, $F_{t,j}(q^{-1})$, $G_{t,j}^{'}(q^{-1})$, and $H_{t,j}(q^{-1})$ are the solutions of two Diophantine equations below

$$
1 = E_{t,j}^{'}(q^{-1})A_t(q^{-1}) + q^{-j}F_{t,j}(q^{-1}) \tag{5}
$$

$$
E_{t,j}^{'}(q^{-1})B_t(q^{-1}) = G_{t,j}^{'}(q^{-1}) + q^{-j}H_{t,j}(q^{-1}) \tag{6}
$$

Define

$$
\begin{aligned}
&\hat{\mathbf{Y}}(t) = [\hat{y}(t+1\,|\,t), \hat{y}(t+2\,|\,t), \cdots, \hat{y}(t+N\,|\,t)]^{\mathrm{T}} \\
&\mathbf{Y}_0(t) = [y_0(t+1\,|\,t), y_0(t+2\,|\,t), \cdots, y_0(t+N\,|\,t)]^{\mathrm{T}} \\
&\mathbf{U}(t) = [u(t), u(t+1), \cdots, u(t+N_u-1)]^{\mathrm{T}} \\
&\mathbf{Y}_r(t) = [y_r(t+1), y_r(t+2), \cdots, y_r(t+N)]^{\mathrm{T}}
\end{aligned}
$$

where N is the prediction horizon, and N_u is the control horizon after which control is assumed to have no change, *i.e.* $u(t+j) = u(t+N_u-1)$ $(j \geq N_u)$, and $\mathbf{Y}_r(t)$ is a given desired-output-sequence. From formula (3), yields

$$
\hat{\mathbf{Y}}(t) = \mathbf{G}_t\mathbf{U}(t) + \mathbf{Y}_0(t) \tag{7}
$$

$$
\mathbf{G}_t = \begin{bmatrix}
g_{t,0} & & & \mathbf{0} \\
g_{t,1} & g_{t,0} & & \\
\vdots & \vdots & \ddots & \\
g_{t,N_u-1} & g_{t,N_u-2} & \cdots & g_{t,0} \\
\vdots & \vdots & \vdots & \vdots \\
g_{t,N-1} & g_{t,N-2} & \cdots & \sum_{i=0}^{N-N_u} g_{t,i}
\end{bmatrix}_{N \times N_u} \tag{8}
$$

$$
G_{t,j}^{'}(q^{-1}) = g_{t,0} + g_{t,1}q^{-1} + \cdots + g_{t,j-1}q^{-j+1}
$$

Consider the following optimization problem without constraint:

$$
\min_{\mathbf{U}(t)} J = \left\| \hat{\mathbf{Y}}(t) - \mathbf{Y}_r(t) \right\|_{\mathbf{I}_{(n\times N)\times(n\times N)}}^2 + \left\| \mathbf{U}(t) \right\|_{\mathbf{R}}^2
$$

where $\mathbf{R} = diag\{r_1, \cdots, r_{N_u}\}$ is the tunable control weighting matrix. The optimal solution of the above optimization problem without constraint is given by

$$
\mathbf{U}(t) = (\mathbf{G}_t^{\mathrm{T}}\mathbf{G}_t + \mathbf{R})^{-1}\mathbf{G}_t^{\mathrm{T}}[\mathbf{Y}_r(t) - \mathbf{Y}_0(t)] \tag{9}
$$

In the solved optimal controls from (9), just first component $u(t)$ in $\mathbf{U}(t)$ is used as control input. In practice, if just considering only input constraint, to avoid using QP routines, according to the posterior input constraint rule below, the predictive control $u(t)$ can be derived as follows

$$
\begin{cases}
\tilde{\mathbf{U}}(t) = (\mathbf{G}_t^{\mathrm{T}}\mathbf{G}_t + \mathbf{R})^{-1}\mathbf{G}_t^{\mathrm{T}}[\mathbf{Y}_r(t) - \mathbf{Y}_0(t)]; \\
\tilde{u}(t) = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}\tilde{\mathbf{U}}(t); \\
\Delta\tilde{u}(t) = \tilde{u}(t) - u(t-1); \\
\text{if } u_{\min} \leq \tilde{u}(t) \leq u_{\max}, \text{ and } \Delta u_{\min} \leq \Delta\tilde{u}(t) \leq \Delta u_{\max}, \\
\qquad u(t) = \tilde{u}(t); \\
\text{else } \ \hat{u}(t) = u(t-1) + \Delta u_{\max}, \quad \text{if } \Delta\tilde{u}(t) > \Delta u_{\max}; \\
\qquad \hat{u}(t) = u(t-1) + \Delta u_{\min}, \quad \text{if } \Delta\tilde{u}(t) < \Delta u_{\min}; \\
\qquad u(t) = u_{\max}, \quad \text{if } \hat{u}(t) \geq u_{\max}; \\
\qquad u(t) = u_{\min}, \quad \text{if } \hat{u}(t) \leq u_{\min}; \\
\qquad u(t) = \hat{u}(t), \quad \text{if } u_{\min} < \hat{u}(t) < u_{\max}.
\end{cases} \tag{10}
$$

Note that the above RBF-ARX model-based predictive controller does not require the parameter estimation online, because its internal model is an off-line estimated global model.

3. Stability analysis

In this paper, we give the stability analysis of the off-line estimated RBF-ARX model-based nonlinear predictive control strategy presented in Section 2. To this end, assume that the plant under control is now described by

$$A_t(q^{-1})y(t) = a_0(t-1) + B_t(q^{-1})u(t-1) + D_t(q^{-1})v(t-1) + e(t) \tag{11}$$

which is just rewritten by model (2), and $e(t)$ is the modeling error including the unmodeled dynamics. Besides, assume that the reference signal $y_r(t)$ and the measurable disturbance signal $v(t)$ are all bounded. For $e(t)$ in model (11), we have now to introduce the following (see [5]).

Assumption 3.1: *The modeling error $e(t)$ in (11) is relatively bounded, i.e. for any t, there exist nonnegative constants $\phi(0)$, ε_u, ε_y, and $0 \le \sigma_e < 1$, such that*

$$|e(t)| \le \phi(t) \tag{12}$$

where $\phi(t)$ is the state of the following dynamical system:

$$\phi(t+1) = \sigma_e \phi(t) + \varepsilon_u |u(t)| + \varepsilon_y |y(t)| \tag{13}$$

Note that a relatively bounded modeling error $e(t)$ is not guaranteed to be bounded unless the plant input and output sequences are bounded. If now defining the state vector $\boldsymbol{\alpha}(t)$ below, then model (11) can be written as

$$\begin{aligned}\boldsymbol{\alpha}(t) = {} & \hat{\mathbf{A}}(t-1)\boldsymbol{\alpha}(t-1) + \hat{\mathbf{B}}(t-1)u(t-1) \\ & + \hat{\mathbf{D}}(t-1)\mathbf{V}(t-1) + \mathbf{L}a_0(t-1) + \mathbf{L}e(t)\end{aligned} \tag{14}$$

where

$$\boldsymbol{\alpha}(t) = [y(t), \cdots, y(t-n_y+1), u(t-1), \cdots, u(t-n_u+1)]^{\mathrm{T}} \tag{15}$$

$$\mathbf{V}(t-1) = [v(t-1), v(t-2), \cdots, v(t-n_v)]^{\mathrm{T}}$$

and the matrixes $\hat{\mathbf{A}}(t-1)$, $\hat{\mathbf{B}}(t-1)$, $\hat{\mathbf{D}}(t-1)$, and $\mathbf{L}$ are shown in Appendix. Consider now the following optimization problem without constraints to system (11) with $e(t) = 0$:

$$\begin{aligned}\min_{\mathbf{U}(t)} J &= \left\| \hat{\mathbf{Y}}(t) - \mathbf{Y}_r(t) \right\|^2_{\mathbf{I}_{(n\times N)\times(n\times N)}} + \left\| \mathbf{U}(t) \right\|^2_{\mathbf{R}} \\ &= \left\| \mathbf{G}_t \mathbf{U}(t) + \mathbf{Y}_0(t) - \mathbf{Y}_r(t) \right\|^2 + \mathbf{U}^{\mathrm{T}}(t)\mathbf{R}\mathbf{U}(t)\end{aligned} \tag{16}$$

The optimal control is given by

$$\mathbf{U}(t) = (\mathbf{G}_t^{\mathrm{T}}\mathbf{G}_t + \mathbf{R})^{-1}\mathbf{G}_t^{\mathrm{T}}[\mathbf{Y}_r(t) - \mathbf{Y}_0(t)] \tag{17}$$

in which the first element $u(t)$ is applied as the predictive control law. Since there are the linear relations between $\mathbf{U}(t)$ and $\mathbf{Y}_r(t)$ or $\mathbf{Y}_0(t)$, and between $\mathbf{Y}_0(t)$ and $\boldsymbol{\alpha}(t)$ or $\mathbf{V}(t)$, the predictive control law $u(t)$ can be written as

$$u(t) = \mathbf{K}(t)\boldsymbol{\alpha}(t) + \mathbf{Q}(t)\mathbf{Y}_r(t) + \mathbf{P}(t)\mathbf{V}(t) + s(t)a_0(t-1) \tag{18}$$

where $\mathbf{K}(t) = [k_0(t), k_1^y(t), \cdots, k_{n_y}^y(t), k_1^u(t), \cdots, k_{n_u-1}^u(t)]$

$\mathbf{Q}(t) = [h_1(t), h_2(t), \cdots, h_N(t)]$

$\mathbf{P}(t) = [p_1(t), p_2(t), \cdots p_{n_v}(t)]$

here $\mathbf{K}(t)$, $\mathbf{Q}(t)$, $\mathbf{P}(t)$ and $s(t)$ in (18) are all time-varying suitable row vectors or scalar depending on the parameters of model (11), which are easily obtained from (17) and (4-6). Therefore, the closed-loop control system (14) and (18) can be written as

$$\begin{aligned}\boldsymbol{\alpha}(t) = {} & \bar{\mathbf{A}}(t-1)\boldsymbol{\alpha}(t-1) + \bar{\mathbf{B}}(t-1)\mathbf{Y}_r(t-1) \\ & + \bar{\mathbf{D}}(t-1)\mathbf{V}(t-1) + \bar{\mathbf{L}}(t-1)a_0(t-1) + \mathbf{L}e(t)\end{aligned} \tag{19}$$

$$\begin{cases} \bar{\mathbf{A}}(t-1) = \hat{\mathbf{A}}(t-1) + \hat{\mathbf{B}}(t-1)\mathbf{K}(t-1) \\ \bar{\mathbf{B}}(t-1) = \hat{\mathbf{B}}(t-1)\mathbf{Q}(t-1) \\ \bar{\mathbf{D}}(t-1) = \hat{\mathbf{D}}(t-1) + \hat{\mathbf{B}}(t-1)\mathbf{P}(t-1) \\ \bar{\mathbf{L}}(t-1) = \mathbf{L} + \hat{\mathbf{B}}(t-1)s(t-1) \end{cases} \tag{20}$$

3.1 Unforced system stability

Before considering the stability of the closed-loop system (19), let us first give the stability conclusion of the unforced system below

$$\boldsymbol{\alpha}(t) = \hat{\mathbf{A}}(t-1)\boldsymbol{\alpha}(t-1) + \hat{\mathbf{B}}(t-1)u(t-1) \tag{21}$$

$$u(t) = \mathbf{K}(t)\boldsymbol{\alpha}(t) \tag{22}$$

or

$$\begin{cases} \boldsymbol{\alpha}(t) = \bar{\mathbf{A}}(t-1)\boldsymbol{\alpha}(t-1) \\ \bar{\mathbf{A}}(t-1) = \hat{\mathbf{A}}(t-1) + \hat{\mathbf{B}}(t-1)\mathbf{K}(t-1) \end{cases} \tag{23}$$

To derive the results given in Theorem 3.1, we have now to introduce the following.

Lemma 3.1: *For RBF-ARX model (11) or (2) with the RBF network-style parameters, the parameters are bounded, and the parameter variations are slow, i.e. there exist positive constants c_1 and smaller c_2, such that*

$$\left\| \boldsymbol{\theta}(t) \right\| \le c_1 \ , \ \forall t \tag{24}$$

$$\left\| \boldsymbol{\theta}(t) - \boldsymbol{\theta}(t-1) \right\| \le c_2 \ , \ \forall t \tag{25}$$

$$\boldsymbol{\theta}(t) = [a_0(t), a_1(t), \cdots, a_{n_y}(t), b_1(t), \cdots, b_{n_u}(t), d_1(t), \cdots, d_{n_v}(t)]^{\mathrm{T}}$$

***Proof*:** For the off-line estimated RBF-ARX model (2 or 11), assume the scaling factors in (2) are positive, *i.e.*

$$\lambda_k^j > 0, \quad (k = 1, \cdots, m \ ; \quad j = y, u, v)$$

This is able to be guaranteed, for which the one way is to apply the heuristic approach to compute λ_k^j, such that

$$\begin{cases} \lambda_k^j = -\log \varepsilon_k \, / \max_{\forall t} \{ \left\| \mathbf{W}(t-1) - \mathbf{Z}_k^j \right\|_2^2 \} \\ \varepsilon_k \in [0.1 \sim 0.0001] \end{cases} \tag{26}$$

after updating the RBF center $\mathbf{Z}_k^j$ at each search iteration in the model parameter optimization process (Peng *et al.*, 2003). Thus, in (2), one can see that

$$0 < \exp[-\lambda_k^j \left\| \mathbf{W}(t-1) - \mathbf{Z}_k^j \right\|_2^2] \le 1 \ , \ \forall \mathbf{W} \ , \ \forall \mathbf{Z}_k^j$$

and all the estimated linear weights are finite constants. It implies the boundedness of $\boldsymbol{\theta}(t)$. Furthermore, it is easy to confirm that for an exponential function below

$$y = \exp(-\lambda_k^j x^2) \ , \ \lambda_k^j > 0 \ , x \in (-\infty \ , +\infty)$$

its maximal differential is given by

$$\max_x |\dot{y}| = \sqrt{2\lambda_k^j} e^{-\frac{1}{2}}$$

which is a constant, furthermore, from the computation formula of λ_k^j in (26), yields that λ_k^j is typically smaller. This implies that the parameter variations of model (11) are slow. □

Lemma 3.2: *If properly choosing* N, N_u *and* $\mathbf{R}$ *in (16), make all eigenvalues of* $\bar{\mathbf{A}}(t-1)$ $(\forall t)$ *in (23) lie in the unit circle, i.e.*

$$\max_{j,t}\{\left|\lambda_j[\bar{\mathbf{A}}(t-1)]\right|\} \le 1-2\rho < 1,\ \rho > 0$$

there then exist constants $c_3 > 0$, $c_4 > 0$ *and* $0 < \eta < 1$ *such that, the unforced system (21) and (22), or (23) is exponentially stable.*

***Proof*:** From Lemma 3.1, one can see that there exists an $c_3 > 0$ for the unforced system (23) such that

$$\left\|\bar{\mathbf{A}}(t-1)\right\| \le c_3\ ,\ \forall t \tag{27}$$

Furthermore, since

$$\bar{\mathbf{A}}(t)-\bar{\mathbf{A}}(t-1) = [\hat{\mathbf{A}}(t)-\hat{\mathbf{A}}(t-1)]+[\hat{\mathbf{B}}(t)-\hat{\mathbf{B}}(t-1)]\mathbf{K}(t) + \hat{\mathbf{B}}(t-1)[\mathbf{K}(t)-\mathbf{K}(t-1)]$$

which leads that there exists an small $c_4 > 0$ such that

$$\left\|\bar{\mathbf{A}}(t)-\bar{\mathbf{A}}(t-1)\right\| \le c_4\ ,\ \forall t \tag{28}$$

For a proper choice of N, N_u and $\mathbf{R}$ in (16), at any fixed time the predictive control law (22) stabilizes system (21), this implies that all eigenvalues of $\bar{\mathbf{A}}(t-1)$ $(\forall t)$ are strictly inside the unit circle. Therefore, one can conclude that the unforced system (23) is exponentially stable (see [4]), where the explicit parameter relations can be given as follows [4]

$$\begin{cases} c_4 = \dfrac{(1-(1-\rho)^2)^2}{2\vartheta^4 c_3}(1-\eta) \\ \vartheta = (1-\rho)\dfrac{(1-\rho)+c_3^{n_\alpha-1}}{\rho^{n_\alpha}} \\ 0<\eta<1\ ,\ n_\alpha = \dim\{\boldsymbol{\alpha}(t)\} \end{cases}$$

□

3.2 Stability conclusions

The stability properties of the RBF-ARX model-based nonlinear predictive control system without constraints can now be stated in the next Theorem 3.1.

Theorem 3.1: *There exists a constant* $\bar{\varepsilon}_e > 0$ *such that, for the system satisfying (12) with known* $0 \le \sigma_e < 1$, $\varepsilon_u < \bar{\varepsilon}_e$ *and* $\varepsilon_y < \bar{\varepsilon}_e$, *and satisfying the condition given in Lemma 3.2, the closed-loop control system (11) and (18) is such that the signals* $y(t)$ *and* $u(t)$ *are uniformly bounded for any initial state, any bounded disturbance signal* $v(t)$ *and any bounded reference signal* $y_r(t)$.

***Proof*:** For a proper choice of N, N_u and $\mathbf{R}$ in (17), yields that the unforced system (23) is exponentially stable as is stated in Lemma 3.2, then, in view of the boundedness of the reference signal, the measurable disturbance signal and the local mean value, from closed-loop system (19), we obtain

$$\left\|\boldsymbol{\alpha}(t)\right\| \le \beta_0\left\|\boldsymbol{\alpha}(0)\right\| + \beta_1 + \sum_{i=1}^{t}\beta_2\sigma_\alpha^{t-i}\left|e(i)\right| \tag{29}$$

for $0 \le \sigma_\alpha < 1$ and suitable finite nonnegative constants β_0, β_1 and β_2. By recalling (13), it turns out that, for some $0 \le \beta_3 < \infty$, we have

$$\begin{aligned}\phi(t) &\le \beta_3 + \varepsilon_u\sum_{i=0}^{t-1}\sigma_e^i\left|u(t-i-1)\right| + \varepsilon_y\sum_{i=0}^{t-1}\sigma_e^i\left|y(t-i-1)\right| \\ &\le \beta_3 + \frac{\varepsilon_u}{1-\sigma_e}\sup_{0\le\tau\le t}\left|u(\tau)\right| + \frac{\varepsilon_y}{1-\sigma_e}\sup_{0\le\tau\le t}\left|y(\tau)\right|\end{aligned}$$

and from the definition of $\boldsymbol{\alpha}(t)$, yields

$$\phi(t) \le \beta_3 + \frac{2\varepsilon_{uy}}{1-\sigma_e}\sup_{0\le\tau\le t}\left\|\boldsymbol{\alpha}(\tau)\right\| \tag{30}$$

where $\varepsilon_{uy} = \max\{\varepsilon_u, \varepsilon_y\}$. By combining (29), (12) and (30), we obtain

$$\begin{aligned}\left\|\boldsymbol{\alpha}(t)\right\| &\le \beta_4 + \frac{2\varepsilon_{uy}\beta_2}{(1-\sigma_e)}\sum_{i=1}^{t}\sigma_\alpha^{t-i}\sup_{0\le\tau\le t}\left\|\boldsymbol{\alpha}(\tau)\right\| \\ &\le \beta_4 + \frac{2\varepsilon_{uy}\beta_2}{(1-\sigma_e)(1-\sigma_\alpha)}\sup_{0\le\tau\le t}\left\|\boldsymbol{\alpha}(\tau)\right\|\end{aligned} \tag{31}$$

where $0 \le \beta_4 < \infty$. Since the right-hand size of (31) is monotonic nondecreasing, it follows that

$$\sup_{0\le\tau\le t}\left\|\boldsymbol{\alpha}(\tau)\right\| \le \beta_4 + \frac{2\varepsilon_{uy}\beta_2}{(1-\sigma_e)(1-\sigma_\alpha)}\sup_{0\le\tau\le t}\left\|\boldsymbol{\alpha}(\tau)\right\| \tag{32}$$

Now assuming that $\varepsilon_{uy} < \dfrac{(1-\sigma_e)(1-\sigma_\alpha)}{2\beta_2} = \bar{\varepsilon}_e$, it follows

$$\sup_{0\le\tau\le t}\left\|\boldsymbol{\alpha}(\tau)\right\| \le \beta_5$$

for $0 \le \beta_5 < \infty$. This also implies that

$$\left\|\boldsymbol{\alpha}(t)\right\| \le \beta_5 \tag{33}$$

for any t. From the definition of $\boldsymbol{\alpha}(t)$ in (15), we can see that $y(t)$ is bounded. Again, from (18), and in view of the boundedness of $\mathbf{K}(t)$, $\mathbf{Q}(t)\mathbf{Y}_r(t)$ and $\mathbf{P}(t)\mathbf{V}(t)$, yields that $u(t)$ is also bounded. □

Consider now the stability problem of the control system with input constrains. We have first to introduce the following.

Assumption 3.2: *The system (11) to be controlled is open-loop stable at any* t. □

It is easy to be confirmed that all eigenvalues of $\hat{\mathbf{A}}(t)$ in Appendix are strictly inside the unit circle under Assumption 3.2.

The objective function of the RBF-ARX model (11) based predictive control with the *posterior* input constraint is given

by

$$\min_{\mathbf{U}(t)} J = \left\|\hat{\mathbf{Y}}(t) - \mathbf{Y}_r(t)\right\|^2_{\mathbf{I}_{(n\times N)\times(n\times N)}} + \left\|\mathbf{U}(t)\right\|^2_{\mathbf{R}} = \left\|\mathbf{G}_t\mathbf{U}(t) + \mathbf{Y}_0(t) - \mathbf{Y}_r(t)\right\|^2 + \mathbf{U}^{\mathrm{T}}(t)\mathbf{R}\mathbf{U}(t) \tag{34}$$

(*posterior*) s. t.

$$u_{\min} \le u(t) \le u_{\max} \quad , \quad \Delta u_{\min} \le \Delta u(t) \le \Delta u_{\max}$$

First solve (34) without constraint to obtain $\tilde{u}(t)$, then update the solution according to the constraint, and the final predictive control $u(t)$ is given by (10). Now the control system stability in case of the posterior input constraints is stated in the next Theorem 3.2.

Theorem 3.2: *Under the conditions satisfying Assumption 3.1-3.2, and the condition given in Lemma 3.2, there exists a constant* $\tilde{\varepsilon}_e > 0$ *such that, for the systems satisfying (12) with known* $0 \le \tilde{\sigma}_e < 1$, $\tilde{\varepsilon}_u < \tilde{\varepsilon}_e$ *and* $\tilde{\varepsilon}_y < \tilde{\varepsilon}_e$, *the closed-loop control system (11) and (10) is such that the signals* $y(t)$ *and* $u(t)$ *are uniformly bounded for any initial state, any bounded disturbance signal* $v(t)$ *and any bounded reference signal* $y_r(t)$.

***Proof*:** For system (11) with the posterior input constraints, in the case that the input is not constrained, the unforced system is described by (23). Otherwise, in the constrained case, system (14) can be also represented by

$$\boldsymbol{\alpha}(t) = \hat{\mathbf{A}}(t-1)\boldsymbol{\alpha}(t-1) + \hat{\mathbf{B}}(t-1)\bar{u}(t-1) + \hat{\mathbf{D}}(t-1)\mathbf{V}(t-1) + \mathbf{L}a_0(t-1) + \mathbf{L}e(t) \tag{35}$$

here the constrained input $\bar{u}(t-1)$ obtained by (10) can be regarded as a bounded disturbance signal, thus, the unforced system in this case is as follows

$$\boldsymbol{\alpha}(t) = \hat{\mathbf{A}}(t-1)\boldsymbol{\alpha}(t-1) \tag{36}$$

which is exponentially stable, according to Assumption 3.2, the boundedness of $\left\|\hat{\mathbf{A}}(t)\right\|$, the small $\left\|\hat{\mathbf{A}}(t) - \hat{\mathbf{A}}(t-1)\right\|$ showed in Lemma 3.1, as well as the conclusion given in Lemma 3.2 (see [3,4]). Furthermore, for a proper choice of N, N_u and $\mathbf{R}$ in (34), yields that the unforced system (23) in the unconstrained case can be also exponentially stabilized as is stated in Theorem 3.1. Then, in view of the boundedness of the reference signal, the measurable disturbance signal and the local mean value, from closed-loop system (19) in unconstrained case or (35) in the constrained case, we have

$$\left\|\boldsymbol{\alpha}(t)\right\| \le \delta_0 \left\|\boldsymbol{\alpha}(0)\right\| + \delta_1 + \sum_{i=1}^{t} \delta_2 \tilde{\sigma}_\alpha^{t-i} \left|e(i)\right|$$

where for $0 \le \tilde{\sigma}_\alpha < 1$ and suitable finite nonnegative constants δ_0, δ_1 and δ_2. The rest of part in this proof is similar to Theorem 3.1, and is omitted here. □

4. Real-time control example

A real-time control example from an industrial experiment is given in this section. Fig.1 shows the structure diagram of a Nitrogen Oxide (NOx) decomposition (de-NOx) process in thermal power plants. This process is nonlinear non-stationary, which has the dynamics changing with the load demand $w(t)$ of power plants [7]. In fact, the working-point of the process is dependent on the load $w(t)$, and at a different load $w(t)$ the process dynamics may be described by a different linear model, thus this process may be described by a load $w(t)$ dependent RBF-ARX model. In Fig.1, the predictive controller works with the existing gain-scheduling PI controller in parallel. The purpose of the de-NOx process control is to reduce the NOx concentration in exhaust gas in order to protect environments. The PI controller is hard to achieve a good trade-off between control performance and ammonia (NH3) consumption.

In this paper, we use a RBF-ARX model (37) with the coefficients similar to (2) as the internal model of MPC

$$A_t(q^{-1})y(t) = a_{0,t} + B_t(q^{-1})(u(t-1) + v_2(t-1)) + D_t^1(q^{-1})v_1(t-1) + D_t^2(q^{-1})w(t-1) + \xi(t) \tag{37}$$

where $y(t)$ is the NOx concentration in exhaust gas; $u(t)$ is the NH3 flowrate-set computed by the MPC; $v_2(t)$ is the NH3 flowrate-set offered by the PI controller; and $v_1(t)$ is the NOx concentration in the de-NOx device inlet, which is measurable disturbance. Note that the variables of describing working-point state in RBF-ARX model (37) is the load demand sequence $\mathbf{W}(t-1)$.

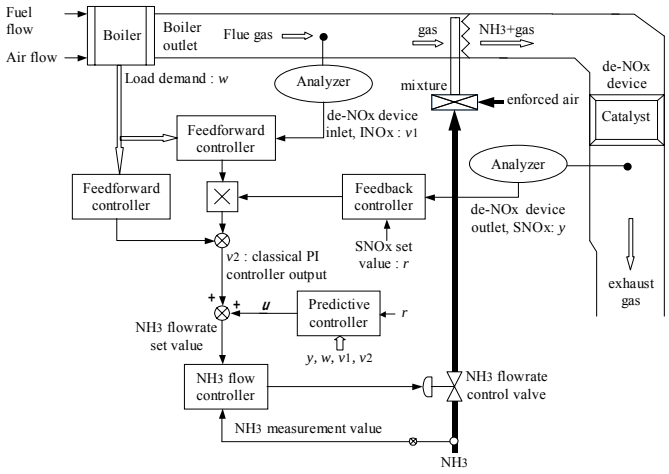

Fig. 1. Structure diagram of Nitrogen Oxide (NOx) decomposition (de-NOx) process and control system.

The estimated result of RBF-ARX model (37) using the structured nonlinear parameter optimization method [9] by the real data from a de-NOx process under the gain-scheduling PI controller alone is shown in Fig. 2. Fig. 3 reveals a set of real-time control results both of the PI controller and of the estimated RBF-ARX model-based predictive controller presented in Section 2. From Figs.2-3, one can see that the estimated RBF-ARX model shows better modeling precision, and the RBF-ARX model-based MPC proposed gives much better real-time control performance compared to the PI control alone.

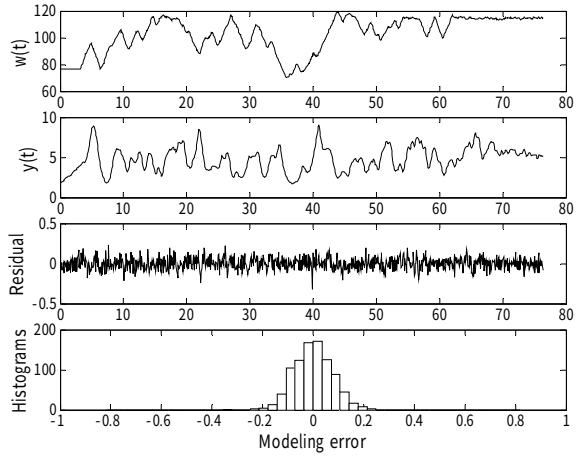

Fig. 2. The estimated residual and histogram of RBF-ARX model (37) for the sample data.

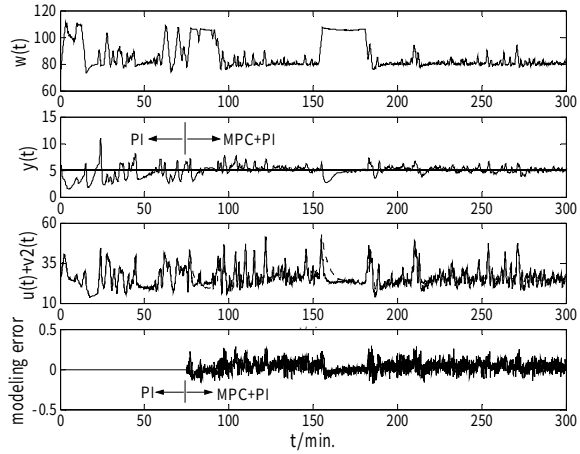

Fig. 3. Real-time control results; the solid line is $v_2(t)$, and the dotted line is $u(t)+v_2(t)$ in the third subplot above.

5. Conclusions

For a class of smooth SISO nonlinear systems which may be characterized by a linear model at certain fixed working-point, an off-line estimated RBF-ARX model might be applied as a global model to represent the system over a larger operation range. The RBF-ARX model based MPC presented did not resort to the parameter estimation on-line, and the adaptation for the time-varying working-point was offered by the working-point dependent RBF neural network-style coefficients of its internal RBF-ARX model. The stability analyses of the proposed predictive control strategy without constraint or just with the posterior input constraint have been given. A real-time control example showed the effectiveness and feasibility of the nonlinear MPC design presented. The future work would be to study the stability analysis with both input and output constraints.

References

[1] Bloemen, H. H. J., Van Den Boom, T. J. J., and Verbruggen, H. B., 2001, Model-based predictive control for Hammerstein-Wiener systems. *International Journal of Control.* **74**, 482-485.

[2] Clarke, D. W., Mohtadi, C., and Tuffs, P. S., 1987, Generalized predictive control. Parts 1 and 2. *Automatica*, **23**, 137-160.

[3] Datta, A., 1998, *Adaptive Internal Model Control* (London: Springer).

[4] Desoer, C. A., 1970, Slowly varying discrete system $x_{i+1} = A_i x_i$. *Electronics Letters,* **6**(11), 339-340.

[5] Kreisselmeier, G., and B. D. O. Anderson, 1986, Robust model reference adaptive control. *IEEE Trans. on Automatic Control,* **31**(2), 127-133.

[6] Mahfouf, M., and Linkens, D. A., 1998, Non-linear generalized predictive control (NLGPC) applied to muscle relaxant anaesthesia. *International Journal of Control.* 71, 239-257.

[7] Matsumura, S., Iwahara, T., Ogata, K., Fujii, S., and Suzuki, M., 1997, Improvement of de-NOx device control performance using software sensor. *Proceedings of the 11th IFAC Symposium on System Identification*, Fukuoka, Japan, pp. 1507-1512.

[8] Peng, H., Ozaki, T., Toyoda, Y., 2002, Nonlinear predictive control based on a global model identified off-line, *Proceedings of the American Control Conference,* Anchorage, AK, USA, 4197/4202.

[9] Peng, H., Ozaki, T., Haggan-Ozaki, V., and Toyoda, Y., 2003, A parameter optimization method for the radial basis function type models. *IEEE Trans. on Neural Networks, in press.*

Appendix

The matrixes $\hat{\mathbf{A}}(t-1)$, $\hat{\mathbf{B}}(t-1)$, $\hat{\mathbf{D}}(t-1)$ and $\mathbf{L}$ in model (14) are defined as follows:

$$\hat{\mathbf{A}}(t)=\begin{bmatrix} -a_1(t) & -a_2(t) & \cdots & -a_{n_y-1}(t) & -a_{n_y}(t) & b_2(t) & b_3(t) & \cdots & b_{n_u-1}(t) & b_{n_u}(t) \\ 1 & 0 & \cdots & 0 & 0 & & & & & \\ 0 & 1 & \cdots & 0 & 0 & & & \mathbf{0} & & \\ \vdots & \vdots & \ddots & \vdots & \vdots & & & & & \\ 0 & 0 & \cdots & 1 & 0 & & & & & \\ & & & & & 0 & 0 & \cdots & 0 & 0 \\ & & & & & 1 & 0 & \cdots & 0 & 0 \\ & & \mathbf{0} & & & \vdots & \vdots & & \vdots & \vdots \\ & & & & & 0 & 0 & \cdots & 1 & 0 \end{bmatrix}, \quad \hat{\mathbf{D}}(t)=\begin{bmatrix} d_1(t) & d_2(t) & \cdots & d_{n_v}(t) \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}$$

$$\hat{\mathbf{B}}(t)=\begin{bmatrix} b_1(t) & 0 & \cdots & 0 & 1 & 0 & \cdots & 0 \end{bmatrix}^{\mathrm{T}}, \qquad \mathbf{L}=\begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^{\mathrm{T}}.$$