We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

On-line learning of fuzzy rule emulated networks for a class of unknown nonlinear discrete-time controllers with estimated linearization

Chidentree Treesatayapun
Department of Robotic and Manufacturing, CINVESTAV-Saltillo, Ramos Arizpe, 25900
Mexico
Department of Electrical Engineering, North-Chiangmai University, Chiangmai, 50230
Thailand

1. Introduction

Recently, the linearization of a class of unknown discrete-time dynamic systems has achieved considerable topics for the controller design. The unknown functions after system linearization have been estimated by several methods including artificial intelligence techniques such as neural networks, fuzzy logic systems and neurofuzzy networks. In a number of published articles, the issues of system theoretic analysis have been introduced and addressed in the topics of stabilization, tracking performance and the bounded parameters. For all of these cases, the results are validated in the domain around the equilibrium point or state (9; 11). These methods of linearization including local linearization, Taylor series expansion and feedback linearization impose Lipschitz conditions (4; 6; 10; 14; 18). The closed-loop system stability and tracking error have been analyzed in the case of neural network adaptive control (5; 7) but during the learning phase the stability and convergence can not be ensured because of the special conditions. The system stability or bounded signals analysis has been verified (1; 13) and references therein. However, these nonlinear systems under control should be obtained in the format as $y(k+1) = f(k) + g(k)u(k)$ when $y(k)$ and $u(k)$ are the system output and the control input at time index k, respectively and $f(k)$ and $g(k)$ are unknown nonlinear functions. The small learning rate is often defined to solve the stability problem but the convergence is very slow. The discrete-time projection has been introduced for adaptive control systems in (16). The node number of multi-layer neural networks can take more effect of closed-loop stability and tracking performance. In (15), the unknown nonlinear part has been compensated by neural networks and the closed-loop system stability has been also guaranteed for a class on discrete-time systems. Nevertheless, this algorithm needs the renovation when the operating point is changed. In the case of robust system, the dead-zone function has been applied for feedback linearization systems (8) but this control algorithm are only limited for the system with slow trajectory tracking. In this chapter, we discuss about the controller for a class of nonlinear discrete-time systems with estimated unknown nonlinear functions by Muti-input Fuzzy Rules Emulated Networks (MIFRENs). These nonlinear functions are occurred when

the control law is constructed and they are completely unknown as *a priori*. All adjustable parameters inside MIFRENs are automatically tuned by the proposed leaning algorithm. By the theoretical analysis, these parameters are all bounded during the system operation with out any request of off-line learning phase. The closed-loop tracking error is also bounded by the universal function approximation of MIFREN.

2. Preliminaries

2.1 Formulation of Nonlinear discrete-time systems

In this work, we devote our interest in to the discrete-time systems which can be described by

$$y(k+1) = f(p(k), u(k)), \tag{1}$$

where $f(\cdot, \cdot)$ is an unknown nonlinear function, k is time index, $y(k) \in R$ denotes the measurable output, $u(k) \in R$ is the control effort and $p(k) = [y(k), y(k-1), \ldots, y(k-n+1), u(k-1), u(k-2), \ldots, u(k-m+1)]$ when $m \leq n$. For system design in the next section, these following assumptions are still needed

Assumption : System derivative Let define two compact sets Ω_y and Ω_u for the system output y and the control effort u, respectively. The derivative of $f(\cdot, \cdot)$ in (1) with respect to the control effort $u(k)$ is always existed $\forall k = 1, 2, \cdots$ and $0 < |\frac{\partial f(\cdot, u)}{\partial u}| \leq \bar{y}_u$ when $y(\cdot) \in \Omega_y$ and $u(\cdot) \in \Omega_u$ where $\bar{y}_u$ is a finite positive value.

Assumption : Existence of controller For any desired trajectories $r(k)$, let the ideal control effort of the system (1) $u^*(k)$ be existed by

$$u^*(k) = g_u(p(k), r(k+1)), \tag{2}$$
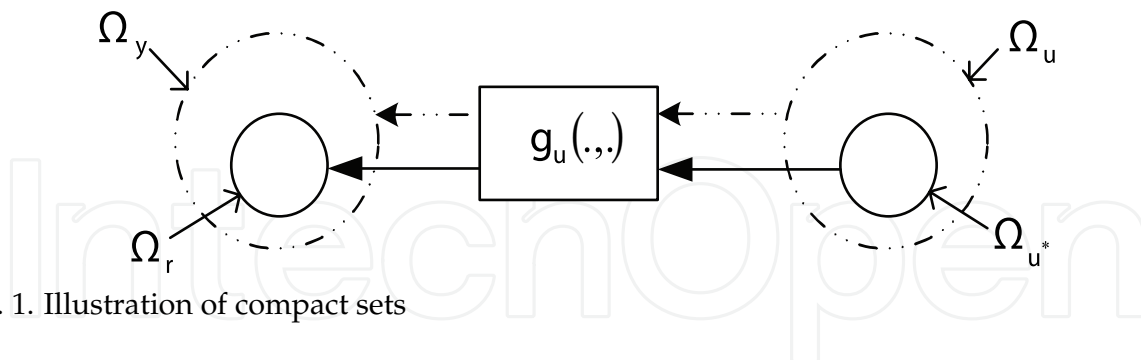
when $g_u(\cdot, \cdot)$ is a smooth function.

Fig. 1. Illustration of compact sets

With the ideal control effort obtained by (2), the controlled system can provide the output to be the desired trajectory as

$$r(k+1) = f(p(k), u^*(k)). \tag{3}$$

Let $u^*(k) \in \Omega_{u^*}$ and $r(k) \in \Omega_r$, for the output $y(k) \in \Omega_y$ such that $\Omega_r \subset \Omega_y$. The function $g_u(-)$ is a one-to-one mapping function of Ω_r into Ω_{u^*}, that is $\Omega_{u^*} \subset \Omega_u$. With the last assumption, $g_u(-)$ is smooth and Ω_r is a compact set, then Ω_{u^*} must be a compact set also. The clearly illustration is given in Fig. 1.

2.2 Function Approximation with MIFREN

In (2) and (3), the function approximation MIFREN property had been introduced. An unknown nonlinear function $f_u(.)$ can be estimated by MIFREN as

$$f_u(k) = \beta^T F_\mu(y(k), \cdots, y(k - \hat{n} - 1), \cdots, u(k - \hat{m} - 1)) + \varepsilon(k), \tag{4}$$

where β^T is the target linear parameter of MIFREN, $F_\mu(.)$ is the rule vector at MIFREN's rule-layer $\hat{n}$ and $\hat{m}$ are designed delay-order integers for y and u, respectively and $\varepsilon(k)$ stands for the MIFREN function approximation error. Eventually, the using function approximation result of MIFREN can be given as

$$\hat{f}_u(k) = \hat{\beta}^T(k) F_\mu(y(k), \cdots, y(k - \hat{n} - 1), \cdots, u(k - \hat{m} - 1)), \tag{5}$$

when $\hat{\beta}(k)$ is the actual linear parameter vector of MIFREN. The vector $\hat{\beta}(k)$ can be automatically tuned via the proposed algorithm as will be discussed in the next section. In this subsection, it will be shown that MIFREN has the property of a universal function approximation using the Stone-Weierstrass theorem (1; 17).

Theorem 2.1 (Universal function approximation of MIFREN). *Let Ω be a compact space of N dimensions and let $\mathcal{F}$ be a set of real functions on a compact set Ω. If*

(1) $\mathcal{F}$ is an algebra,

(2) $\mathcal{F}$ separates points on Ω, and

(3) $\mathcal{F}$ vanishes at no point on Ω,

then $\mathcal{F}$ is dense in $C(\Omega)$, the set of continuous real-valued function on Ω. In other words, for any $\hat{\varepsilon} > 0$ and any function f in $C(\Omega)$, there is a function $\hat{f}$ in $\mathcal{F}$ such that $|f(x) - \hat{f}(x)| < \hat{\varepsilon}$ for all $x \in \Omega$.

Proof: The proof is omitted here moreover for the interested reader can refer to (2) and (3).

$\square$

3. Controller design

In this section, the controller for system given in (1) is constructed with the approximated linearization and MIFRENs approximation.

3.1 Control law based on system linearization

From the system equation described in (1), let use the second-order Taylor expansion with the mean value theorem, we have

$$
\begin{aligned}
y(k+1) &= f(p(k), u(k-1)) + \frac{\partial f(p(k), u)}{\partial u}\bigg|_{u=u(k-1)} \Delta u(k) \\
&\quad + \frac{1}{2} \frac{\partial^2 f(p(k), u)}{\partial u^2}\bigg|_{u=\bar{u}_k} \Delta u^2(k),
\end{aligned}
\tag{6}
$$

where $\bar{u}_k = \gamma u(k) + (1 - \gamma) u(k-1)$ with $0 \leq \gamma \leq 1$ and $\Delta u(k) = u(k) - u(k-1)$. To simplify, (6) can be rewritten as

$$y(k+1) = f(p(k), u(k-1)) + f_1(p(k), u(k-1)) \Delta u(k) + f_2(p(k), \bar{u}_k) \Delta u^2(k), \tag{7}$$

when $f_1(p(k), u(k-1)) = \frac{\partial f(p(k), u)}{\partial u}\Big|_{u=u(k-1)}$ and $f_2(p(k), \bar{u}_k) = \frac{1}{2}\frac{\partial^2 f(p(k), u)}{\partial u^2}\Big|_{u=\bar{u}_k}$. By using (2) and the second assumption mentioned in the previous section, $f_2(\cdot, \cdot)$ can be given by

$$f_2(p(k), \bar{u}_k)\Delta u^2(k) = f_2(p(k), \gamma u(k) + (1-\gamma)u(k-1))[u(k) - u(k-1)]^2, \tag{8}$$

when $u(k)$ can be obtained by

$$u(k) = g_u(p(k), y(k+1)). \tag{9}$$

Substitute (9) into (8), thus $f_2(\cdot, \cdot)$ can be simplified by

$$\begin{aligned}
f_2(p(k), \bar{u}_k)\Delta u^2(k) &= f_2(p(k), \gamma g_u(p(k), y(k+1)) + (1-\gamma)u(k-1)), \\
&\quad \times [g_u(p(k), y(k+1)) - u(k-1)]^2, \\
&= \bar{f}_2(p(k), y(k+1)). \tag{10}
\end{aligned}$$

By substituting (10) into (7), we have

$$\begin{aligned}
y(k+1) &= f(p(k), u(k-1)) + f_1(p(k), u(k-1))\Delta u(k) + \bar{f}_2(p(k), y(k+1)), \\
&= f_3(p(k), y(k+1)) + f_1(p(k), u(k-1))\Delta u(k), \tag{11}
\end{aligned}$$

where $f_3(p(k), y(k+1)) = f(p(k), u(k-1)) + \bar{f}_2(p(k), y(k+1))$. In (11), clearly, we have been forced with the causality problem. Fortunately, with the second assumption, the ideal control effort $u^*(k)$ can provide $r(k+1)$ as described in (3), thus we have

$$r(k+1) = f_3(p(k), r(k+1)) + f_1(p(k), u(k-1))[u^*(k) - u(k-1)]. \tag{12}$$

To continue our design procedure, the ideal control effort $u^*(k)$ can be obtained by

$$\begin{aligned}
u^*(k) &= u(k-1) + \frac{r(k+1) - f_3(p(k), r(k+1))}{f_1(p(k), u(k-1))}, \\
&= u(k-1) + \frac{1}{f_1(p(k), u(k-1))}r(k+1) - \frac{f_3(p(k), r(k+1))}{f_1(p(k), u(k-1))}, \tag{13}
\end{aligned}$$

or

$$u^*(k) = u(k-1) + f_1^*(p(k))r(k+1) - f_2^*(p(k), r(k+1)), \tag{14}$$

when $f_1^*(p(k)) = \frac{1}{f_1(p(k), u(k-1))}$ and $f_2^*(p(k), r(k+1)) = \frac{f_3(p(k), r(k+1))}{f_1(p(k), u(k-1))}$. From the control law given by (14), the singularity problem of $\frac{1}{f_1(p(k), u(k-1))}$ can be avoided by MIFREN approximation which will be discussed later. Let us consider the ideal control effort in (14), thus these nonlinear functions $f_1^*(\cdot, \cdot)$ and $f_2^*(\cdot, \cdot)$ are unknown. In this work, two MIFRENs are constructed to approximate $f_1^*(\cdot, \cdot)$ and $f_2^*(\cdot, \cdot)$ by MIFREN$_1$ and MIFREN$_2$, respectively. We have

$$u^*(k) = u(k-1) + [\beta_1^{*T}F_1(p(k)) + \varepsilon_1(k)]r(k+1) - \beta_2^{*T}F_2(p(k), r(k+1)) - \varepsilon_2(k), \tag{15}$$

where $F_1(\cdot)$ and $F_2(\cdot)$ are rule-functions of MIFREN$_1$ and MIFREN$_2$, respectively, $\beta_1^* = [\beta_{1,1}^* \quad \beta_{1,2}^* \quad \cdots \quad \beta_{1,n_1}^*]^T$, $\beta_2^* = [\beta_{2,1}^* \quad \beta_{2,2}^* \quad \cdots \quad \beta_{2,n_2}^*]^T$ are ideal weight vectors, n_1 and n_2 denote number of rules for each MIFREN and $\varepsilon_1(\cdot)$ and $\varepsilon_2(\cdot)$ are approximation errors. Let

us neglect these errors and use the actual weight vector as $\beta_1(k)$ and $\beta_2(k)$ thus the proposed control law can be given by

$$u(k) = u(k-1) + [\beta_1^T(k)F_1(p(k))]r(k+1) - \beta_2^T(k)F_2(p(k), r(k+1)). \tag{16}$$

With this control equation, the causality problem has been solved by the MIFRENs approximation of unknown nonlinear functions. In the next subsection, the system performance will be analyzed with the designed parameters and main theorem.

3.2 Feedback system error

To guarantee the system performance, we need to design some parameters and theirs operating regions. Let the control error be defined by

$$e(k) = r(k) - y(k), \tag{17}$$

or

$$e(k+1) = r(k+1) - y(k+1), \tag{18}$$

for time index $k+1$. Substitute $y(k+1)$ from (11) into (18), we have

$$e(k+1) = r(k+1) - f_3(p(k), y(k+1)) - f_1(p(k))\Delta u(k). \tag{19}$$

By using Taylor expression and mean value theorem, the control error in (19) can be obtained as

$$
\begin{aligned}
e(k+1) &= r(k+1) - \left[f_3(p(k), r(k+1)) + \frac{\partial f_3(p(k), y)}{\partial y}\bigg|_{y=\bar{y}_{k+1}} (y(k+1) \right. \\
&\quad \left. -r(k+1)) \right] - f_1(p(k))\Delta u(k),
\end{aligned}
\tag{20}
$$

where $\bar{y}_{k+1}$ is between $r(k+1)$ and $y(k+1)$. Let us consider the system in (11) with the control effort given by (9), we have

$$y(k+1) = f_3(p(k), y(k+1)) + f_1(p(k))[g_u(p(k), y(k+1)) - u(k-1)]. \tag{21}$$

From (21), we can reconsider into two cases as these followings:

Case I In this case, we assume that $y(k+1) = r(k+1)$ and take the derivative with respect to $y(k+1)$ for the both sides of (21) thus we have

$$1 = \frac{\partial f_3(p(k), y(k+1))}{\partial y(k+1)} + f_1(p(k))\frac{\partial g_u(p(k), y(k+1))}{\partial y(k+1)}. \tag{22}$$

Case II For this second case, we reconsider (21) again with $y(k+1) \neq r(k+1)$, take the derivative with respect to $y(k+1)$ for the both sides of (21) and use Tayler expansion

with the mean value theorem thus we have

$$
\begin{aligned}
y(k+1) &= f_3(p(k),y(k+1)) + f_1(p(k))[g_u(p(k),y(k+1)) - u(k-1)], \\
&= f_3(p(k),r(k+1)) + \left.\frac{\partial f_3(p(k),y)}{\partial y}\right|_{y=\bar{y}(k+1)}[y(k+1) - r(k+1)] \\
&\quad + f_1(p(k))\Big[g_u(p(k),r(k+1)) + \left.\frac{\partial g_u(p(k),y)}{\partial y}\right|_{y=\check{y}(k+1)} \\
&\quad \times [y(k+1) - r(k+1)] - u(k-1)\Big], \\
&= f_3(p(k),r(k+1)) + f_1(p(k))[g_u(p(k),r(k+1)) - u(k-1)] \\
&\quad + \left.\frac{\partial f_3(p(k),y)}{\partial y}\right|_{y=\bar{y}(k+1)}[y(k+1) - r(k+1)] \\
&\quad + f_1(p(k))\left.\frac{\partial g_u(p(k),y)}{\partial y}\right|_{y=\check{y}(k+1)}[y(k+1) - r(k+1)].
\end{aligned}
\tag{23}
$$

From (12) and $u^*(k) = g_u(p(k),r(k+1))$, we can rearrange (23) to be

$$
\begin{aligned}
y(k+1) - r(k+1) &= + \left.\frac{\partial f_3(p(k),y)}{\partial y}\right|_{y=\bar{y}(k+1)}[y(k+1) - r(k+1)] \\
&\quad + f_1(p(k))\left.\frac{\partial g_u(p(k),y)}{\partial y}\right|_{y=\check{y}(k+1)}[y(k+1) \\
&\quad - r(k+1)].
\end{aligned}
\tag{24}
$$

With the previous assumption, we still have $\bar{y}(k+1) = \check{y}(k+1)$ and $y(k+1) \neq r(k+1)$ thus (24) can be rewritten as

$$
1 = \frac{\partial f_3(p(k),y(k+1))}{\partial y(k+1)} + f_1(p(k))\frac{\partial g_u(p(k),y(k+1))}{\partial y(k+1)}.
\tag{25}
$$

Taking the results from the both cases, the following relation can be obtained

$$
\frac{\partial f_3(p(k),y(k+1))}{\partial y(k+1)} = 1 - f_1(p(k),u(k-1))\frac{\partial g_u(p(k),y(k+1))}{\partial y(k+1)}.
\tag{26}
$$

Substitute (26) into (20), we have

$$
\begin{aligned}
e(k+1) &= r(k+1) - f_3(p(k),r(k+1)) + e(k+1) - f_1(p(k))\Delta u(k) \\
&\quad - f_1(p(k))\left.\frac{\partial g_u(p(k),y)}{\partial y}\right|_{y=\bar{y}(k+1)}e(k+1),
\end{aligned}
\tag{27}
$$

or

$$
\begin{aligned}
f_1(p(k))\left.\frac{\partial g_u(p(k),y)}{\partial y}\right|_{y=\bar{y}(k+1)}e(k+1) &= r(k+1) - f_3(p(k),r(k+1)) \\
&\quad - f_1(p(k))\Delta u(k).
\end{aligned}
\tag{28}
$$

For the controllable system in (11), clearly, $f_1(p(k)) \neq 0$ and $u^*(k) = g_u(p(k), r(k+1))$ or
$u(k) = g_u(p(k), y(k+1))$ thus the system sensibility $[\frac{\partial y}{\partial u}]^{-1}$ should be obtained as

$$
\begin{aligned}
\frac{\partial u(k)}{\partial y}\Big|_{y=y(k+1)} &= \frac{\partial g_u(p(k), y)}{\partial y}\Big|_{y=y(k+1)}, \\
&= \frac{1}{\gamma_y(k)}.
\end{aligned}
\tag{29}
$$

The next time-index error can be rewritten again as

$$
e(k+1) = \gamma_y(k)\left[\frac{r(k+1)}{f_1(p(k))} - \frac{f_3(p(k), r(k+1))}{f_1(p(k))} - \Delta u(k)\right].
\tag{30}
$$

Rearrange (30) with MIFRENs approximation given by (15), we have

$$
\begin{aligned}
e(k+1) &= \gamma_y(k)\left[\beta_1^{*T} F_1(p(k)) r(k+1) - \beta_2^{*T} F_2(p(k), r(k+1)) - \Delta u(k)\right] \\
&\quad + \gamma_y(k)\left[\varepsilon_1(k) r(k+1) - \varepsilon_2(k)\right].
\end{aligned}
\tag{31}
$$

Substitute the proposed control law (16) into (31), we obtain

$$
\begin{aligned}
e(k+1) &= \gamma_y(k)\Big[\beta_1^{*T} F_1(p(k)) r(k+1) - \beta_2^{*T} F_2(p(k), r(k+1)) \\
&\quad - \beta_1^T(k) F_1(p(k)) r(k+1) + \beta_2^T(k) F_2(p(k), r(k+1))\Big] \\
&\quad + \gamma_y(k)\Big[\varepsilon_1(k) r(k+1) - \varepsilon_2(k)\Big], \\
&= \gamma_y(k)\Big[\widetilde{\beta}_1^T(k) F_1(k) r(k+1) - \widetilde{\beta}_2^T(k) F_2(k)\Big] + \gamma_y(k)\varepsilon_t(k),
\end{aligned}
\tag{32}
$$

when $\widetilde{\beta}_i^T(k) = \beta_i^{*T} - \beta_i^T(k)$ for $i = 1, 2$ and $\varepsilon_t(k) = \varepsilon_1(k) r(k+1) - \varepsilon_2(k)$.

3.3 MIFRENs tuning laws

The parameter vectors $\beta_1(k)$ and $\beta_2(k)$ are required to update during the system operation or
on-line learning. To simplify, let us rewrite (32) to be

$$
e(k+1) = \gamma_y(k)\begin{bmatrix} \widetilde{\beta}_1^T(k) & \widetilde{\beta}_2^T(k) \end{bmatrix} F(k) + \gamma_y(k)\varepsilon_t(k),
\tag{33}
$$

where $F(k) = \begin{bmatrix} F_1(k) r(k+1) \\ -F_2(k) \end{bmatrix}$. With (33), we can define the update law as the following:

$$
\begin{bmatrix} \beta_1(k+1) \\ \beta_2(k+1) \end{bmatrix} = \begin{bmatrix} \beta_1(k) \\ \beta_2(k) \end{bmatrix} + \frac{\eta}{\bar{\gamma_u}||F(k)||^2} F(k)\mathfrak{D}(e(k)),
\tag{34}
$$

where η is the selected learning rate which will be discussed next and $\mathfrak{D}(\cdot)$ is the dead-zone
function which can be defined by

$$
\mathfrak{D}(e(k)) = \begin{cases} e(k) - \varepsilon_m & \text{if } e(k) > \varepsilon_m \\ 0 & \text{if } |e(k)| \leq \varepsilon_m \\ e(k) + \varepsilon_m & \text{if } e(k) < -\varepsilon_m, \end{cases}
\tag{35}
$$

when $|\gamma_y(k)\varepsilon_t(k)| \leq \varepsilon_m$ as a small positive number. In the case of $|e(k-1)| > \varepsilon_m$, with the dead-zone function (35) and the next time-index error (33), we have

$$\mathfrak{D}(e(k+1)) = \alpha_D \gamma_y(k) \begin{bmatrix} \tilde{\beta}_1^T(k) & \tilde{\beta}_2^T(k) \end{bmatrix} F(k), \tag{36}$$

where $0 < \alpha_D \leq 1$.

3.4 System analysis

To analyze the system performance and stability, the bounded weight vectors $\tilde{\beta}_i^T(k)$ and the bounded tracking error $e(k)$ are both given in this work.

Lemma 1: For the nonlinear discrete-time system given in (1) with the control law defined in (16), the error weight vectors $\tilde{\beta}_i^T(k)$ for $i = 1, 2$ are bounded by the tuning law in (34) and the selected learning rate η as the followings:

$$0 < \eta < \frac{2\bar{y_u}}{\alpha_D \gamma_y(k)}, \tag{37}$$

when $0 < \bar{y_u}$, and

$$\frac{2\bar{y_u}}{\alpha_D \gamma_y(k)} < \eta < 0, \tag{38}$$

when $\bar{y_u} < 0$.

Proof: Let us define a Lyapunov candidate function as

$$V_{\tilde{\beta}}(k) = \tilde{\beta}_1^T(k)\tilde{\beta}_1(k) + \tilde{\beta}_2^T(k)\tilde{\beta}_2(k). \tag{39}$$

The first difference can be obtained by

$$\begin{aligned} \Delta V_{\tilde{\beta}}(k) &= V_{\tilde{\beta}}(k+1) - V_{\tilde{\beta}}(k), \\ &= \tilde{\beta}_1^T(k+1)\tilde{\beta}_1(k+1) + \tilde{\beta}_2^T(k+1)\tilde{\beta}_2(k+1) - \tilde{\beta}_1^T(k)\tilde{\beta}_1(k) \\ &\quad - \tilde{\beta}_2^T(k)\tilde{\beta}_2(k). \end{aligned} \tag{40}$$

Let us define $\tilde{\beta}_\Sigma(k+1) = \tilde{\beta}_1^T(k+1)\tilde{\beta}_1(k+1) + \tilde{\beta}_2^T(k+1)\tilde{\beta}_2(k+1)$, from the tuning law given by (34) and $\tilde{\beta}_i^T(k) = \beta_i^{*T} - \beta_i^T(k)$, we have

$$\begin{aligned} \tilde{\beta}_\Sigma(k+1) &= \tilde{\beta}_1^T(k)\tilde{\beta}_1(k) + \tilde{\beta}_2^T(k)\tilde{\beta}_2(k) - \frac{2\eta}{\bar{y_u}||F(k)||^2} \begin{bmatrix} \beta_1(k) \\ \beta_2(k) \end{bmatrix}^T F(k) \\ &\quad \times \mathfrak{D}(e(k+1)) + \frac{\eta^2}{\bar{y_u}^2||F(k)||^4}||F(k)||^2\mathfrak{D}^2(e(k+1)). \end{aligned} \tag{41}$$

Substitute (41) into (40) and use (36), we obtain

$$\begin{aligned} \Delta V_{\tilde{\beta}}(k) &= -\frac{2\eta}{\bar{y_u}||F(k)||^2} \begin{bmatrix} \beta_1(k) \\ \beta_2(k) \end{bmatrix}^T F(k)\mathfrak{D}(e(k+1)) + \frac{\eta^2}{\bar{y_u}^2||F(k)||^4}||F(k)||^2 \\ &\quad \times \mathfrak{D}^2(e(k+1)) \\ &= -\frac{2\eta}{\alpha_D \gamma_y(k)\bar{y_u}||F(k)||^2}\mathfrak{D}^2(e(k+1)) + \frac{\eta^2}{\bar{y_u}^2||F(k)||^2}\mathfrak{D}^2(e(k+1)) \\ &= \left[\frac{-2}{\alpha_D \gamma_y(k)} + \frac{\eta}{\bar{y_u}}\right] \frac{\eta}{\bar{y_u}||F(k)||^2}\mathfrak{D}^2(e(k+1)). \end{aligned} \tag{42}$$

On-line learning of fuzzy rule emulated networks for a class of unknown nonlinear discrete-time controllers with estimated linearization

259

With the selected learning rate defined by (37) and (38) and $\gamma_y(k)$ given in (29), the first difference of Lyapunov function is negative, thus $\widetilde{\beta}_i^T(k)$ for $i = 1, 2$ are bounded.

$\square$

Remark: Normally, with out loss of generality, $\bar{y_u}$ is assumed to be positive thus $\gamma_y(k) < \bar{y_u}$: $\forall k$. The bounded tracking error for the closed-loop system is introduced by the following theorem.

Theorem 3.1 (Bounded tracking error). *For the nonlinear discrete-time system given in (1) with the control law defined in (16), let define a compact set $\Omega_\varepsilon = \{e(k) | |e(k) \leq 4\varepsilon_m\}$, thus the ultimate boundary on the tracking error is $\lim_{k \to \infty} |e(k)| \leq \varepsilon_m$ or in a compact set Ω_ε.*

Proof: Let a Lyapunov candidate function be given by

$$V_e(k) = \frac{\eta}{2\bar{y_u}^2 F_o^2} e^2(k) + V_{\widetilde{\beta}}(k), \tag{43}$$

when F_o is defined by $0 < ||F(k)|| \leq F_0, \forall k$. The first difference can be obtained by

$$\begin{aligned}
\Delta V_e(k) &= V_e(k+1) - V_e(k), \\
&= \frac{\eta}{2\bar{y_u}^2 F_o^2}[e^2(k+1) - e^2(k)] + \Delta V_{\widetilde{\beta}}(k).
\end{aligned} \tag{44}$$

Substitute (42) into (44), we have

$$\begin{aligned}
\Delta V_e(k) &= \frac{\eta}{2\bar{y_u}^2 F_o^2}[e^2(k+1) - e^2(k)] - \frac{2\eta \mathfrak{D}^2(e(k+1))}{\alpha_D \gamma_y(k)\bar{y_u}||F(k)||^2} \\
&\quad + \frac{\eta^2 \mathfrak{D}^2(e(k+1))}{\bar{y_u}^2 ||F(k)||^2}.
\end{aligned} \tag{45}$$

From the learning rate given by (37- 37), we can rearrange (45) as

$$\begin{aligned}
\Delta V_e(k) &< \frac{\eta}{2\bar{y_u}^2 F_o^2}e^2(k+1) - \frac{\eta}{\alpha_D \gamma_y(k)\bar{y_u}||F(k)||^2}\mathfrak{D}^2(e(k+1)), \\
&< \frac{\eta}{2\bar{y_u}^2 F_o^2}e^2(k+1) - \frac{\eta}{\bar{y_u}^2 F_o^2}\mathfrak{D}^2(e(k+1)), \\
&= \frac{\eta}{2\bar{y_u}^2 F_o^2}[e^2(k+1) - 2\mathfrak{D}^2(e(k+1))].
\end{aligned} \tag{46}$$

In this proof, we need to provide only the case when $|e(k+1)| > \varepsilon_m$. With $|e(k+1)| > \varepsilon_m$, the dead-zone function in (35) can be obtained as

$$\mathfrak{D}(e(k+1)) = e(k+1) - \varepsilon_m \text{sign}\{e(k+1)\}. \tag{47}$$

Substitute (47) into (46), we have

$$\begin{aligned}
\Delta V_e(k) &< \frac{\eta}{2\bar{y_u}^2 F_o^2}\left[e^2(k+1) - 2[e(k+1) - \varepsilon_m \text{sign}\{e(k+1)\}]^2\right], \\
&= \frac{\eta}{2\bar{y_u}^2 F_o^2}\left[-e^2(k+1) + 4e(k+1)\varepsilon_m \text{sign}\{e(k+1)\} - 2\varepsilon_m^2\right], \\
&= \frac{\eta}{2\bar{y_u}^2 F_o^2}\left[-e^2(k+1) - 2\varepsilon_m^2 + 4|e(k+1)|\varepsilon_m\right], \\
&< \frac{\eta}{2\bar{y_u}^2 F_o^2}\left[-e^2(k+1) + 4|e(k+1)|\varepsilon_m\right].
\end{aligned} \tag{48}$$

Consider the result in (48), clearly, $\Delta V_e(k)$ is always negative where $|e(k+1)| > 4\varepsilon_m$, thus $\Delta V_e(k) < 0$ when $|e(k+1)|$ is out side a compact set Ω_ε.

$\square$

4. Computer simulation example

The proposed control algorithm and theorem are verified by the computer simulation. The selected controllable system is described by

$$y(k+1) = \sin(y(k)) + \cos(y(k)u(k))u(k) + 5u(k). \tag{49}$$

The system performance can be demonstrated into two cases as the nominal system and the robust control.

4.1 Nominal system

With the system displayed in (49), the system parameters can be selected as $\varepsilon_m = 0.001$, $\eta = 0.8$ and $\bar{y}_u = 5$. All IF-THEN rules for both MIFRENs are given by the followings:

	Rule 1	If $y(k)$ is N	and $u(k-1)$ is N	Then $f_{1,1}(k) = \beta_{1,1}(k)F_{1,1}(k)$,
	Rule 2	If $y(k)$ is N	and $u(k-1)$ is Z	Then $f_{1,2}(k) = \beta_{1,2}(k)F_{1,2}(k)$,
	Rule 3	If $y(k)$ is N	and $u(k-1)$ is P	Then $f_{1,3}(k) = \beta_{1,3}(k)F_{1,3}(k)$,
	Rule 4	If $y(k)$ is Z	and $u(k-1)$ is N	Then $f_{1,4}(k) = \beta_{1,4}(k)F_{1,4}(k)$,
MIFREN$_1$	Rule 5	If $y(k)$ is Z	and $u(k-1)$ is Z	Then $f_{1,5}(k) = \beta_{1,5}(k)F_{1,5}(k)$,
	Rule 6	If $y(k)$ is Z	and $u(k-1)$ is P	Then $f_{1,6}(k) = \beta_{1,6}(k)F_{1,6}(k)$,
	Rule 7	If $y(k)$ is P	and $u(k-1)$ is N	Then $f_{1,7}(k) = \beta_{1,7}(k)F_{1,7}(k)$,
	Rule 8	If $y(k)$ is P	and $u(k-1)$ is Z	Then $f_{1,8}(k) = \beta_{1,8}(k)F_{1,8}(k)$,
	Rule 9	If $y(k)$ is P	and $u(k-1)$ is P	Then $f_{1,9}(k) = \beta_{1,9}(k)F_{1,9}(k)$,

	Rule 1	If $y(k)$ is N	and $r(k+1)$ is N	Then $f_{2,1}(k) = \beta_{2,1}(k)F_{2,1}(k)$,
	Rule 2	If $y(k)$ is N	and $r(k+1)$ is Z	Then $f_{2,2}(k) = \beta_{2,2}(k)F_{2,2}(k)$,
	Rule 3	If $y(k)$ is N	and $r(k+1)$ is P	Then $f_{2,3}(k) = \beta_{2,3}(k)F_{2,3}(k)$,
	Rule 4	If $y(k)$ is Z	and $r(k+1)$ is N	Then $f_{2,4}(k) = \beta_{2,4}(k)F_{2,4}(k)$,
MIFREN$_2$	Rule 5	If $y(k)$ is Z	and $r(k+1)$ is Z	Then $f_{2,5}(k) = \beta_{2,5}(k)F_{2,5}(k)$,
	Rule 6	If $y(k)$ is Z	and $r(k+1)$ is P	Then $f_{2,6}(k) = \beta_{2,6}(k)F_{2,6}(k)$,
	Rule 7	If $y(k)$ is P	and $r(k+1)$ is N	Then $f_{2,7}(k) = \beta_{2,7}(k)F_{2,7}(k)$,
	Rule 8	If $y(k)$ is P	and $r(k+1)$ is Z	Then $f_{2,8}(k) = \beta_{2,8}(k)F_{2,8}(k)$,
	Rule 9	If $y(k)$ is P	and $r(k+1)$ is P	Then $f_{2,9}(k) = \beta_{2,9}(k)F_{2,9}(k)$,

when N, Z and P denote negative, zero and positive linguistic levels respectively. The membership functions for these rules are illustrated in Fig. (2) and (3). In this work, we use the same membership functions of $y(k)$ and $r(k+1)$ because these variables have equality linguistic levels in the sense of human.
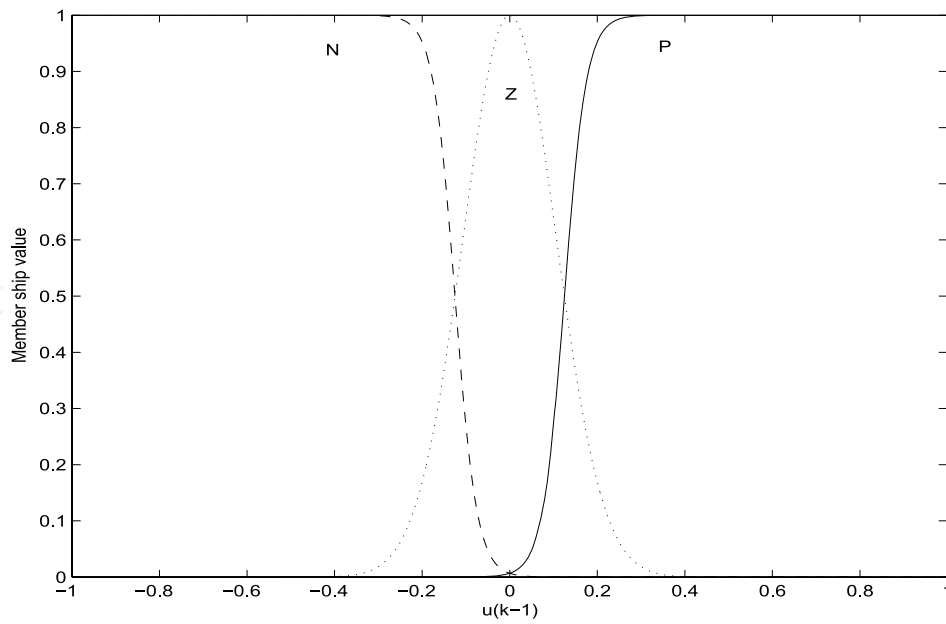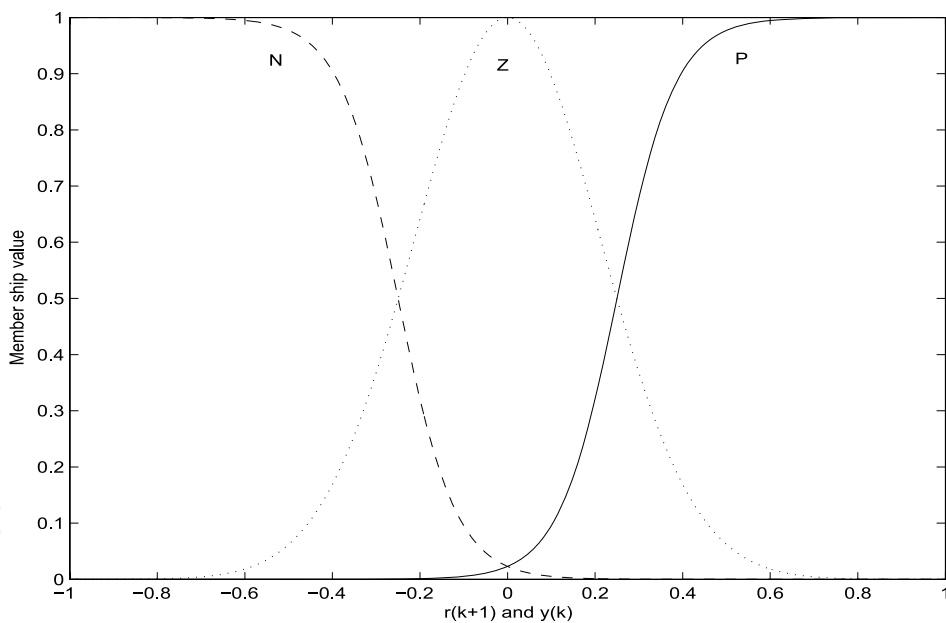
Fig. 2. Membership functions of $u(k-1)$

Fig. 3. Membership functions of $r(k+1)$ and $y(k)$

The initial setting $\beta_{i,j}(1)$ for $i=1,2$ and $j=1,2,\cdots,9$ can be given as

$\beta_{i,1}(1)$=-1	$\beta_{i,2}(1)$=-0.75	$\beta_{i,3}(1)$=-0.5,
$\beta_{i,4}(1)$=-0.25	$\beta_{i,5}(1)$=0	$\beta_{i,6}(1)$=0.25,
$\beta_{i,7}(1)$=0.5	$\beta_{i,8}(1)$=0.75	$\beta_{i,9}(1)$=1.

In Fig. 4, the tracking performance is quite satisfied with out the off-line learning. The control effort is illustrated in Fig. 5. The convergence of $\beta_i(k)$ is shown by $||\beta_i(k)||$ in Fig. 6 for both MIFRENs.
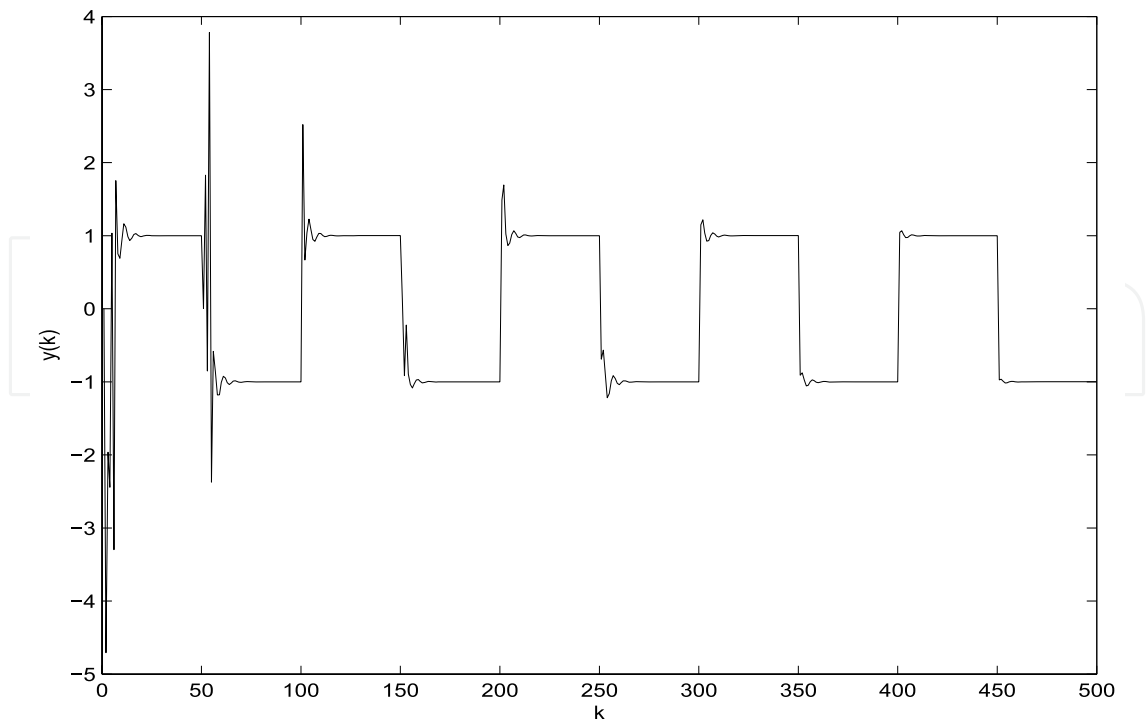
Fig. 4. Tracking performance $y(k)$ for nominal plant.
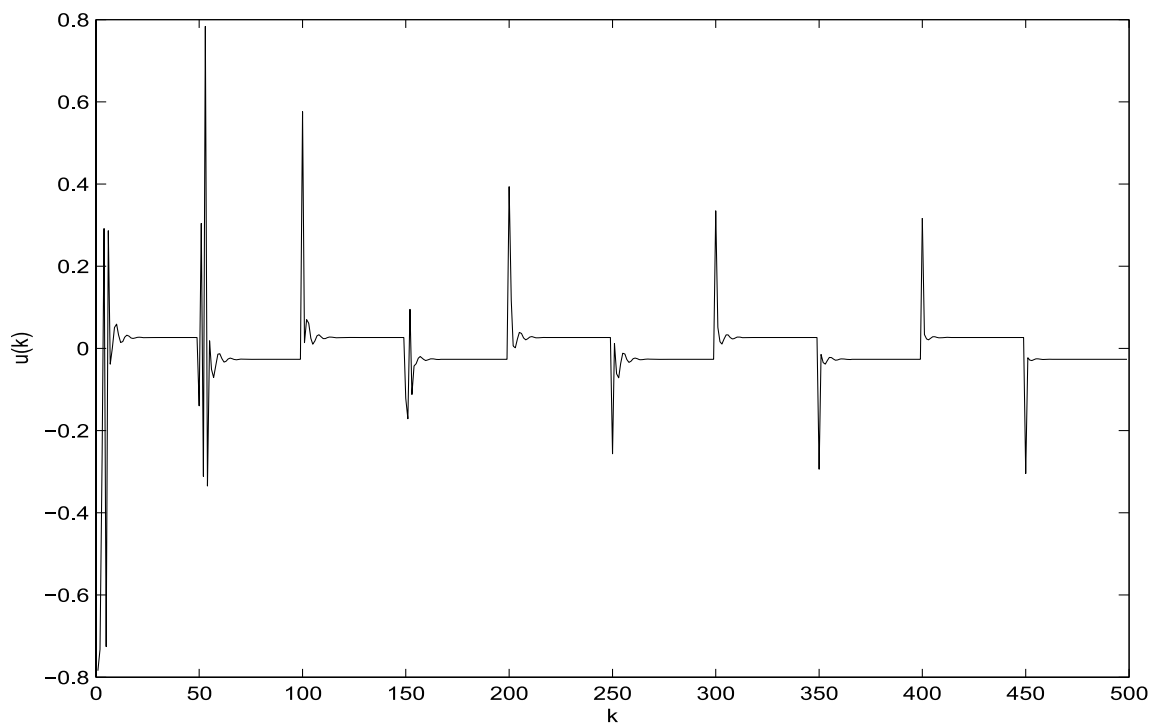
Fig. 5. Control effort $u(k)$ for nominal plant.
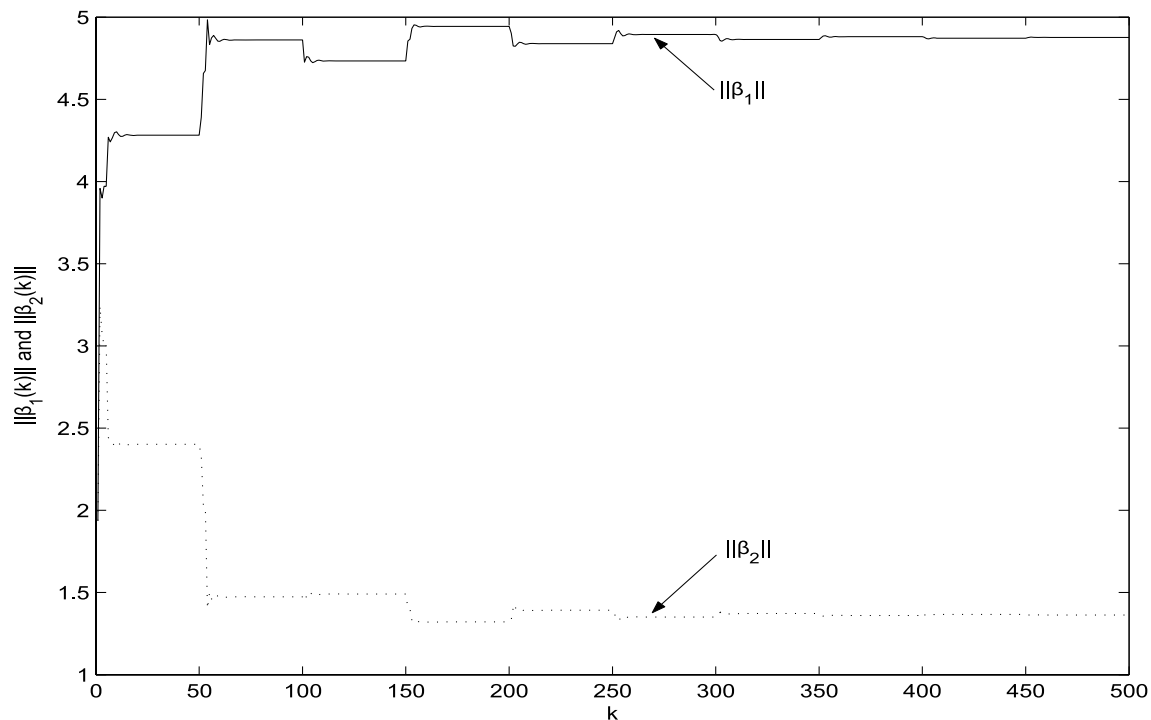
Fig. 6. $||\beta_i(k)||$ for nominal plant.

4.2 Robust control

In the robust system case, the uncertainty terms $\Delta f_1(k)$ and $\Delta f_2(k)$ are included in the system (49) as

$$y(k+1) = \sin(y(k)) + \Delta f_1(k) + \cos(y(k)u(k))u(k) + 5u(k) + \Delta f_2(k)u(k), \qquad (50)$$

when

$$\Delta f_1(k) = \begin{cases} 1 & \text{if } 0 < k < 125 \\ 0.75 & \text{if } 125 \leq k < 325 \\ -1.25 & \text{if } 325 \leq k < 425 \\ 1.25 & \text{if } 425 \leq k < 500, \end{cases} \qquad (51)$$

and

$$\Delta f_2(k) = \begin{cases} -0.5 & \text{if } 0 < k < 125 \\ 1 & \text{if } 125 \leq k < 225 \\ -0.75 & \text{if } 225 \leq k < 425 \\ -0.5 & \text{if } 425 \leq k < 500. \end{cases} \qquad (52)$$

We use the initial setting IF-THEN rules, membership functions, ε_m, η, $\bar{y}_u$ and parameter vectors β_i, as the same as the previous one. With out any off-line learning for MIFRENs, the tracking performance is represented in Fig. 8. The control effort $u(k)$ is shown in Fig. 9. The time variation of $||\beta_i(k)||$ can be illustrated in Fig. 10. These uncertainty terms $\Delta f_1(k)$ and $\Delta f_2(k)$ are varied with time but the tuning vectors are all bounded.
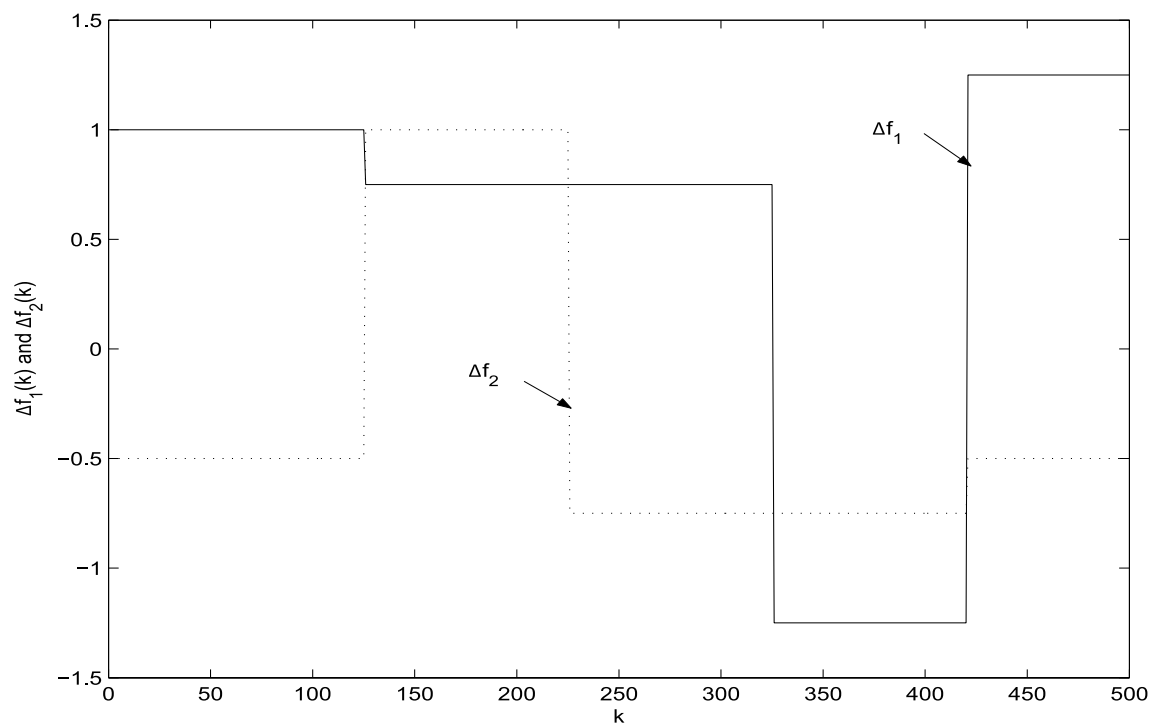
Fig. 7. Illustration of uncertainty $\Delta f_1(k)$ and $\Delta f_2(k)$.
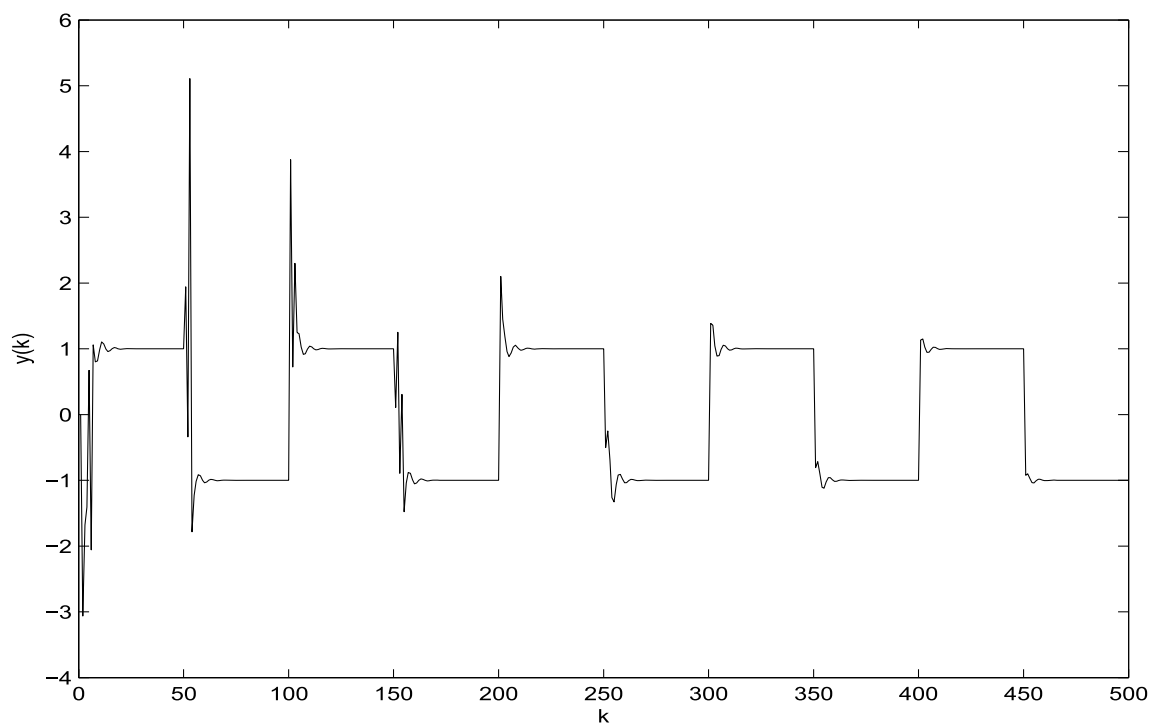
Fig. 8. Tracking performance $y(k)$ for robust system.
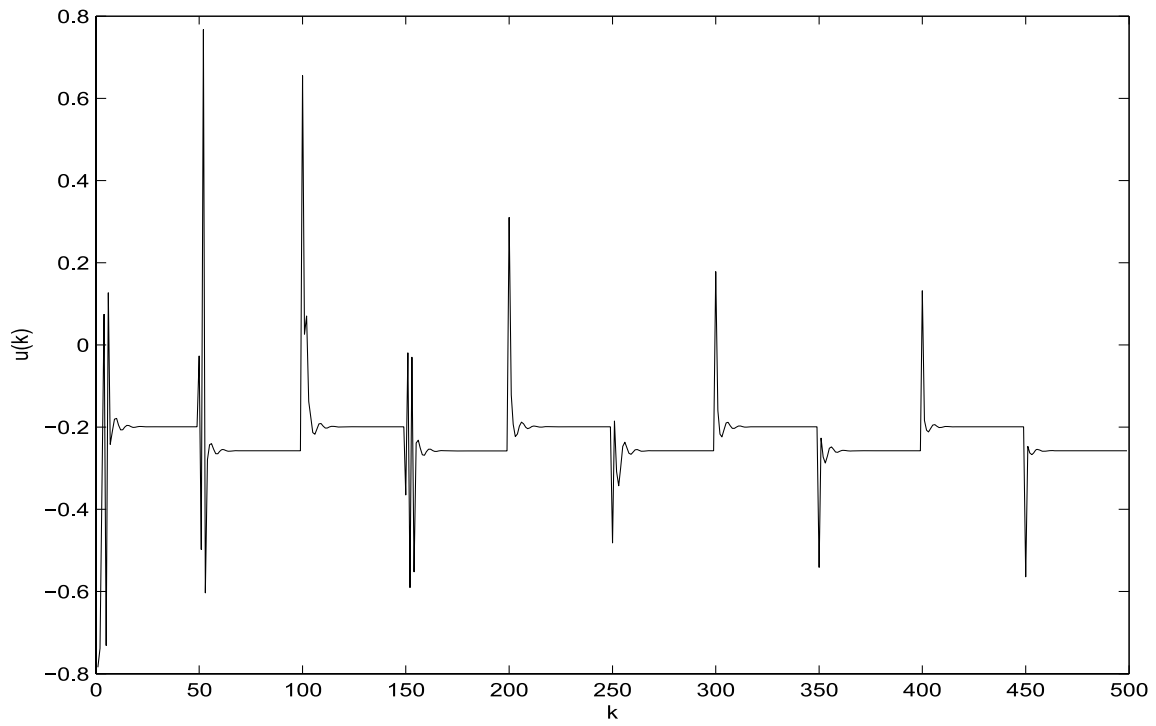
Fig. 9. Control effort $u(k)$ for robust system.
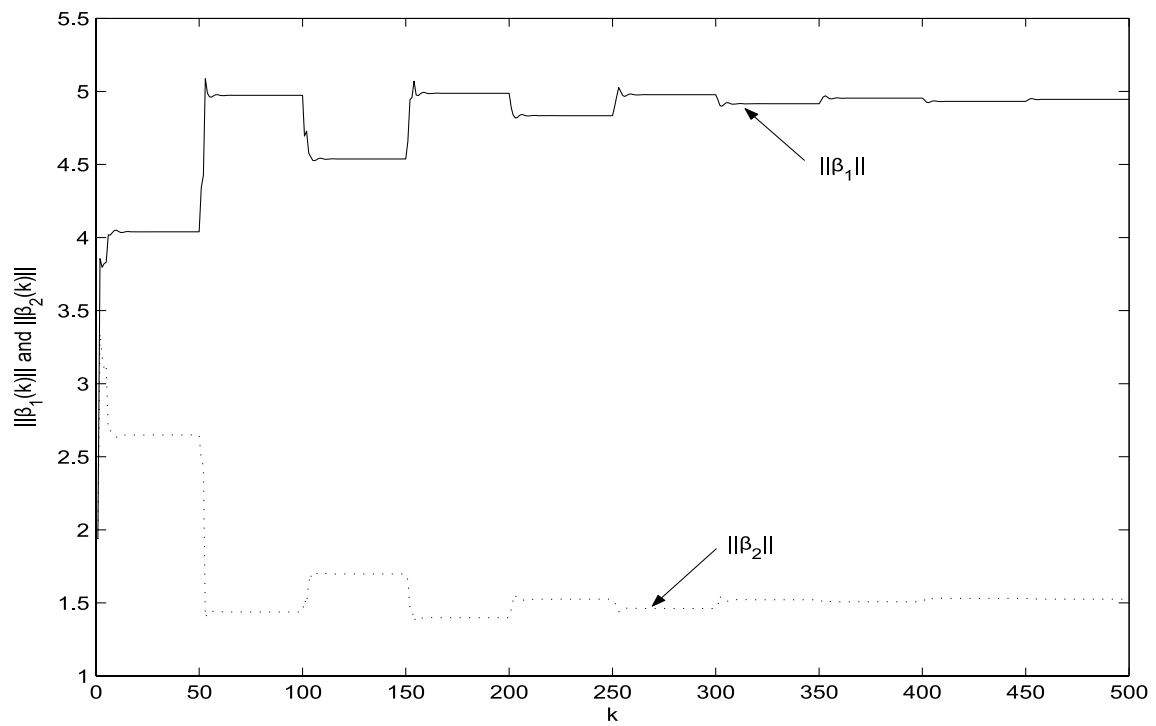
Fig. 10. $||\beta_i(k)||$ for robust system.

5. Experimental setup example

In this section, the performance of our proposed controller is demonstrated by an experimental setup with FESTO mobile robot system called Robotino® . Our task is to design the controller for moving this Robotino® to reach the desired position in (x,y) coordinate as $x_d(i,k)$ and $y_d(i,k)$, respectively. During the movement, the desired angular of Robotino® denoted as $\phi_d(i,k)$ should be maintained as $0°$ for all i^{th} desired position and time index k. The system configuration can be illustrated in Fig. 12 by the block diagram.
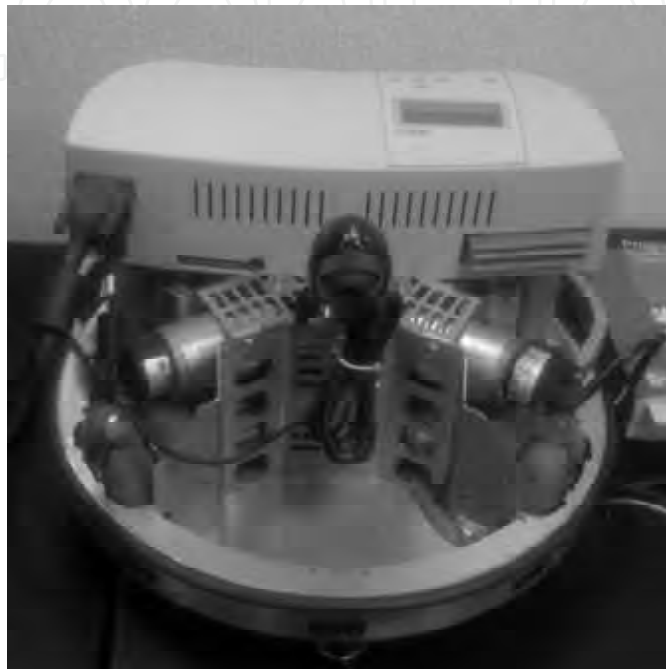
Fig. 11. Robotino.
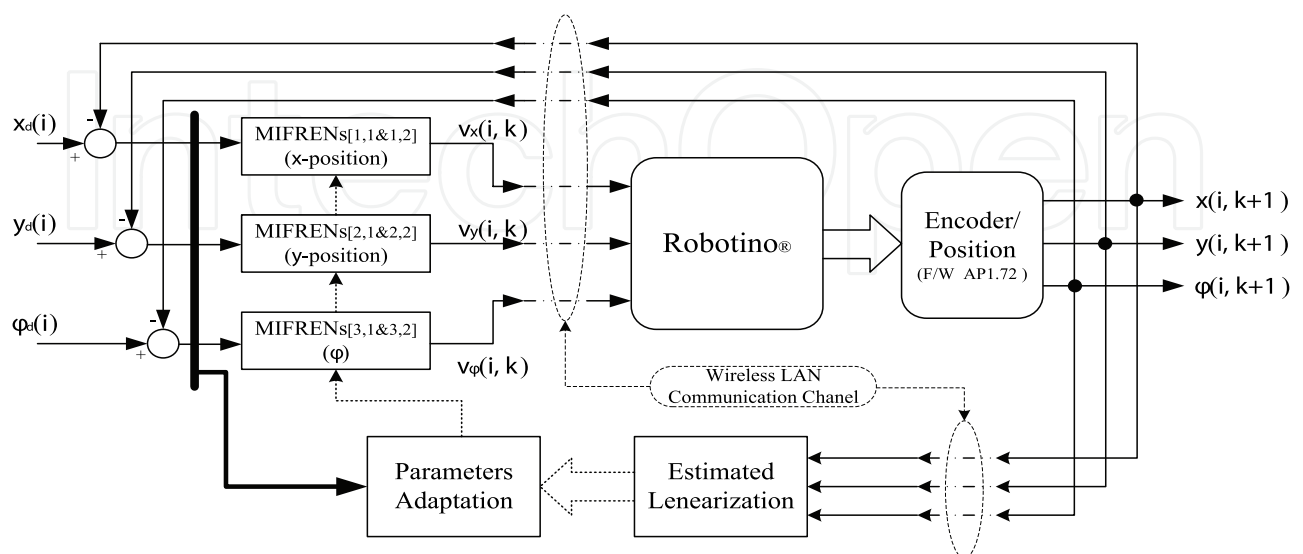
Fig. 12. Block diagram for experimental setup.

The commercial Robotino® needs velocity to control its movement such as velocity in $x-$axis $v_x(i,k)$ for $x-$direction, $v_y(i,k)$ for $y-$direction and $v_{\phi(i,k)}$ for the rotation. In this work, we consider these signals as the control efforts which can be generated by the pair of MIFRENs. The experiment has been demonstrated by 4 desired points and 4 routes as the following: route 1 [(0.0, 0.5)→(0.5, 0)], route 2 [(0.5, 0)→(0.0, 0.0)], route 3 [(0.0, 0.0)→(0.5, 0.5)] and route 4 [(0.5, 0.5)→(0.0, 0.5)]. In Fig. 13, the movement of Robotino® is illustrated in $x-y$ coordinate with errors in x and y axis as e_x and e_y shown in Fig. 14. Because of the fixed angular $\phi_d = 0$, we need to consider only two control efforts v_x and v_y as presented in Fig. 15. At the beginning, on route 1 and 2, the movement of robot is not strange line because MIFRENs need to tune the parameters inside. After that the better results can be obtained in route 3 and 4. In case of losing the wireless signal, we still have the satisfied result as shown in Fig. 16.
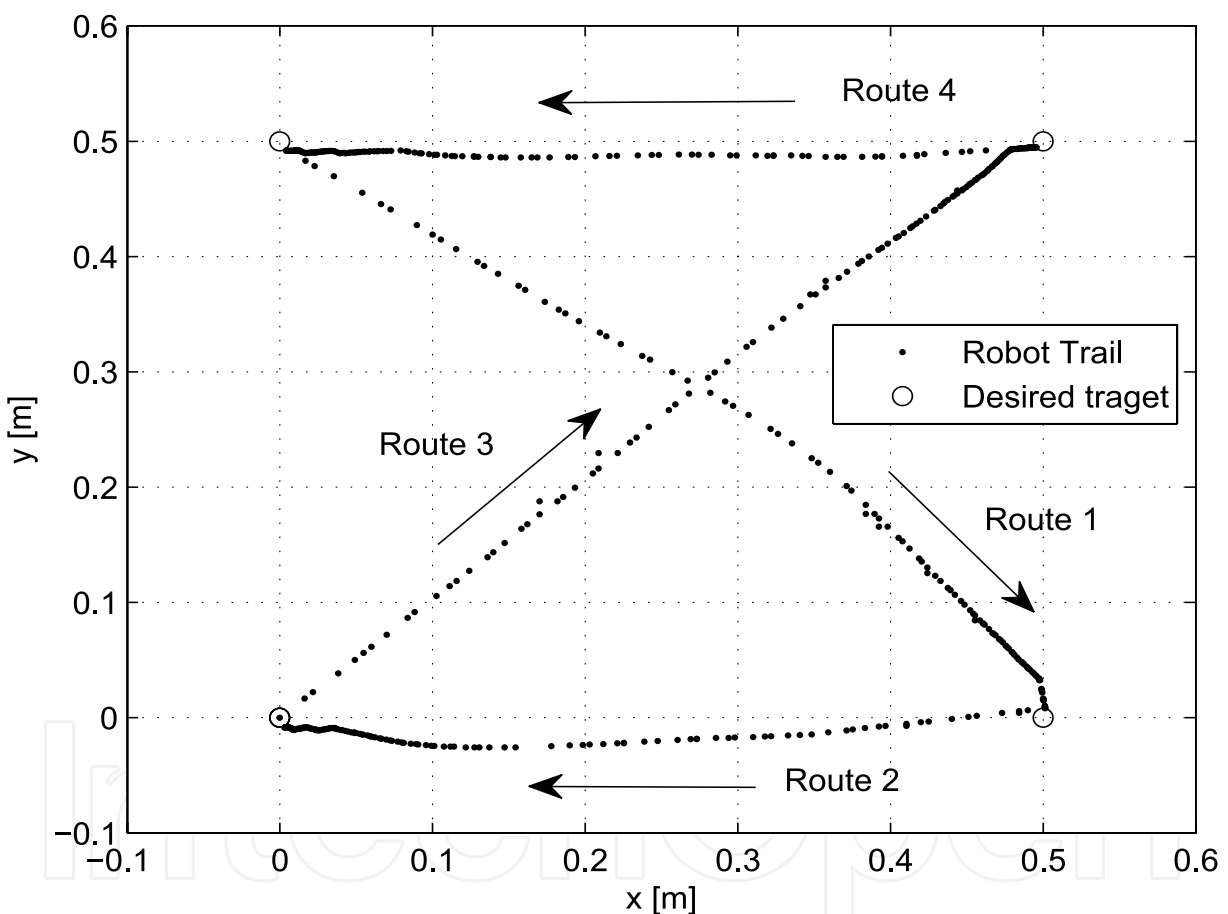
Fig. 13. Experimental result: position $x - y$.
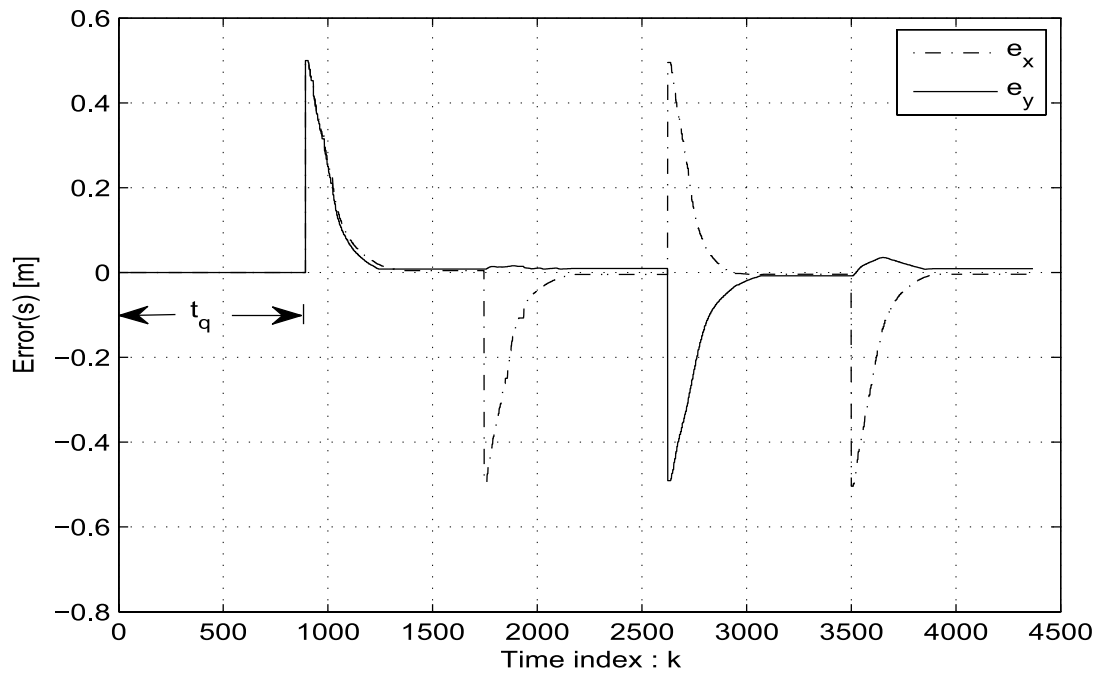
Fig. 14. Experimental result: position errors e_x and e_y.
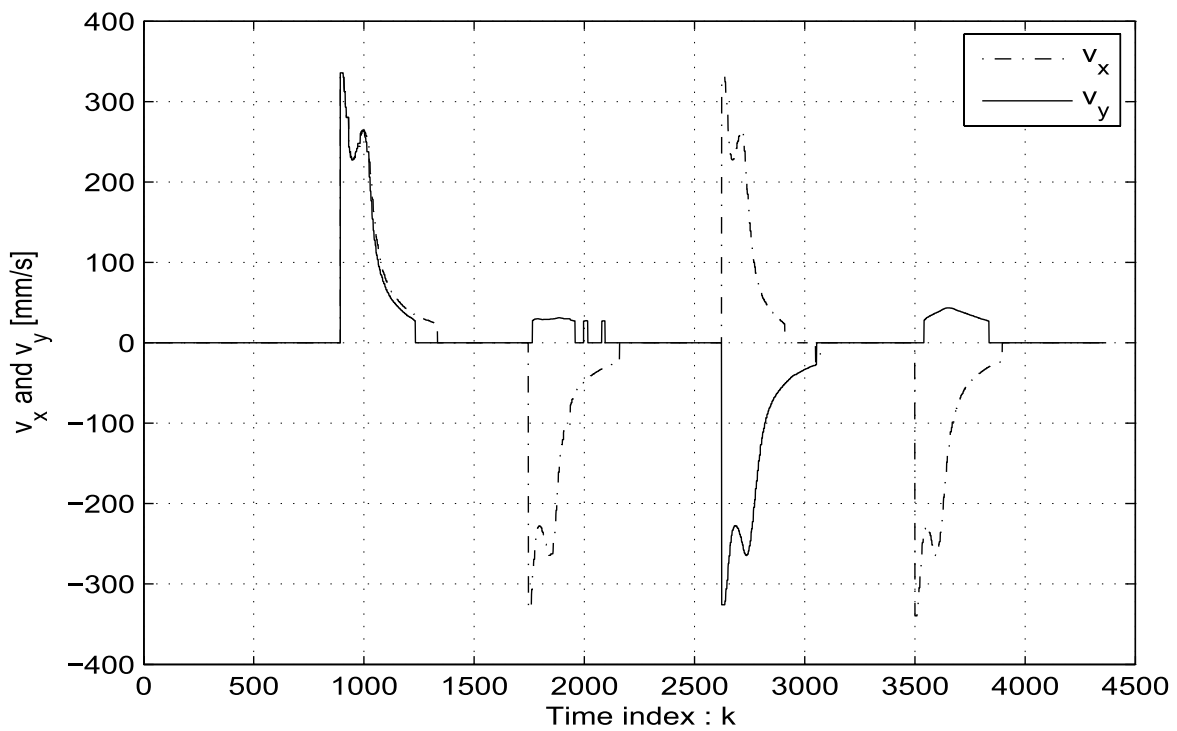
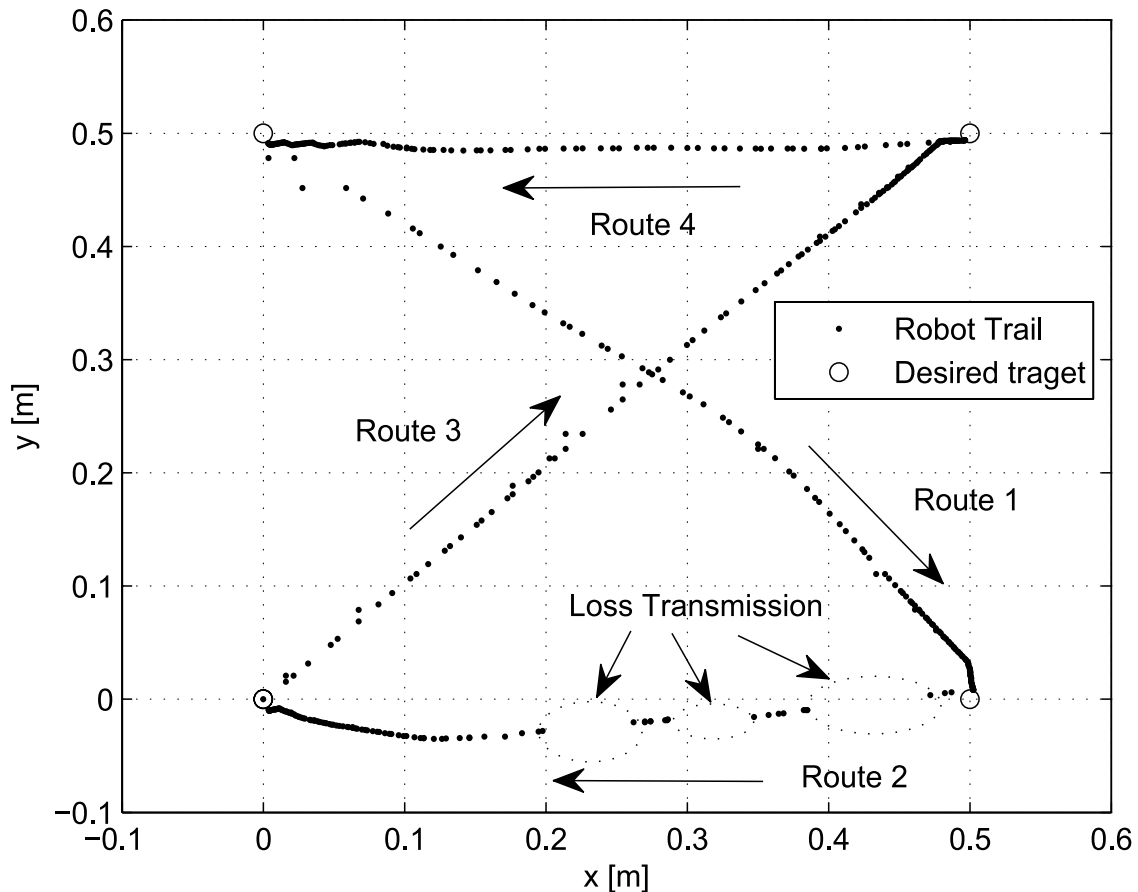Fig. 15. Experimental result: velocity v_x and v_y.

Fig. 16. Experimental result: position $x - y$ (in case of loss signal transmission).

6. Conclusion

In this chapter, an adaptive control for a class of nonlinear discrete-time systems based on multi-input fuzzy rules emulated networks (MIFREN) is introduced by the approximation with Taylor and mean value theorem. With out the need of mathematical system model, the approximation can be existed directly to construct the control law. Two MIFRENs are implemented to estimate these unknown functions obtained by the nonaffine linearization. With the main theorem, the learning algorithm for parameters inside MIFRENs guarantees the convergence of these parameters and the satisfied tracking performance. The computer simulation system demonstrates the accuracy of our mathematic proof. We already consider both operating cases for the nominal plant and the plant with some uncertainties. The bounded parameters $||\beta_1||$ and $||\beta_2||$ and the satisfied tracking performance can be presented for the both cases with the same initial setting. The experimental setup the commercial mobile robot system called Robotino® is given to demonstrate the controller performance.

7. References

[1] C.T. Lin, *Neural fuzzy systems*, Prentice-Hall, 1996.

[2] C. Treesatayapun and S. Uatrongjit, "Adaptive controller with Fuzzy rules emulated structure and its applications'," *Engineering Applications of Artificial Intelligence,* Elsevier, vol. 18, pp. 603-615, 2005

[3] C. Treesatayapun "Nonlinear Systems Identification Using Multi Input Fuzzy Rules Emulated Network," *ITC-CSCC2006 International Technical Conference on Circuits/Systems, Computers and Communications,* Chiangmai, Thailand, 10-13 July 2006

[4] E. Armanda-Bricaire, U. Kotta, and C. H. Moog, "Linearization of discrete-time systems," *SIAM J. Contr. Optim.,* vol. 34, pp. 1999-2023, 1996.

[5] G. L. Plett, "Adaptive Inverse control of linear and nonlinear system using dynamic neural networks," *IEEE Trans. Neural Netw.*, vol. 14, no. 2, pp. 360-376, Mar. 2003.

[6] H. X. Li and H. Deng, "An approximation internal model-based neural control for unknown nonlinear discrete processes," *IEEE Trans. Neural Networks*, vol. 17, no. 3, pp. 659-670, May. 2006.

[7] H. Deng and H. X. Li, "A novel neural network approximate inverse control for unknown nonlinear discrete dynamical systems," *IEEE Trans. Syst. Man Cybern. B, Cybern.*, vol. 35, no. 1, pp. 115-123, Feb. 2005.

[8] H. Deng and H. X. Li, "On the new method for the control of discrete nonlinear dynamic systems using neural networks," *IEEE Trans. Neural Netw.*, vol. 17, no. 2, pp. 526-529, Mar. 2006.

[9] J. B. D. Cabrera and K. S. Narendra, "Issues in the application of neural networks for tracking based on inverse control," *IEEE Trans. Automat. Contr.*, vol. 44, pp. 2007-2027, Nov. 1999.

[10] J. Slotine and W. Li, *Applied Nonlinear Control*. Englewood Cliffs, NJ: Prentice-Hall, 1991.

[11] L. Chen and K. Narendra, "Identification and Control of a Nonlinear Discrete-Time System Based on its Linearization: A Unified Framework," *IEEE Trans. Neural Networks*, vol. 15, no. 3, pp. 663-673, May. 2004.

[12] M. Chen and D.A. Linkens,"A hybrid neuro-fuzzy PID controllers," *Fuzzy Sets and System*, pp. 27-36, 99, 1998.

[13] P. He and S. Jagannathan, "Reinforcement Learning Neural-Network-Based Controller for Nonlinear Discrete-Time Systems With Input Constraints," *IEEE Trans. Syst., Man., Cybern.*, vol. 37, no. 2, pp. 425-436 Apr. 2007.

[14] Q. Gan and C. J. Harris, "Fuzzy local linearization and local basis function expansion in nonlinear system modeling," *IEEE Trans. Syst., Man, Cybern. B*, vol. 29, pp. 559-565, Aug. 1999.

[15] Q. M. Zhu and L. Guo, "Stable adaptive neurocontrol for nonlinear discrete-time systems," *IEEE Trans. Neural Netw.*, vol. 15, no. 3, pp. 653-662, May 2004.

[16] S. S. Ge, J. Zhang, and T. H. Lee, "Adaptive MNN control for a class of non-affine NARMAX systems with disturbances," *Syst. Control Lett.*, vol. 53, pp. 1-12, 2004.

[17] W. Rudin, *Principles of Mathematical Analysis*, New York: McGraw Hill, 1976.

[18] Z. Q.Wu and C. J. Harris, "A neurofuzzy network structure for modeling and state estimation of unknown nonlinear systems," *Int. J. Syst. Sci.*, vol. 28, pp. 335-345, 1997.
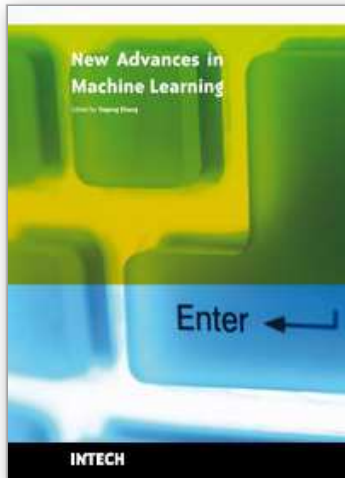
New Advances in Machine Learning

Edited by Yagang Zhang

The purpose of this book is to provide an up-to-date and systematical introduction to the principles and algorithms of machine learning. The definition of learning is broad enough to include most tasks that we commonly call "learning" tasks, as we use the word in daily life. It is also broad enough to encompass computers that improve from experience in quite straightforward ways. The book will be of interest to industrial engineers and scientists as well as academics who wish to pursue machine learning. The book is intended for both graduate and postgraduate students in fields such as computer science, cybernetics, system sciences, engineering, statistics, and social sciences, and as a reference for software professionals and practitioners. The wide scope of the book provides a good introduction to many approaches of machine learning, and it is also the source of useful bibliographical information.