A Prony Approach to Modeling of Time-Varying Fuzzy Systems

Trung T. Pham
D1central Corporation
17625 El Camino Real, Suite 200
Houston, Texas 77058

Abstract. This paper extends the Prony method to the identification of a class of fuzzy systems parameterized by a set of time-varying parameters. The Prony formulation is an optimal solution minimizing the fluctuation effects while accurately estimating the time-varying system parameters. A closed-form and efficient algorithm for adaptively calculating the time-varying parameters is derived. This solution is used to model nonlinear systems as time-varying fuzzy systems. The parameters of these fuzzy systems are adaptively calculated for each timeframe, leading to the completion of the system modeling. Numerical simulations are given to the modeling of different nonlinear systems.

1 Introduction

Nonlinear system theory usually applied some rather advanced and complicated mathematical formulations to describe the dynamical behaviors of a physical system. This mathematical representation is often used for analysis, prediction, and controller design of the actual physical system. The more accurate is the mathematical representation, the better are the analysis, prediction, and control results [1-4].

An accurate mathematical representation of a system is derived through a process known as modeling [5-7]. In this process, dynamical behaviors of well-known mechanical and electrical systems are described with differential equations derived from laws of physics. In cases where the system is not completely known, a system identification method is used. This is a numerical approach in which a nominal mathematical expression is selected. Data profiles of input and output are collected for this system and used to calculate the parameters of the selected mathematical expression under some best-fit criteria.

Fuzzy logic [8-10] has recently been developed to loosely describe a set of unknown dynamical systems to a satisfactory level. In this process, a form of fuzzy rules characterized by a set of parameters associated with a symmetric triangular-shaped membership functions would yield a set of mathematical expressions containing upper and lower bounds covering all possible systems described by these fuzzy rules. This fuzzy modeling approach allows robustness in control applications: a controller designed based on such a fuzzy model can be used to effectively control any system that fits between the upper bound and lower bound of the fuzzy model. Thus, the need
to accurately describe a specific system in order to effectively design a controller is greatly relieved.

The parameters of these fuzzy rules can be identified using numerical methods for solving an optimization problem, which minimizes the total errors between the model and the observable data set [11-13]. The observable data are collected with respect to a set of inputs to form an input-output profile. The parameters of the fuzzy model are then estimated using this input-output profile. In general, these parameters are estimated using the criterion of minimizing the squared-norm error, i.e., the sum of the errors between the actual data and the upper bound and the errors between the actual data and the lower bound, or the errors between the actual data and the average between the two bounds.

The Prony formulation [14-16] is a technique of estimating the parameters of a system based on two sets of criteria, one based on a specific behavior of a system, and the other based on the resulting error between the model and the observable data. These parameters are partitioned into two independent groups, with one group specifically corresponds to a certain behavior of a system. This specific group of parameters is estimated in a way of optimizing that particular behavior. Then, the remaining parameters are estimated to minimize the total error calculated from the available observable data. The Prony method is often used to estimate the parameters of a time-varying system, where sensitivity to shifting poles and zeros is required to be robust.

2 Fuzzy Logic Modeling

2.1 Single-Input Single-Output (SISO) Systems

A set of fuzzy logic can be used to describe the dynamical behavior of a system. In this approach, a single-input single-output (SISO) system is modeled as follows [9]:

\[ R: \text{if } x \in S_n \text{ and } u \in S_m \text{ then } x \in S_{n+bn}, \]

where \( x \) is the state variable, \( u \) the input, and the transition of the system. The fuzzy sets \( S_n, S_m \) are defined over intervals \([(-1)\sigma, (n+1)\sigma]\) and \([(-m-1)\sigma, (m+1)\sigma]\) on a partition of the real line, with the corresponding fuzzy membership function \( \mu_{S_n}(\cdot) \) and \( \mu_{S_m}(\cdot) \), for some chosen \( \sigma > 0 \). In this paper, a specific membership function of the triangular form is used:

\[
\mu_{S_n}(x) = \begin{cases} 
\frac{x}{\sigma} - (n-1) & (n-1)\sigma \leq x < n\sigma, \\
(n+1) - \frac{x}{\sigma} & n\sigma \leq x < (n+1)\sigma, \\
0 & \text{else.}
\end{cases}
\]

This membership function has the unique property of yielding an upper bound and lower bound for the fuzzy rule (1). The upper bound is given as
\[ \dot{\xi}(t) = a \xi(t) + bu(t) - \rho, \]  

(3a)

and the lower bound is given as

\[ \dot{\zeta}(t) = a \zeta(t) + bu(t) + \rho, \]  

(3b)

where

\[ \rho = (|a| + |b|) \sigma. \]  

(3c)

Any system with state variable \( x(t) \), when given the control input \( u(t) \), that is bounded from above by \( \dot{\xi}(t) \) and from below by \( \dot{\zeta}(t) \), i.e.,

\[ \dot{\xi}(t) \leq x(t) \leq \dot{\zeta}(t), \]

is a member of the fuzzy rule (1).

The case of SISO is presented here as the foundation for setting up and deriving the Prony formulation. It is also used as the mathematical base to extend to the general case of multiple-input multiple-output (MIMO) systems.

For a time-varying system, the parameters \( a \) and \( b \) are assumed to be non-constant, i.e., they become dependent on time \( t \). The fuzzy rule (1) and the two bounds in (3) can be re-written, with the constants \( a \) and \( b \) being time-dependent, i.e., \( a(t) \) and \( b(t) \) instead of \( a \) and \( b \). These parameters \( a(t) \) and \( b(t) \) will be estimated as functions of time \( t \). The time-varying system linear model can also be used to approximate some nonlinear systems in applications.

### 2.2 Multiple-Input Multiple-Output (MIMO) Systems

A set of fuzzy rules, similar to that of the SISO case, can also be used to describe the dynamical behavior of a multiple-input multiple-output (MIMO) system. In this case, an MIMO system is modeled as follows [9]:

\[ R^i: \text{if} \ x_1 \in S_{n_1} \land \ldots \land x_K \in S_{n_K} \land u_1 \in S_{m_1} \land \ldots \land u_J \in S_{m_J}, \text{then} \ \dot{x}_i \in S_{a_{i,1}u_1 + \ldots + a_{i,J}u_J + b_{i,1}u_1 + \ldots + b_{i,J}u_J}, \]  

(4)

for \( i = 1, 2, \ldots, K \); where \( K \) is the number of state variables, and \( J \) the number of control inputs. For a triangular membership function given in (2), the fuzzy rules in (4) yield the upper bound \( \zeta \) and lower bound \( \xi \) as follows:

\[ \dot{\zeta}_i(t) = \sum_{k=1}^{K} a_{i,k} \dot{\xi}_k(t) + \sum_{j=1}^{J} b_{i,j} u_j(t) - \left[ \sum_{i=1}^{K} \sum_{k=1}^{K} a_{i,k} \right] + \left[ \sum_{i=1}^{K} \sum_{j=1}^{J} b_{i,j} \right] \sigma, \]  

(5a)
A Prony Approach to Modeling of Time-Varying

\[ \hat{z}_i(t) = \sum_{k=1}^{K} a_{i,k} \hat{z}_{k}(t) + \sum_{j=1}^{J} b_{i,j} u_j(t) + \left[ \sum_{j=1}^{J} \sum_{k=1}^{K} |a_{i,j}| + \sum_{i=1}^{I} \sum_{j=1}^{J} |b_{i,j}| \right] \sigma. \quad (5b) \]

These two bounds effectively contain any MIMO system described by the fuzzy rules in (4). This model, when extended to the time-varying case, will have the parameters \(a_{i,k}(t)\) and \(b_{i,j}(t)\) showing their dependence on time \(t\).

3 Prony Approach

In this section, the two-step Prony method is used to estimate the parameters of the fuzzy rule(s) describing the dynamical behavior of a modeled system. First, the problem is formulated for an SISO system for illustrative purpose. Then, the general MIMO case is handled based on the result of the SISO case.

The Prony method partitions the identification of the parameters into two parts. First, a set of parameters are estimated with the assumption that the remaining other parameters are the same as in the previous timeframe. The criteria used in the estimation of these parameters can be selected and formulated accordingly. In this paper, two sets of criteria are examined: the norm of the matrix \(A\) containing the constants \(a_{i,j}\) is minimized for the bounded-input bounded-output stability, and only the selected elements of the matrix \(A\) containing the constants \(a_{i,j}\) are minimized so as to formulate the system to have pre-specific poles in the transfer function.

3.1 Single-Input-Single-Output (SISO) Systems

For the SISO system described by the fuzzy rules (1), assume that two profiles \(X\) and \(U\) of observable data for the state variable \(x(t)\) and the control input \(u(t)\) are available for a duration of time, \(t \in [t_i,t_f]\):

\[ X = \{ x_0, x_1, x_2, ..., x_N \} \]  \hspace{1cm} (6a)

\[ U = \{ u_0, u_1, u_2, ..., u_N \} \]  \hspace{1cm} (6b)

**Problem Formulation:** Assuming that a time-varying fuzzy model is used to represent this system, as follows:

\[ R: \quad \text{if } x(t) \in S_n \text{ and } u(t) \in S_m \text{ then } x(t) \in S_{n(u+b)+m}. \]

The objective of solving the system identification problem is to find the parameters \(a(t)\) and \(b(t)\) that best fit the profiles \(X\) and \(U\) to the fuzzy model above.

The Prony method will estimate the parameter \(a(t)\) first under one criterion and with the assumption that \(b(t)\) is the same as that used in the previous cycle. Here, the parameter \(a(t)\) is estimated with the criterion that the norm of \(a(t)\) is minimized for
achieving the bounded-input bounded-output stability. Then, it will estimate the parameter \( b(t) \) under the criterion of minimizing the total error between the model and the actual data, using the parameter \( a(t) \) just obtained.

The norm of \( a(t) \) is minimized in the optimization below:

\[
\min_{a(t)} \| a(t) \|^2, \quad (7)
\]

subject to:

\[
x(t_{n+1}) \leq x(t_n) + \Delta t [ a(t_n)x(t_n) + b(t_n)u(t_n) + \rho ], \quad (8a)
\]

\[
x(t_{n+1}) \geq x(t_n) + \Delta t [ a(t_n)x(t_n) + b(t_n)u(t_n) - \rho ]. \quad (8b)
\]

where \( \rho \) was defined earlier in (3c) as dependent on the width of the fuzzy sets defined for each indexed interval \( S_k \). To simplify the problem, assume that the constant \( \rho \) is calculated based on \( a(t_{n-1}) \) and \( b(t_{n-1}) \), and the parameter \( b(t_n) \) is the same as \( b(t_{n-1}) \) in this calculation.

The second part of the Prony approach is to estimate the remaining parameters. In this SISO case, the only remaining parameter is \( b(t_n) \). The least squared-norm error approach is:

\[
\min_{b(t_n)} \| x_{n+1} - z_{n+1} \|^2, \quad (9)
\]

subject to:

\[
z(t_{n+1}) \leq x(t_n) + \Delta t [ \hat{a}(t_n)x(t_n) + b(t_n)u(t_n) + \rho ], \quad (10a)
\]

\[
z(t_{n+1}) \geq x(t_n) + \Delta t [ \hat{a}(t_n)x(t_n) + b(t_n)u(t_n) - \rho ]. \quad (10b)
\]

Solution: Since the optimization problem in (7) and (8) has the form of a quadratic programming problem [17-18], the existence of its solution can be easily established. Furthermore, this particular quadratic programming problem yields the following closed-form solution:

\[
\hat{a}(t_n) = \begin{cases} 
\kappa \cdot x(t_n) & \text{if } \kappa \leq 0, \\
\lambda \cdot x(t_n) & \text{if } \lambda \geq 0, \\
0 & \text{else,}
\end{cases} \quad (11a)
\]

where the constants \( \kappa \) and \( \lambda \) are defined as:

\[
\kappa = \frac{x(t_{n+1}) - x(t_n)}{\Delta t} - b(t_{n-1})u(t_n) + \rho, \quad (11b)
\]

\[
\lambda = \frac{x(t_{n+1}) - x(t_n)}{\Delta t} - b(t_{n-1})u(t_n) - \rho. \quad (11c)
\]
The solution in (11) is unique. Notice that since the parameter \( b(t_n) \) is not yet calculated here, the parameter \( b(t_{n-1}) \) is used instead, as mentioned above.

It can be seen that the constrained optimization problem in (10) and (11) is in the form of a quadratic programming problem. This guarantees the existence of a solution. This solution can be obtained in closed-form as follows:

\[
\hat{b}(t_n) = \begin{cases} 
\frac{k_b}{u(t_n)} & \text{if } k_b \leq 0, \\
\frac{\lambda_b}{u(t_n)} & \text{if } \lambda_b \geq 0, \\
0 & \text{else},
\end{cases}
\]  

\[
k_b = \left[ \frac{x(t_{n+1}) - x(t_n)}{\Delta t} \right] - \hat{a}(t_n)x(t_n) + \rho,
\]

\[
\lambda_b = \left[ \frac{x(t_{n+1}) - x(t_n)}{\Delta t} \right] - \hat{a}(t_n)x(t_n) - \rho,
\]

which can be shown to be unique.

Since the SISO model only has one element in the matrix containing \( a(t) \), it is irrelevant to setup the Prony approach for minimizing selected elements to achieve the pole assignment. This case is deferred to the later section of MIMO system.

### 3.2 Multiple-Input Multiple-Output (MIMO) Systems

For an MIMO system described by the fuzzy rule (4), assume that two profiles \( X \) and \( U \) of observable data for the state variable \( x(t) \) and the control input \( u(t) \) are available for a duration of time \( t \in [t_i,t_f] \):

\[
X = \{ x_0, x_1, x_2, \ldots, x_N \} \tag{13a}
\]

\[
U = \{ u_0, u_1, u_2, \ldots, u_N \} \tag{13b}
\]

**Problem Formulation:** For the Prony setting, to minimize the norm of the matrix \( A \) containing the parameters \( a_{ij} \) in the fuzzy rules in (4), it is set up as follows:

\[
\min_{A_{ij}} \sum_{p=1}^{N} \sum_{q=1}^{N} |a_{p,q}(t_n)|^2, \tag{14}
\]

s.t.

\[
| x(t_{n+1}) - x(t_n) - \Delta t \left[ \sum_{j=1}^{N} a_{1,j}(t_n)x_j(t_n) + \sum_{j=1}^{M} b_{i,j}(t_n)u_j(t_n) \right] | \leq \Delta \rho,
\]

for \( i = 1, 2, \ldots, N \). \tag{15}
Since the objective function in (14) is the summation of independent terms, this problem can be decoupled into $N$ simpler independent quadratic programming problems, each has the form:

$$\min_{a_{ij}} \sum_{p=1}^{N} |a_{ij}(t_p)|^2,$$  \hspace{1cm} (16)

subject to

$$|x(t_{n+1}) - x(t_n) - \Delta t (\sum_{j=1}^{N} a_{ij}(t_n)x_j(t_n) + \sum_{j=1}^{M} b_{pq}(t_n)u_j(t_n))| \leq \Delta t \rho,$$  \hspace{1cm} (17)

where $i = 1, 2, \ldots, N$.

Sometimes, it is obvious that the model is set up to have a specific canonical form where the constants $a_{ij}$ and $b_{pq}$ follow the following rules:

$$a_{ij} = \begin{cases} 1 & \text{if } j < N \text{ and } j = i + 1, \\ 0 & \text{if } j < N \text{ and } j \neq i + 1, \\ a_{iN} & \text{if } j = N, \end{cases}$$  \hspace{1cm} (18)

$$b_{pq} = \begin{cases} 0 & \text{if } q < N, \\ b_{pN} & \text{else}, \end{cases}$$  \hspace{1cm} (19)

and thus the problem in (16) and (17) becomes applicable only for the case of $i = N$. All other constants, $a_{ij}$ for $j < N$, are assumed to be either 1 or 0 according to (18). This model allows the identification problem to be set up under the criterion of controlling the placement of the poles in the transfer function, similarly to the way the Prony method is set up in the frequency domain.

The second part of estimating the parameters in the matrix $B$ can be set up as an optimization problem minimizing the total error between the model and the actual system, as follows:

$$\min_{b(t_n)} |x(t_{n+1}) - z(t_{n+1})|^2,$$  \hspace{1cm} (20)

subject to

$$z(t_{n+1}) \leq x(t_n) + \Delta t [\sum_{j=1}^{N} \tilde{a}_{ij}(t_n)x_j(t_n) + \sum_{j=1}^{M} \tilde{b}_{ij}(t_n)u_j(t_n) + \rho],$$  \hspace{1cm} (21)

$$z(t_{n+1}) \geq x(t_n) + \Delta t [\sum_{j=1}^{N} \tilde{a}_{ij}(t_n)x_j(t_n) + \sum_{j=1}^{M} \tilde{b}_{ij}(t_n)u_j(t_n) - \rho].$$  \hspace{1cm} (22)
Solution: The solution to the quadratic programming problem in (16) and (17) exists and can be uniquely derived in a closed form, as follows:

\[
\hat{a}_{i,j}(t_n) = \begin{cases} 
\frac{x_{\text{a},i}(t_n)K_{\text{a},j}}{|x(t_n)|} & \text{if } \kappa_{a,j} \leq 0, \\
\frac{x_{\text{a},i}W_{\text{a},j}}{|x(t_n)|} & \text{if } \kappa_{a,j} \geq 0, \\
0 & \text{else,}
\end{cases}
\]  

(23)

\[
\kappa_{a,j} = \left[ \frac{x_{\text{a},i}(t_{n+1}) - x_{\text{a},i}(t_n)}{\Delta t} \right] - \sum_{j=1}^{M} h_{a,j}(t_{n-1})u_j(t_n) + \rho, 
\]  

(24)

\[
\hat{b}_{j,j}(t_n) = \begin{cases} 
\frac{x_{\text{b},i}(t_n)K_{\text{b},j}}{|x(t_n)|} & \text{if } \kappa_{b,j} \leq 0, \\
\frac{x_{\text{b},i}W_{\text{b},j}}{|x(t_n)|} & \text{if } \kappa_{b,j} \geq 0, \\
0 & \text{else,}
\end{cases}
\]  

(25)

\[
\kappa_{b,j} = \left[ \frac{x_{\text{b},i}(t_{n+1}) - x_{\text{b},i}(t_n)}{\Delta t} \right] - \sum_{j=1}^{N} \hat{a}_{i,j}(t_n)x_j(t_n) + \rho. 
\]  

(26)

\[
\hat{b}_{j,j}(t_n) = \begin{cases} 
\frac{x_{\text{b},i}(t_n)K_{\text{b},j}}{|x(t_n)|} & \text{if } \kappa_{b,j} \leq 0, \\
\frac{x_{\text{b},i}W_{\text{b},j}}{|x(t_n)|} & \text{if } \kappa_{b,j} \geq 0, \\
0 & \text{else,}
\end{cases}
\]  

(27)

4 Numerical Simulations

4.1 Example 1: A Time-Invariant SISO System

In this example, a simple time-invariant SISO model is used as a real system to be modeled. The profiles \(X\) and \(U\) are calculated from this mathematical model to simulate the real system. The parameters \(a(t)\) and \(b(t)\) of the fuzzy model are calculated using the Prony method and data profiles \(X\) and \(U\).
Let the simulated system (to be modeled) be
\[ \dot{x}(t) = -2x(t) + 3u(t). \]

Let \( u(t) \) be a step response, and the computational period \( \Delta t = 20 \text{ msec} \), the acceptable time duration for many real-time simulations. Let the initial state \( x(t_0) = 10 \). Then profiles \( X \) and \( U \) are calculated as follows:

\[
U = \{ 1, 1, 1, \ldots, 1 \},
\]

\[
X = \{ x_0, x_1, x_2, \ldots, x_{N-1} \},
\]

where \( x_n = x(t_0 + n\Delta t) \) and is calculated according to the iterative formula:
\[
x_n = x_{n-1} + (a_{n-1}x_{n-1} + b_{n-1}u_{n-1})\Delta t.
\]

In this example, the parameters \( a_n = -2 \), and \( b_n = 3 \) for all integers \( n = 0, \ldots, N-1 \). The simulated profile \( X \) is plotted against time \( t \) in Figure 1 for \( t \in \{ t_0, t_1, t_2, \ldots, t_{250} \} \) with the initial time \( t_0 = 0 \) and the interval \( \Delta t = 0.02 \text{ sec} \).

The fuzzy model is established for each fuzzy set with the width \( \sigma = 0.05 \). Figure 1 shows the simulated system when a step response is used as the input. Figure 2 shows the output of the model when the parameters \( a \) and \( b \) are estimated using Equations (11) and (12). Figure 3 shows the error between the simulated system and the estimated model. Figure 4 shows the estimated value of \( a(t) \). Note that the error (Fig. 3) is significantly reduced as time moves forward, confirming the intuition that as more data are collected, the model is more accurate. However, the estimated constant \( a \) does not converge to the true value \(-2.0\) in the simulated model. This phenomenon can be partially explained as due to the gap reserved for the constant \( p \) in Equations (11b) and (11c), and partially explained by the way the constants are set to zeros in Equations (11a) and (12a).

### 4.2 Example 2: A Time-Varying SISO System

Let the simulated system (to be modeled) be
\[ \dot{x}(t) = -\sin(t-1)x(t) + \cos(t)u(t). \]

Let \( u(t) \) be a step response, and the computational period \( \Delta t = 20 \text{ msec} \). Let the initial state \( x(t_0) = 10 \). Then profiles \( X \) and \( U \) are calculated similarly to that of Example 1. The fuzzy model is the same as that of Example 1.

Figure 5 shows the simulated system when a step response is used as the input. Figure 6 shows the output of the model when the parameters \( a \) and \( b \) are estimated using Equations (11) and (12), respectively. Figure 7 shows the error between the simulated system and the estimated model. Figure 8 shows the estimated value of \( a(t) \).
A Prony Approach to Modeling of Time-Varying

Figure 1. Output $x(t)$ of simulated system

$$x(t) = -2x(t) + 3u(t).$$

Figure 5. Output $x(t)$ of simulated system

$$x(t) = -\sin(t-1)x(t) + \cos(t)\sin(t).$$

Figure 2. Output $x(t)$ of modeled system in Example 1.

Figure 6. Output $x(t)$ of modeled system in Example 2.

Figure 3. Error of modeled system in Example 1.

Figure 7. Error of modeled system in Example 2.

Figure 4. Estimated time-varying constant $\hat{a}$ in Example 1.

Figure 8. Estimated time-varying constant $\hat{b}$ in Example 2.
Note that the error, while in the comfortable range (percentage-wise when compared to the actual data), is significantly more than that in the previous example. This can be expected because the linear model is used to estimate a sinusoidal behavior. However, the error never converges to zero due to the fact that the simulated system can only be estimated accurately for a duration of time before the system moves away from the reference point of linearization. The estimated parameter $a(t)$, shown in Figure 8, reflects this oscillation.

### 4.3 Example 3: A Time-Varying MIMO System

Let the simulated system (to be modeled) be

\[
\begin{align*}
\dot{x}_1(t) &= -\sin(t-1)\ x_1(t) + \cos(t)\ x_2(t), \\
\dot{x}_2(t) &= -x_1(t) + \cos(t)\ x_2(t) + u(t).
\end{align*}
\]

Let $u(t)$ be a step response, and the computational period $\Delta t = 20$ msec. Let the initial state $x_1(t_0) = 5$ and $x_2(t_0) = 5$. Then profiles $X$ and $U$ are calculated similar to that of Example 1. The fuzzy model is the same as that of Example 1.

Figure 9 shows the output $x_2(t)$ of the simulated system when a step response is used as the input. Figure 10 shows the output $x_2(t)$ of the model when the parameters $a_{1,1}$, $a_{1,2}$, $a_{2,1}$, $a_{2,2}$, $b_{1,1}$, and $b_{2,1}$ are estimated using Equations (23) and (25). Figure 11
shows the error between the output of the simulated system and the output of the estimated model. Figure 12 shows the estimated value of $a_{2,2}(t)$.

Again, the overall model matches with the simulated system. However, the error shows that it is not converging nicely, displaying a larger norm than that of the linear case in Example 1. This larger error can be explained similarly to that of Example 2: the difference between a linear model and a sinusoidal system is expected to be higher. The constant $a_{2,2}$ shows oscillation with the same frequency as the output $x_2(t)$ of the simulated system. This is what one would expect intuitively in a linearized model. However, the error function seems to behave in a disturbing trend of growing. This may due to the instability nature of the system itself, magnifying the error as the magnitude of the output gets larger.

5 Conclusion

It has been shown that a specific fuzzy representation based on indexed fuzzy sets and triangular membership functions can be used to model a class of dynamical systems. These fuzzy representations are characterized by a set of parameters that can be estimated with a given set of observable data from the actual system. The Prony approach is used to partition the parameter identification problem into two steps: the first step to achieve the BIBO stability, and the second step to optimize the least squared-norm error. Numerical simulations have demonstrated the workability and validated the results derived in closed-form solutions.

References


Biography

▲ Name: Trung T. Pham
Address: Dlcentral Corporation, 17625 El Camino Real, Suite 200, Houston, Texas 77058
Education & Work experience: BS, MS, and PhD in electrical and computer engineering (Rice University, Houston, Texas), MBA (University of Houston – Clear Lake). Chief Technology Officer at Dlcentral, responsible for R&D in machine translation, intelligent systems, and automated processes. ISA Senior Member and Director Elect of Robotics and Expert Systems Division.
Tel: +1-281-701-6050
E-mail: t_robo@alumni.rice.edu

©GESTS-Oct.2005