

Article

A Fuzzy Difference Equation Matrix Model for the Control of Multivariable Nonlinear Systems

Basil Mohammed Al-Hadithi ^{1,2,*} , Javier Blanco Rico ³  and Agustín Jiménez ¹

¹ Intelligent Control Group, Centre for Automation and Robotics UPM—CSIC, Universidad Politécnica de Madrid, 28006 Madrid, Spain; agustin.jimenez@upm.es

² Department of Electrical, Electronics, Control Engineering and Applied Physics, School of Industrial Design and Engineering, Universidad Politécnica de Madrid, 28012 Madrid, Spain

³ Escuela Técnica Superior de Ingenieros Industriales, Universidad Politécnica de Madrid, 28006 Madrid, Spain; j.brico@alumnos.upm.es

* Correspondence: basil.alhadithi@upm.es

Abstract

This paper proposes the Fuzzy Difference Equation Matrix Model (FDEMM), a novel predictive control algorithm designed for nonlinear multivariable systems. Standard Dynamic Matrix Control (DMC) often struggles with computational load and nonlinearities. FDEMM addresses this by integrating the Difference Equation Matrix Model (DEMM) with a generalized Takagi-Sugeno (T-S) fuzzy framework, utilizing a parameter-weighting scheme to handle overlapping membership functions. The method is validated on two distinct nonlinear systems: a binary distillation column and a delayed thermal mixing tank. Results demonstrate FDEMM's ability to control complex systems achieving the desired output even in the presence of disturbances and noise. The proposed strategy offers a computationally efficient alternative for real-time control of complex nonlinear processes.

Keywords: predictive control; T-S fuzzy model; nonlinear systems, multivariable systems; thermal mixing tank; distillation column

1. Introduction

A new predictive algorithm, the Fuzzy Difference Equation Matrix Model (FDEMM), is presented in the context of Model Predictive Control (MPC). In previous work, we demonstrated that the Difference Equation Matrix Model (DEMM) offers reduced computational load and faster execution than the original Dynamic Matrix Control (DMC), making it particularly suitable for real-time applications. In this work, we extend DEMM by integrating it with the Takagi-Sugeno (T-S) fuzzy modeling approach to handle nonlinear and complex multivariable systems.

One of the main limitations of standard T-S modeling lies in its inability to work with overlapping pairs of Membership Functions (MFs), a common situation. To address this, we propose a generalized T-S modeling structure that accommodates overlapping MFs. This is achieved through a parameter weighting methodology designed to optimize the performance index used in the MPC formulation. We also introduce a robust fuzzy controller based on a Linear Quadratic Regulator (LQR), leveraging the flexibility of fuzzy systems and the stability guarantees of LQR. The performance and robustness of the proposed framework are demonstrated through two practical case studies: a nonlinear multivariable distillation column and a delayed thermal mixing tank.



Academic Editors: Andrey Miroshnichenko and Ephraim Suhir

Received: 3 December 2025

Revised: 24 January 2026

Accepted: 19 February 2026

Published: 20 February 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

MPC has become a foundational strategy for handling complex control problems in modern industry. Its ability to predict future plant behavior and optimize control actions over a finite prediction horizon makes it particularly effective for multi-variable and constraint-heavy systems [1]. At each control step, MPC solves an optimization problem using a model of the plant to determine the best possible control inputs.

Within the family of MPC algorithms, DMC stands out for its ease of implementation, real-time feasibility, and broad industrial adoption. DMC is especially powerful for Multiple-Input–Multiple-Output (MIMO) systems, as it naturally accommodates input/output constraints and unusual dynamic behavior. However, DMC's main drawback lies in its computational intensity, particularly when used in applications involving adaptive control, online system identification, or nonlinear dynamics. This limitation has motivated a range of adaptations aimed at reducing computational demands without sacrificing control performance.

MPC has seen wide application in fields ranging from energy management [2,3], aerospace systems [4] to autonomous vehicles control [5,6]. For instance, in the building sector, Heating, Ventilation, and Air Conditioning (HVAC) systems represent a common application area where MPC has demonstrated significant energy savings. These systems are governed by dynamic variables such as occupancy levels, temperature, weather conditions, and solar radiation [2]. MPC's ability to plan future actions based on forecasted disturbances enhances performance under such uncertainty.

In the aerospace domain, Ref. [4] presents an MPC scheme combined with event-triggered control for high-precision spacecraft maneuvers. Rather than recomputing control actions at fixed intervals, their method triggers control updates only when prediction errors exceed certain thresholds, thereby reducing computational burden while maintaining accuracy.

Even when accurate plant models are not available, MPC remains useful as a reference trajectory generator or as a component in hybrid control systems. In [5], Reinforcement Learning (RL) is used in tandem with MPC to control autonomous vehicles motion. RL techniques are employed both to generate fuzzy approximations of the plant and to enhance the MPC optimization process, demonstrating improved adaptability in uncertain environments.

Battery management is another area where MPC and DMC are used. In [3], DMC is employed to enhance State of Charge (SoC) estimation in battery systems by continuously updating the model based on real-time voltage and current measurements. As batteries age and deviate from initial conditions, DMC helps recalibrate predictions to maintain estimation accuracy.

Similarly, Ref. [6] applies DMC in the coordinated control of quadcopters, which involves both collision avoidance in 3D space and consensus in shared variables like yaw or horizontal positioning. The study highlights DMC's ability to operate effectively in high-dimensional, dynamic environments.

The effectiveness of MPC-based control is highly dependent on the accuracy of the process model used. In many cases, practitioners rely on linearized models, derived from physical laws and valid around a specific operating point. However, such models often fail to capture nonlinearities, dead times, or aging effects, especially in complex systems.

As a result, researchers have sought alternative modeling techniques capable of providing accurate representations of nonlinear systems, even when system dynamics are only partially known. Among these, fuzzy modeling has gained popularity due to its capacity to approximate nonlinear behavior using a collection of simple linear models, combined via simple MFs.

The T-S fuzzy model [7] is one of the most studied fuzzy modeling frameworks. It allows complex nonlinear systems to be represented as a weighted combination of linear subsystems, where the weights are determined by the degree of truth in fuzzy rules. This

modeling approach is particularly attractive when classical identification techniques are infeasible or when partial knowledge of the system is available.

Several improvements to standard T-S models have been proposed in recent literature: in [8] a self-adjusting real-time fuzzy model is used to control a robotic arm under unknown dynamics, in [9] unknown MFs are used to construct a fuzzy model for unmanned marine vehicles, addressing complex disturbances in ocean navigation, and in [10] fractional-order uncertain T-S fuzzy model is developed.

Once constructed, fuzzy models can be integrated with advanced control strategies. For example, ref. [11] combines H_∞ control with piecewise Lyapunov functions for finite-time stability guarantees.

In chemical process industries, for example in the control of distillation columns, MPC has shown clear advantages over traditional control strategies. Distillation or other processes that involve mixtures and maintaining temperature and concentration conditions are nonlinear and often involve dead times, which complicates linear model-based control [12]: in [13] machine learning–MPC is shown to improve energy efficiency and reduce error on a distillation column, and in [14], a reinforcement learning temperature control model is used for the MPC responsible of a rubber mixing process.

In [15], the inadequacy of PID for distillation control is discussed, and DMC is found to outperform both PID and fuzzy controllers. However, in our work, fuzzy logic is not used as a competing controller but as a modeling tool to support and enhance MPC performance.

A comparable approach is found in [16], where the system model for a binary distillation column is generated using a Differential Extreme Learning Machine. Their Adaptive Distributed MPC outperforms other MPC-based alternatives. In contrast, our method uses fuzzy modeling to achieve similar adaptability and accuracy with a more interpretable model structure.

Thermal mixing tanks and fluid storage systems are also prime examples of nonlinear systems where MPC excels. When hot and cold liquids mix, the resulting temperature profile depends on input timing, flow rate, and stratification effects [17]. These complexities make conventional control methods less effective: in [18], water storage is modeled as part of smart microgrid optimization, involving nonlinear pump scheduling and daily usage forecasts, and in [19], thermal storage is optimized for smart home energy systems, requiring accurate models of water temperature evolution over time.

In [20], an Adaptive Fuzzy PID is shown to outperform conventional PID in controlling a two-input thermal tank, reducing overshoot and improving settling time. Our own case study extends this by using fuzzy logic for both plant modeling and control, enabling greater adaptability and performance.

In systems where control actions must be computed before discrete or reactive events, such as in compression ignition engines, MPC provides a structured way to plan actions under uncertainty. In [21], MPC is applied to an engine where outputs (e.g., ignition power, emissions, and noise) are only observable after the reaction occurs. Neural networks are used to approximate the system dynamics, enabling predictive control in this challenging scenario.

Building upon the foundational work in predictive control and fuzzy modeling, the primary objective of this study is to develop and validate the FDEMM as a robust solution for multivariable nonlinear processes. While the traditional DMC [1,2] remains a standard for linear systems, its extension to nonlinear domains often suffers from high computational costs. The proposed FDEMM uses the DEMM [22] as its structural prototype, which has been proven more efficient than DMC in linear applications, and generalizes it through a T-S fuzzy framework [13,14]. Unlike other T-S based MPC approaches that rely on complex online optimization or LMI-based stability conditions [15], the FDEMM provides an ana-

lytical control law that maintains computational simplicity while offering the adaptability required for nonlinear dynamics. The efficacy of this approach is demonstrated through its application to a binary distillation column and a thermal mixing tank, highlighting its performance in both high-coupling and high-delay scenarios.

The remainder of the paper is structured as follows. Section 2 introduces the generalized T-S modeling approach. Section 3 revisits the DEMM algorithm. Section 4 presents the proposed FDEMM method. In Section 5, two case studies (a distillation column and a thermal mixing tank) are used to demonstrate the method's effectiveness. Section 6 concludes with a discussion of findings and suggestions for future research.

2. Estimation of T-S Model

Both the system and the FDEMM are described using a T-S fuzzy model [7]. The T-S model estimates the parameters of a nonlinear system by minimizing a quadratic performance index. It is based on approximating the system behavior through a set of local linear models, typically expressed in the form

$$\begin{aligned} f: \mathbb{R}^n &\longrightarrow \mathbb{R} \\ y &= f(x_1, x_2, \dots, x_n) \end{aligned} \quad (1)$$

Each IF–THEN rule $S^{(i_1 \dots i_n)}$ for an n^{th} -order system is defined as:

$$\begin{aligned} S^{(i_1 \dots i_n)}: & \text{if } x_1 \text{ is } M_1^{i_1} \text{ and } x_2 \text{ is } M_2^{i_2} \text{ and } \dots x_n \text{ is } M_n^{i_n} \\ & \text{then } \hat{y} = p_0^{(i_1 \dots i_n)} + p_1^{(i_1 \dots i_n)} x_1 + p_2^{(i_1 \dots i_n)} x_2 + \dots + p_n^{(i_1 \dots i_n)} x_n \end{aligned} \quad (2)$$

The fuzzy estimation of the system output is then given by

$$\hat{y} = \frac{\sum_{i_1=1}^{r_1} \dots \sum_{i_n=1}^{r_n} \omega^{(i_1 \dots i_n)}(\mathbf{x}) \left[p_0^{(i_1 \dots i_n)} + p_1^{(i_1 \dots i_n)} x_1 + \dots + p_n^{(i_1 \dots i_n)} x_n \right]}{\sum_{i_1=1}^{r_1} \dots \sum_{i_n=1}^{r_n} \omega^{(i_1 \dots i_n)}(\mathbf{x})} \quad (3)$$

where $(r_1 \dots r_n)$ is the number of fuzzy sets for variable $(x_1 \dots x_n)$.

$$\omega^{(i_1 \dots i_n)}(\mathbf{x}) = \mu_{1i_1}(x_1) \mu_{2i_2}(x_2) \dots \mu_{ni_n}(x_n) \quad (4)$$

Here $\mu_{ji}(x_j)$ denotes the MF associated with the fuzzy set M_j^i .

Let m be a set of input/output samples $\{x_{1k}, x_{2k}, \dots, x_{nk}, y_k\}$. The parameters of the fuzzy system can be estimated by minimizing the following quadratic performance index:

$$J = \sum_{k=1}^m (y_k - \hat{y}_k)^2 = \|Y - XP\|^2 \quad (5)$$

where

$$\begin{aligned} Y &= \begin{bmatrix} y_1 & y_2 & \dots & y_m \end{bmatrix}^T \\ P &= \begin{bmatrix} p_0^{(1 \dots 1)} & p_1^{(1 \dots 1)} & p_2^{(1 \dots 1)} & \dots & p_n^{(1 \dots 1)} & \dots & p_0^{(r_1 \dots r_n)} & \dots & p_n^{(r_1 \dots r_n)} \end{bmatrix}^T \\ X &= \begin{bmatrix} \beta_1^{(1 \dots 1)} & \beta_1^{(1 \dots 1)} x_{11} & \dots & \beta_1^{(1 \dots 1)} x_{n1} & \dots & \beta_1^{(r_1 \dots r_n)} & \dots & \beta_1^{(r_1 \dots r_n)} x_{n1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \beta_m^{(1 \dots 1)} & \beta_m^{(1 \dots 1)} x_{1m} & \dots & \beta_m^{(1 \dots 1)} x_{nm} & \dots & \beta_m^{(r_1 \dots r_n)} & \dots & \beta_m^{(r_1 \dots r_n)} x_{nm} \end{bmatrix} \end{aligned}$$

and

$$\beta_k^{(i_1 \dots i_n)} = \frac{\omega^{(i_1 \dots i_n)}(\mathbf{x}_k)}{\sum_{j_1=1}^{r_1} \dots \sum_{j_n=1}^{r_n} \omega^{(i_1 \dots i_n)}(\mathbf{x}_k)} \quad (6)$$

If X is a full-rank matrix, the parameters of the fuzzy system can be computed by minimizing the following performance index as follows:

$$\begin{aligned} J &= \|Y - XP\|^2 = (Y - XP)^T(Y - XP) \\ \nabla J &= X^T(Y - XP) = X^TY - X^TXP = 0 \\ P &= (X^TX)^{-1}X^TY \end{aligned} \quad (7)$$

The method developed in [7] presents significant limitations, as it cannot be applied in the most common case where the MFs are defined as shown in Figure 1.

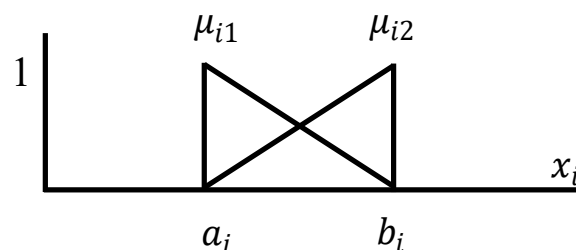


Figure 1. Most common case of MFs of the fuzzy system.

The MFs $\mu_{i1}(x_i) = \frac{b_i - x_i}{b_i - a_i}$ and $\mu_{i2}(x_i) = \frac{x_i - a_i}{b_i - a_i}$ are defined in an interval $[a_i, b_i]$, which must satisfy the following condition:

$$\begin{aligned} \mu_{i1}(a_i) &= 1 & \mu_{i1}(b_i) &= 0 \\ \mu_{i2}(a_i) &= 0 & \mu_{i2}(b_i) &= 1 \\ \mu_{i1}(x_i) + \mu_{i2}(x_i) &= 1 \end{aligned}$$

In this common configuration, it has been shown [23,24] that the matrix X is not of full rank. As a result, the matrix X^TX is not invertible, rendering the original T-S method inapplicable. A detailed proof of this result can be found in [24]. To address the rank deficiency, an efficient alternative, requiring minimal computational effort, is proposed in [24], based on the well-known parameter weighting method.

This method can also be applied to adjust the T-S model parameters using local parameters estimated at specific operating points or obtained directly from physical input/output data. We assume an initial parameter estimate of the form:

$$P_0 = [p_0^0 \ p_1^0 \ p_2^0 \ \dots \ p_n^0]^T \quad (8)$$

which represents a first approximation of the T-S model parameters. The classical least squares method can be used near the equilibrium point to obtain this initial estimate. These reference parameters can then serve as a baseline for all subsystems. Subsequently, the parameter vector of the fuzzy model is refined by minimizing:

$$\begin{aligned} J &= \sum_{k=1}^m (y_k - \hat{y}_k)^2 + \gamma^2 \sum_j (p_{0j} - p_j)^2 = \|Y - XP\|^2 + \gamma^2 \|P_0 - P\|^2 \\ &= \left\| \begin{bmatrix} Y \\ \gamma P_0 \end{bmatrix} - \begin{bmatrix} X \\ \gamma I \end{bmatrix} P \right\|^2 = \|Y_a - X_a P\|^2 \end{aligned} \quad (9)$$

Here, the factor γ reflects the confidence level in the initial parameter estimates. When the system is described by state variables, the model is represented by vector-valued functions of the form

$$f: \mathbb{R}^n \longrightarrow \mathbb{R}^m$$

$$y = f(x_1, x_2, \dots, x_n) \quad y \in \mathbb{R}^m \quad (10)$$

This function is equivalent to m scalar functions:

$$y_j = f_j(x_1, x_2, \dots, x_n) \quad y \in \mathbb{R}, \quad j = 1..m$$

To extend the previous definition (Equation (2)) to m scalar functions, the IF–THEN fuzzy rules are expressed in vectorial form as follows:

$$S_j^{(i_1 \dots i_n)} : \text{If } x_1 \text{ is } M_{j1}^{i_1} \text{ and } x_2 \text{ is } M_{j2}^{i_2} \text{ and } \dots x_n \text{ is } M_{jn}^{i_n} \text{ then}$$

$$\hat{y}_j = p_{j0}^{(i_1 \dots i_n)} + p_{j1}^{(i_1 \dots i_n)} x_1 + p_{j2}^{(i_1 \dots i_n)} x_2 + \dots + p_{jn}^{(i_1 \dots i_n)} x_n \quad (11)$$

By applying the previously described method to each of these scalar models, we obtain:

$$Y_j = \begin{bmatrix} y_{j1} & y_{j2} & \dots & y_{jN} \end{bmatrix}^T$$

$$p_j = \begin{bmatrix} p_{j0}^{(1 \dots 1)} & p_{j1}^{(1 \dots 1)} & \dots & p_{jn}^{(1 \dots 1)} & \dots & p_{j0}^{(r_1 \dots r_n)} & \dots & p_{jn}^{(r_1 \dots r_n)} \end{bmatrix}^T$$

$$X_j = \begin{bmatrix} \beta_{j1}^{(1 \dots 1)} & \beta_{j1}^{(1 \dots 1)} x_{11} & \dots & \beta_{j1}^{(1 \dots 1)} x_{n1} & \dots & \beta_{j1}^{(r_1 \dots r_n)} & \beta_{j1}^{(r_1 \dots r_n)} x_{11} & \dots & \beta_{j1}^{(r_1 \dots r_n)} x_{n1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \beta_{jN}^{(1 \dots 1)} & \beta_{jN}^{(1 \dots 1)} x_{1N} & \dots & \beta_{jN}^{(1 \dots 1)} x_{nN} & \dots & \beta_{jN}^{(r_1 \dots r_n)} & \beta_{jN}^{(r_1 \dots r_n)} x_{1N} & \dots & \beta_{jN}^{(r_1 \dots r_n)} x_{nN} \end{bmatrix}$$

Thus, the adjusted parameter formulation becomes:

$$J_j = \left\| \begin{bmatrix} Y_j \\ \Gamma_j P_{j0} \end{bmatrix} - \begin{bmatrix} X_j \\ \Gamma_j \end{bmatrix} P_j \right\|^2 \quad (12)$$

where Γ_j is equal to $\gamma_j I_j$

$$P_j = (X_{ja}^t X_{ja})^{-1} X_{ja}^t Y_{ja} \quad (13)$$

It is common in T-S model design for the fuzzy sets in the premise part of all subsystems to be identical, such that

$$X_1 = X_2 = \dots = X_m = X$$

Furthermore, if the weighting factors γ_j are equal across all subsystems, they can be grouped as:

$$\begin{bmatrix} P_1 & P_2 & \dots & P_m \end{bmatrix} = (X_a^t X_a)^{-1} X_a^t \begin{bmatrix} Y_1 & Y_2 & \dots & Y_m \\ \Gamma P_{10} & \Gamma P_{20} & \dots & \Gamma P_{m0} \end{bmatrix} \quad (14)$$

A complete proof of this result is provided in [23].

3. Difference Equation Matrix Model Algorithm

Before presenting the proposed FDEMM algorithm, we first briefly review the DEMM algorithm developed by the authors in [22]. In this approach, the system is described by a

difference equation, where past input and output measurements are used as states. For a single-input–single-output (SISO) system, the model is expressed as:

$$y_k + a_1 y_{k-1} + \cdots + a_n y_{k-n} = b_0 + b_1 u_{k-1} + \cdots + b_n u_{k-n} \quad (15)$$

where

- u_k and y_k are the input and output signals at time step k ;
- a_i , for $i = 1, 2, \dots, n$ and b_j , $j = 1, 2, \dots, n$, are the model coefficients.

To incorporate incremental inputs, defined as $\Delta u_k = u_k - u_{k-1}$, into the discrete model of Equation (15), we subtract the same difference equation evaluated at the previous time step. This yields:

$$y_k + (a_1 - 1)y_{k-1} + \cdots + (a_n - a_{n-1})y_{k-n} - a_n y_{k-n-1} = b_1 \Delta u_{k-1} + \cdots + b_n \Delta u_{k-n} \quad (16)$$

The resulting equation can be rewritten as:

$$y_k + \hat{a}_1 y_{k-1} + \hat{a}_2 y_{k-2} + \cdots + \hat{a}_n y_{k-n} + \hat{a}_n y_{k-n-1} = b_1 \Delta u_{k-1} + \cdots + b_n \Delta u_{k-n} \quad (17)$$

Equation (17) can then be expressed in matrix form for a prediction horizon h as:

$$\begin{pmatrix} 1 & 0 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ \hat{a}_1 & 1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ \hat{a}_2 & \hat{a}_1 & 1 & \ddots & \cdots & \cdots & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \cdots & \cdots & \vdots \\ \hat{a}_{n+1} & \hat{a}_n & \cdots & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \hat{a}_{n+1} & \cdots & \cdots & \hat{a}_1 & 1 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & \cdots & 0 & \hat{a}_{n+1} & \cdots & \hat{a}_2 & \hat{a}_1 & 1 \end{pmatrix} \begin{pmatrix} y_{k+1} \\ y_{k+2} \\ \vdots \\ y_{k+h-1} \\ y_{k+h} \end{pmatrix} + \begin{pmatrix} \hat{a}_{n+1} & \hat{a}_n & \hat{a}_{n-1} & \cdots & \hat{a}_1 \\ 0 & \hat{a}_{n+1} & \hat{a}_n & \cdots & \hat{a}_2 \\ 0 & 0 & \hat{a}_{n+1} & \cdots & \hat{a}_3 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \ddots & \hat{a}_{n+1} \\ 0 & 0 & 0 & \ddots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} y_{k-n} \\ y_{k-n+1} \\ \vdots \\ y_{k-1} \\ y_k \end{pmatrix} = \begin{pmatrix} b_n & b_{n-1} & \cdots & \cdots & b_2 \\ 0 & b_n & b_{n-1} & \cdots & b_3 \\ 0 & 0 & b_n & \ddots & b_4 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & b_n \\ \vdots & \vdots & \ddots & \ddots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} \Delta u_{k-n+1} \\ \Delta u_{k-n+2} \\ \vdots \\ \Delta u_{k-2} \\ \Delta u_{k-1} \end{pmatrix} + \begin{pmatrix} b_1 & 0 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ b_2 & b_1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ b_3 & b_2 & b_1 & \ddots & \cdots & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \cdots & \cdots & \vdots \\ b_m & \cdots & \cdots & \cdots & b_1 & 0 & \cdots & 0 \\ 0 & b_m & \cdots & \cdots & b_2 & b_1 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & \cdots & 0 & b_m & \cdots & b_3 & b_2 & b_1 \end{pmatrix} \begin{pmatrix} \Delta u_k \\ \Delta u_{k+1} \\ \vdots \\ \Delta u_{k+h-2} \\ \Delta u_{k+h-1} \end{pmatrix} \quad (18)$$

Rewriting the previous matrix formulation in a compact form and solving for the predicted model out put, we obtain:

$$y_f = -A_1^{-1} A_2 y_p + A_1^{-1} B_1 \Delta u_p + A_1^{-1} B_2 \Delta u_f = A_p y_p + B_p \Delta u_p + B_f \Delta u_f \quad (19)$$

where f and p represent future and past output–input vectors, respectively. Thus, the predicted system output y_f becomes:

$$\begin{aligned} y_f &= \begin{bmatrix} y_{k+1} & y_{k+2} & \cdots & y_{k+h-1} & y_{k+h} \end{bmatrix}^t \in \mathbb{R}^h \\ y_p &= \begin{bmatrix} y_{k-n} & y_{k-n+1} & \cdots & y_{k-1} & y_k \end{bmatrix}^t \in \mathbb{R}^{n+1} \\ \Delta u_p &= \begin{bmatrix} \Delta u_{k-n+1} & \Delta u_{k-n+2} & \cdots & \Delta u_{k-2} & \Delta u_{k-1} \end{bmatrix}^t \in \mathbb{R}^{n-1} \\ \Delta u_f &= \begin{bmatrix} \Delta u_k & \Delta u_{k+1} & \cdots & \Delta u_{k+h-2} & \Delta u_{k+h-1} \end{bmatrix}^t \in \mathbb{R}^h \\ A_p &\in \mathbb{R}^{h \times n+1} \\ B_p &\in \mathbb{R}^{h \times n-1} \\ B_f &\in \mathbb{R}^{h \times h} \end{aligned}$$

At this stage, the remaining control coefficients can be obtained using any state-feedback control design methodology. In this work, an LQR-based formulation is adopted in order to achieve a balanced trade-off between dynamic performance and control effort. It should be noted, however, that the combination of the DEMM structure with an LQR formulation does not, by itself, guarantee closed-loop stability in a strict theoretical sense. While alternative approaches such as pole placement can enforce specific closed-loop dynamics, they may lead to excessive control actions and actuator saturation, which can significantly degrade performance and potentially destabilize the system in practical implementations.

The LQR approach provides a systematic way to balance tracking performance and control effort through the selection of the weighting matrices Q and R , allowing the designer to shape the closed-loop dynamics while maintaining admissible control amplitudes. This results in a practical compromise between stability margins, transient response speed, and control energy. Consequently, stability and performance must be evaluated a posteriori through analysis and simulation, rather than being assumed a priori from the control structure itself. This design philosophy prioritizes implementability and robustness in realistic operating conditions over purely theoretical guarantees.

The controller minimizes the following optimization function:

$$J = (r - y_f)^T Q (r - y_f) + \Delta u_f^T R \Delta u_f \quad (20)$$

Let r be the reference vector over the prediction horizon, Q the output weighting matrix, and R the control increment weighting matrix. The optimization variable is Δu_f (future incremental inputs). The free response y_f denotes the predicted output when $\Delta u_f = 0$. Here, Q penalizes deviations from the reference trajectory, while R weights the control effort. Both Q and R are selected as positive definite matrices, as recommended in [22]. The analytic minimizer is obtained by solving the quadratic program; when no constraints are imposed on Equation (19), the analytical solution is given by:

$$\Delta u_f = (B_f^T Q B_f + R)^{-1} B_f^T Q (r - A_p y_p - B_p \Delta u_p) \quad (21)$$

Furthermore, if the system exhibits a pure time delay of d units, then:

$$y_k + a_1 y_{k-1} + \cdots + a_n y_{k-n} = b_0 + b_{d+1} u_{k-d-1} + \cdots + b_n u_{k-n} \quad (22)$$

This formulation closely resembles the LQR, with the key difference being that the output error is minimized instead of the state error. Additionally, because the controller

computes incremental input values, it inherently provides integral action, ensuring zero steady-state error.

The resulting control law can be expressed as:

$$\Delta u_f = \left(B_f^T Q B_f + R \right)^{-1} B_f^T Q \varepsilon = M \varepsilon \quad (23)$$

where

$$\varepsilon = r - y_l$$

Here, y_l represents the free response of the system, i.e., the predicted output when $\Delta u_f = 0$. Thus, we have:

$$y_l = A_p y_p + B_p \Delta u_p \quad (24)$$

This prediction can be corrected to account for potential model mismatch by adding a correction vector consisting of h identical elements of the form $y_{k_{measured}} - y_{k_{calculated}}$, where $y_{k_{calculated}}$ is obtained from Equation (17):

$$\tilde{y}_l = y_l + [y_{k_{measured}} - y_{k_{calculated}}] \quad (25)$$

The resulting corrected control action becomes:

$$\Delta u_f = \left(B_f^T Q B_f + R \right)^{-1} B_f^T Q (r - \tilde{y}_l) \quad (26)$$

By extending Equation (19) to multivariable systems with i outputs and p inputs, we obtain:

$$y_{if} = A_{ip} y_p + B_{i1p} \Delta u_{1p} + \dots + B_{ipp} \Delta u_{pp} + B_{i1f} \Delta u_{1f} + \dots + B_{ipf} \Delta u_{pf} \quad (27)$$

In this case, the general form of the DEMM algorithm for multivariable systems becomes:

$$\begin{pmatrix} y_{1f} \\ y_{2f} \\ \vdots \\ y_{qf} \end{pmatrix} = \begin{pmatrix} A_{1p} & 0 & \dots & 0 \\ 0 & A_{2p} & \ddots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & A_{qp} \end{pmatrix} \begin{pmatrix} y_{1p} \\ y_{2p} \\ \vdots \\ y_{qp} \end{pmatrix} + \begin{pmatrix} B_{11p} & B_{12p} & \dots & B_{1sp} \\ B_{21p} & B_{22p} & \ddots & B_{2sp} \\ \vdots & \ddots & \ddots & \vdots \\ B_{q1p} & B_{q2p} & \dots & B_{qsp} \end{pmatrix} \begin{pmatrix} \Delta u_{1p} \\ \Delta u_{2p} \\ \vdots \\ \Delta u_{sp} \end{pmatrix} + \begin{pmatrix} B_{11f} & B_{12f} & \dots & B_{1sf} \\ B_{21f} & B_{22f} & \ddots & B_{2sf} \\ \vdots & \ddots & \ddots & \vdots \\ B_{q1f} & B_{q2f} & \dots & B_{qsf} \end{pmatrix} \begin{pmatrix} \Delta u_{1f} \\ \Delta u_{2f} \\ \vdots \\ \Delta u_{sf} \end{pmatrix} \quad (28)$$

which has the same structure as the monovariable case.

4. Fuzzy Difference Equation Matrix Model Algorithm

To extend the DEMM framework to nonlinear systems, it is necessary to modify the T-S model in order to generate a set of fuzzy models of the dynamic system, expressed in matrix form. Each IF–THEN rule $S^{(i_0 \dots i_n)}$, corresponding to the predicted plant model given by Equation (19), takes the form

$$\begin{aligned} S^{(i_0 \dots i_n)} : & \text{if } y_k \text{ is } M_0^{i_0} \text{ and } y_{k-1} \text{ is } M_1^{i_1} \text{ and } \dots y_{k-n} \text{ is } M_n^{i_n} \\ & \text{then } y_f = A_p^{(i_0 \dots i_n)} y_p + B_p^{(i_0 \dots i_n)} \Delta u_p + B_f^{(i_0 \dots i_n)} \Delta u_f \end{aligned} \quad (29)$$

$$y_f = \frac{\sum_{i_0=1}^{r_0} \dots \sum_{i_n=1}^{r_n} \omega^{(i_0 \dots i_n)}(y_p) \left[A_p^{(i_0 \dots i_n)} y_p + B_p^{(i_0 \dots i_n)} \Delta u_p + B_f^{(i_0 \dots i_n)} \Delta u_f \right]}{\sum_{i_0=1}^{r_0} \dots \sum_{i_n=1}^{r_n} \omega^{(i_0 \dots i_n)}(y_p)}$$

$$\omega^{(i_0 \dots i_n)}(y_p) = \mu_{0i_0}(y_k) \mu_{1i_1}(y_{k-1}) \dots \mu_{ni_n}(y_{k-n})$$

where $\mu_{ji_j}(y_{k-j})$ is the MF associated with the fuzzy set $M_j^{i_j}$.

The procedure for computing the control action using the proposed FDEM algorithm is as follows:

1. Compute the set of matrices:

$$\left\{ A_p^{(i_0 \dots i_n)}, B_p^{(i_0 \dots i_n)}, B_f^{(i_0 \dots i_n)} \right\}$$

2. Select the weighting matrices Q and R .
3. Compute matrix M for each T-S fuzzy rule:

$$M^{(i_0 \dots i_n)} = \left(B_f^{(i_0 \dots i_n)T} Q B_f^{(i_0 \dots i_n)} + R \right)^{-1} B_f^{(i_0 \dots i_n)T} Q$$

Then, compute $m_1^{(i_0 \dots i_n)}$ as:

$$m_1^{(i_0 \dots i_n)} = M^{(i_0 \dots i_n)}(1, :)$$

4. Measure the actual output $y_{k_{measured}}$.
5. Compute $y_{k_{calculated}}$ by defuzzifying the consequent parts of all fuzzy rules using Equation (15).
6. Compute the system's free response using the fuzzy model:

$$\begin{aligned} S^{(i_0 \dots i_n)} : & \text{if } y_k \text{ is } M_0^{i_0} \text{ and } y_{k-1} \text{ is } M_1^{i_1} \text{ and } \dots y_{k-n} \text{ is } M_n^{i_n} \\ & \text{then } y_l = A_p^{(i_0 \dots i_n)} y_p + B_p^{(i_0 \dots i_n)} \Delta u_p \end{aligned} \quad (30)$$

$$y_l = \frac{\sum_{i_0=1}^{r_0} \dots \sum_{i_n=1}^{r_n} \omega^{(i_0 \dots i_n)}(y_p) \left[A_p^{(i_0 \dots i_n)} y_p + B_p^{(i_0 \dots i_n)} \Delta u_p \right]}{\sum_{i_0=1}^{r_0} \dots \sum_{i_n=1}^{r_n} \omega^{(i_0 \dots i_n)}(y_p)}$$

$$\tilde{y}_l = y_l + [y_{k_{measured}} - y_{k_{calculated}}]$$

7. Calculate the correction term:

$$\varepsilon = r - \tilde{y}_l$$

8. Compute the control input u_k . First, calculate m_1 :

$$\begin{aligned} C^{(i_0 \dots i_n)} : & \text{if } y_k \text{ is } M_0^{i_0} \text{ and } y_{k-1} \text{ is } M_1^{i_1} \text{ and } \dots y_{k-n} \text{ is } M_n^{i_n} \\ & \text{then } m_1^{(i_0 \dots i_n)} = M^{(i_0 \dots i_n)}(1, :) \end{aligned}$$

$$m_1 = \frac{\sum_{i_0=1}^{r_0} \dots \sum_{i_n=1}^{r_n} \omega^{(i_0 \dots i_n)}(y_p) \left[M^{(i_0 \dots i_n)}(1, :) \right]}{\sum_{i_0=1}^{r_0} \dots \sum_{i_n=1}^{r_n} \omega^{(i_0 \dots i_n)}(y_p)}$$

$$\Delta u_k = m_1 \varepsilon$$

$$u_k = u_{k-1} + \Delta u_k$$

Steps 1, 2, and 3 can be performed offline, while steps 4 through 8 must be executed at each sampling instant during real-time operation.

Computational Complexity of FDEMM

A key advantage of the proposed FDEMM architecture is its computational efficiency. As established in [24], the underlying DEMM formulation is computationally lighter than traditional DMC. Unlike standard Nonlinear MPC (NMPC) strategies, which often rely on computationally expensive iterative optimization solvers online, FDEMM computes the control law analytically (Equation (26)). The integration of the T-S fuzzy model adds only the calculation of membership weights and a linear combination of local models, operations with low computational cost. Consequently, FDEMM retains the real-time feasibility of linear MPC while providing the capability to control nonlinear dynamics.

5. Case Studies

Two case studies were selected to evaluate distinct aspects of the controller. The distillation column serves as a benchmark for high-nonlinearity and coupled multivariable dynamics, whereas the thermal mixing tank is employed specifically to test the controller's efficacy in handling significant transport delays.

5.1. Modelling and Control of a Distillation Column Using FDEMM

A two-product distillation column is chosen as a case study, as illustrated in Figure 2. Acceptable operation of a binary distillation column typically requires controlling both the distillate composition and the bottom product composition.

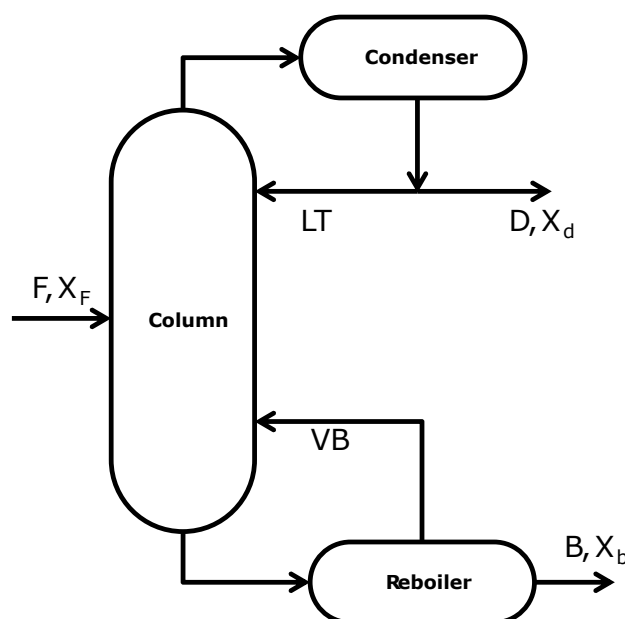


Figure 2. Binary distillation column.

We assume a binary mixture with constant relative volatility throughout the column and ideal (100% efficient) trays; that is, the vapor leaving a tray is in equilibrium with the liquid on the tray. This assumption allows the use of a simplified vapor–liquid equilibrium relationship:

$$y_i^* = \frac{\alpha_{ij}x_i}{1 + (\alpha - 1)x_i} \quad (31)$$

where x_i is the liquid-phase mole fraction of the more volatile component on the i^{th} tray, y_i^* is the vapor-phase composition of the same component on the same tray, and α_{ij} is the relative volatility of component i with respect to component j .

The overhead vapor is condensed in a condenser and collected in the reflux drum, which has a total holdup M_D . The contents of the drum are assumed to be perfectly mixed, with a composition x_d . The distillate is withdrawn at a rate D .

At the base of the column, the bottom product is withdrawn at a rate B , with a composition x_b . The vapor boil-up is generated by a thermosiphon reboiler at a rate V . It is assumed that the liquid in the reboiler and the column base are perfectly mixed and have the same composition x_b , with a total holdup M_B .

Each stage of the column is modeled with a total mass balance and a component mass balance. Each stage includes two parts: a tray where vapor–liquid contact occurs, and a downcomer, through which liquid flows to the next tray.

a Mass Balance: Feed Stage

At the feed stage, the equilibrium and overflow assumptions remain valid. The total mass and component mass balances are:

Total mass balance:

$$\frac{dM_{TN}}{dt} = L_{n+1} + V_{n-1} - L_n - V_n + F \quad (32)$$

Component mass balance:

$$\frac{dM_{TN}x_n}{dt} = L_{n+1}x_{n+1} + V_{n-1}y_{n-1} - L_nx_n - V_ny_n + Fx_F \quad (33)$$

where M_{TN} is the holdup on each tray, L_n is the liquid flow rate between trays, V_n is the vapor flow rate, and x_n, y_n are liquid and vapor compositions, respectively. x_F is the feed composition.

b Mass Balance: Reboiler

Total mass balance:

$$\frac{dM_B}{dt} = L_2 - V_1 - B \quad (34)$$

where V and B are the vapor and liquid flows at the bottom.

Component mass balance:

$$\frac{dM_Bx_1}{dt} = L_2x_2 + V_1y_1 - Bx_1 \quad (35)$$

c Mass Balance: Condenser and Reflux Drum

Total mass balance:

$$\frac{dM_D}{dt} = V_{NT-1} - LT - D \quad (36)$$

where V_{NT-1} is the vapor flow entering the condenser, and D is the distillate flow rate.

Component mass balance:

$$\frac{dM_Dx_d}{dt} = V_{NT-1}y_{NT-1} - LTx_{NT} - Dx_d \quad (37)$$

Here, x_d (distillate composition) and x_b (bottom composition) are the controlled outputs; LT (reflux flow rate) and VB (boil-up flow rate) are the manipulated variables; and F is the feed rate, considered as a disturbance.

To select the time delays for the fuzzy model, we analyze the influence of delayed inputs on each output using the least squares method. The influence of the following input–output combinations is considered:

$$\begin{aligned} &\{LT_{k-1}, LT_{k-2}, LT_{k-3}, LT_{k-4}, LT_{k-5}, LT_{k-6}, LT_{k-7}\} \text{ in } X_b \\ &\{LT_{k-1}, LT_{k-2}, LT_{k-3}, LT_{k-4}, LT_{k-5}, LT_{k-6}, LT_{k-7}\} \text{ in } X_d \\ &\{VB_{k-1}, VB_{k-2}, VB_{k-3}, VB_{k-4}, VB_{k-5}, VB_{k-6}, VB_{k-7}\} \text{ in } X_b \\ &\{VB_{k-1}, VB_{k-2}, VB_{k-3}, VB_{k-4}, VB_{k-5}, VB_{k-6}, VB_{k-7}\} \text{ in } X_d \end{aligned} \quad (38)$$

The most significant delays for each case are selected as:

$$\begin{aligned} <_{k-1} \text{ and } LT_{k-2} \text{ for } X_b \\ <_{k-1} \text{ and } LT_{k-2} \text{ for } X_d \\ &VB_{k-1} \text{ and } VB_{k-2} \text{ for } X_b \\ &VB_{k-2} \text{ and } VB_{k-3} \text{ for } X_d \end{aligned} \quad (39)$$

Thus, the objective is to obtain a model of the form:

$$\begin{aligned} X_{d_k} + a_{d1}X_{d_{k-1}} + a_{d2}X_{d_{k-2}} &= b_{d0} + b_{d1}LT_{k-1} + b_{d2}LT_{k-2} + c_{d1}VB_{k-2} + c_{d2}VB_{k-3} \\ X_{b_k} + a_{b1}X_{b_{k-1}} + a_{b2}X_{b_{k-2}} &= b_{b0} + b_{b1}LT_{k-1} + b_{b2}LT_{k-2} + c_{b1}VB_{k-1} + c_{b2}VB_{k-2} \end{aligned} \quad (40)$$

The system is identified using the generalized T-S model with parameter tuning. The initial estimation is obtained by linearizing the system around an operating point and applying the classical least squares method to calculate the parameters P_0 (see Section 2). To ensure an informative identification, the system is excited using random Gaussian inputs. This leads to the following linear model:

$$\begin{aligned} X_{d_k} - 1.730X_{d_{k-1}} + 0.7436X_{d_{k-2}} \\ &= 0.0125 + 0.0033LT_{k-1} - 0.0001LT_{k-2} - 0.0024VB_{k-2} - 0.0003VB_{k-3} \\ X_{b_k} - 1.1654X_{b_{k-1}} + 0.3161X_{b_{k-2}} \\ &= 0.0948 + 0.0236LT_{k-1} - 0.0072LT_{k-2} - 0.0128VB_{k-1} - 0.0144VB_{k-2} \end{aligned} \quad (41)$$

Next, the nonlinear fuzzy T-S model is obtained. The MFs for X_{d_k} and X_{b_k} are shown in Figure 3a and 3b, respectively. The resulting fuzzy model is:

$$\begin{aligned} &\text{If } X_{d_k} \text{ is } M_{d0}^1 \\ &\quad \text{Then } X_{d_k} + a_{d1}^1X_{d_{k-1}} + a_{d2}^1X_{d_{k-2}} = b_{d0}^1 + b_{d1}^1LT_{k-1} + b_{d2}^1LT_{k-2} + c_{d1}^1VB_{k-2} + c_{d2}^1VB_{k-3} \\ &\text{If } X_{d_k} \text{ is } M_{d0}^2 \\ &\quad \text{Then } X_{d_k} + a_{d1}^2X_{d_{k-1}} + a_{d2}^2X_{d_{k-2}} = b_{d0}^2 + b_{d1}^2LT_{k-1} + b_{d2}^2LT_{k-2} + c_{d1}^2VB_{k-2} + c_{d2}^2VB_{k-3} \\ &\text{If } X_{d_k} \text{ is } M_{d0}^3 \\ &\quad \text{Then } X_{d_k} + a_{d1}^3X_{d_{k-1}} + a_{d2}^3X_{d_{k-2}} = b_{d0}^3 + b_{d1}^3LT_{k-1} + b_{d2}^3LT_{k-2} + c_{d1}^3VB_{k-2} + c_{d2}^3VB_{k-3} \\ &\text{If } X_{b_k} \text{ is } M_{b0}^1 \\ &\quad \text{Then } X_{b_k} + a_{b1}^1X_{b_{k-1}} + a_{b2}^1X_{b_{k-2}} = b_{b0}^1 + b_{b1}^1LT_{k-1} + b_{b2}^1LT_{k-2} + c_{b1}^1VB_{k-1} + c_{b2}^1VB_{k-2} \\ &\text{If } X_{b_k} \text{ is } M_{b0}^2 \\ &\quad \text{Then } X_{b_k} + a_{b1}^2X_{b_{k-1}} + a_{b2}^2X_{b_{k-2}} = b_{b0}^2 + b_{b1}^2LT_{k-1} + b_{b2}^2LT_{k-2} + c_{b1}^2VB_{k-1} + c_{b2}^2VB_{k-2} \\ &\text{If } X_{b_k} \text{ is } M_{b0}^3 \\ &\quad \text{Then } X_{b_k} + a_{b1}^3X_{b_{k-1}} + a_{b2}^3X_{b_{k-2}} = b_{b0}^3 + b_{b1}^3LT_{k-1} + b_{b2}^3LT_{k-2} + c_{b1}^3VB_{k-1} + c_{b2}^3VB_{k-2} \end{aligned} \quad (42)$$

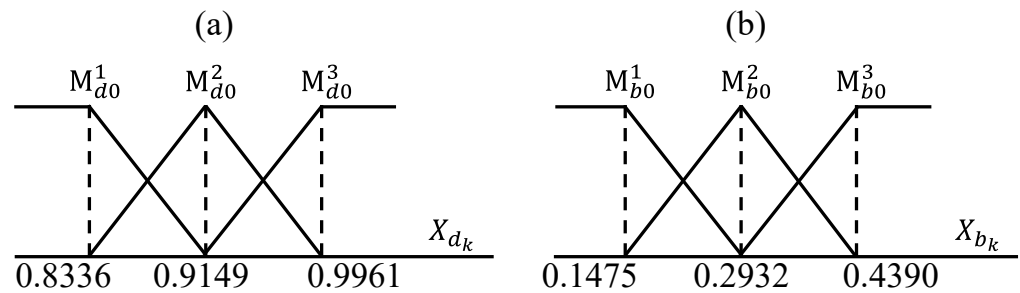


Figure 3. MFs: (a) X_{dk} and (b) X_{bk} .

Using the estimation method described earlier, the final fuzzy model becomes:

$$\begin{aligned}
 &\text{If } X_{dk} \text{ is } M_{d0}^1 \text{ Then } X_{dk} - 1.7230X_{dk-1} + 0.7399X_{dk-2} \\
 &\quad = 0.0147 + 0.0125LT_{k-2} - 0.0043LT_{k-3} - 0.0055VB_{k-5} + 0VB_{k-6} \\
 &\text{If } X_{dk} \text{ is } M_{d0}^2 \text{ Then } X_{dk} - 1.7199X_{dk-1} + 0.7282X_{dk-2} \\
 &\quad = 0.0057 + 0.0057LT_{k-2} + 0.002LT_{k-3} - 0.0037VB_{k-5} - 0.0004VB_{k-6} \\
 &\text{If } X_{dk} \text{ is } M_{d0}^3 \text{ Then } X_{dk} - 1.7351X_{dk-1} + 0.737X_{dk-2} \\
 &\quad = 0.0028 + 0.0004LT_{k-2} - 0.0004LT_{k-3} - 0.0003VB_{k-5} - 0.0007VB_{k-6} \\
 &\text{If } X_{bk} \text{ is } M_{b0}^1 \text{ Then } X_{bk} - 1.1653X_{bk-1} + 0.3288X_{bk-2} \\
 &\quad = 0.0594 + 0.0202LT_{k-2} + 0.0101LT_{k-3} - 0.0128VB_{k-1} + 0VB_{k-2} \\
 &\text{If } X_{bk} \text{ is } M_{b0}^2 \text{ Then } X_{bk} - 1.1029X_{bk-1} + 0.2787X_{bk-2} \\
 &\quad = 0.1087 + 0.02868LT_{k-2} - 0.0061LT_{k-3} - 0.0111VB_{k-1} - 0.0222VB_{k-2} \\
 &\text{If } X_{bk} \text{ is } M_{b0}^3 \text{ Then } X_{bk} - 1.2104X_{bk-1} + 0.2661X_{bk-2} \\
 &\quad = 0.0574 + 0.0152LT_{k-2} - 0.0115LT_{k-3} - 0.0181VB_{k-1} + 0.0018VB_{k-2}
 \end{aligned} \tag{43}$$

Figure 4 presents the block diagram of the proposed FDEM controller applied to the distillation column.

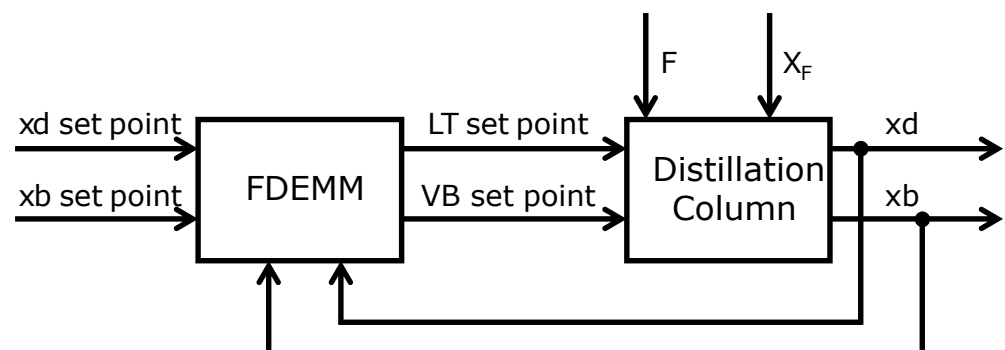


Figure 4. Control loop of the distillation column applying the proposed FDEM.

In this case, the system has been modeled to operate with a prediction horizon of 20 samples ($h = 20$) and the weighting matrices Q and R have been chosen to be $Q_{xd} = I$, $Q_{xb} = 100I$, and $R = I$. The choice has been done utilizing a commonly adopted strategy in the literature, that involves choosing Q as a diagonal matrix with relatively large values on the diagonal to emphasize state penalties and R set as the identity matrix to impose a standard control effort penalty (in our case the diagonal of the matrix Q is divided between the two outputs, Q_{xd} for the reflux drum distilled and Q_{xb} for the bottom distilled, and we are imposing a stronger penalization for the bottom distilled in comparison to the top drum

one). Figure 5a–d shows the transient response using FDEM when reference setpoints for x_d and x_b are changed when the system stabilizes, first at $t = 100$ min and later at $t = 200$ min, respectively for x_d and x_b . The feed rate F and x_F are kept constant.

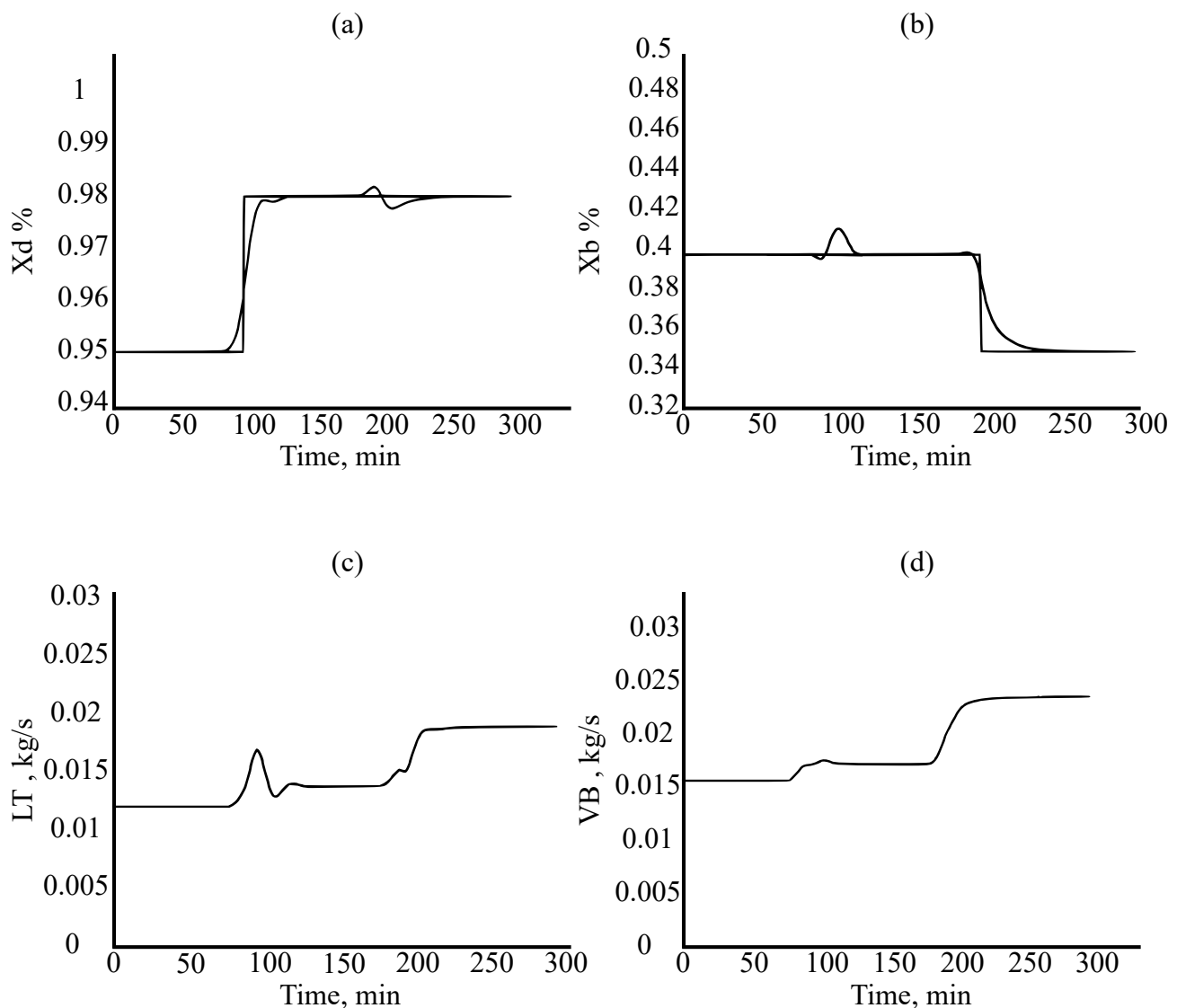


Figure 5. Transient response with a reference change in x_d and x_b . F and x_F are constants: (a) x_d , (b) x_b , (c) LT and (d) VB .

To test disturbance rejection, a step disturbance of $F = 10\%$ and $x_F = 10\%$ is introduced at $t = 100$ and $t = 200$ min, respectively, while keeping the reference inputs constant. The system's response, including outputs and control actions, is shown in Figure 6a–f.

Additionally, noise robustness is evaluated by introducing Gaussian noise with a standard deviation of $\sigma = 5\%$ to the average of F and x_F , with constant reference inputs. The resulting output and control responses are shown in Figure 7a–f.

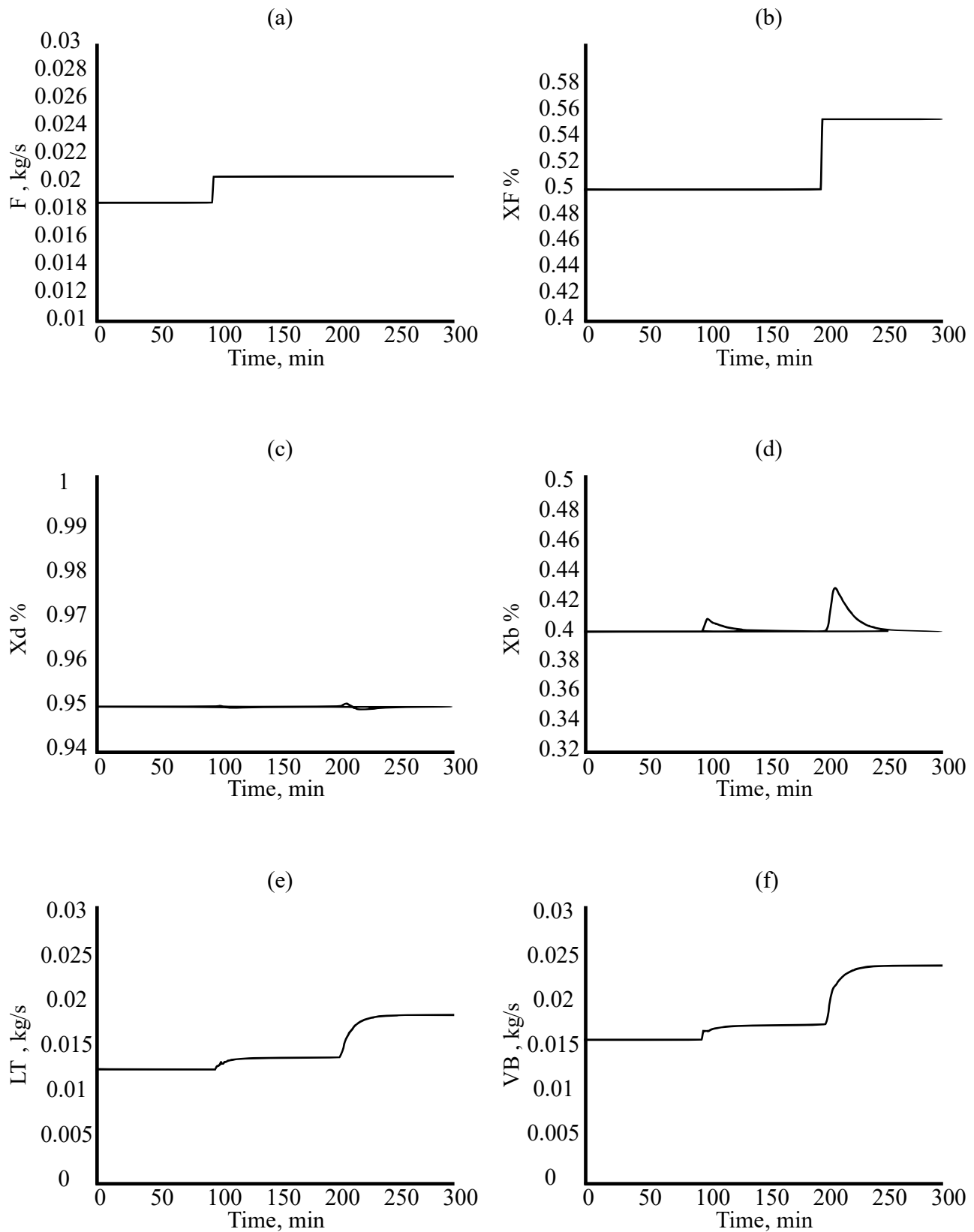


Figure 6. Transient response under step disturbance of $F = 10\%$ and $x_F = 10\%$: (a) F , (b) x_F , (c) x_d , (d) x_b , (e) LT and (f) VB .

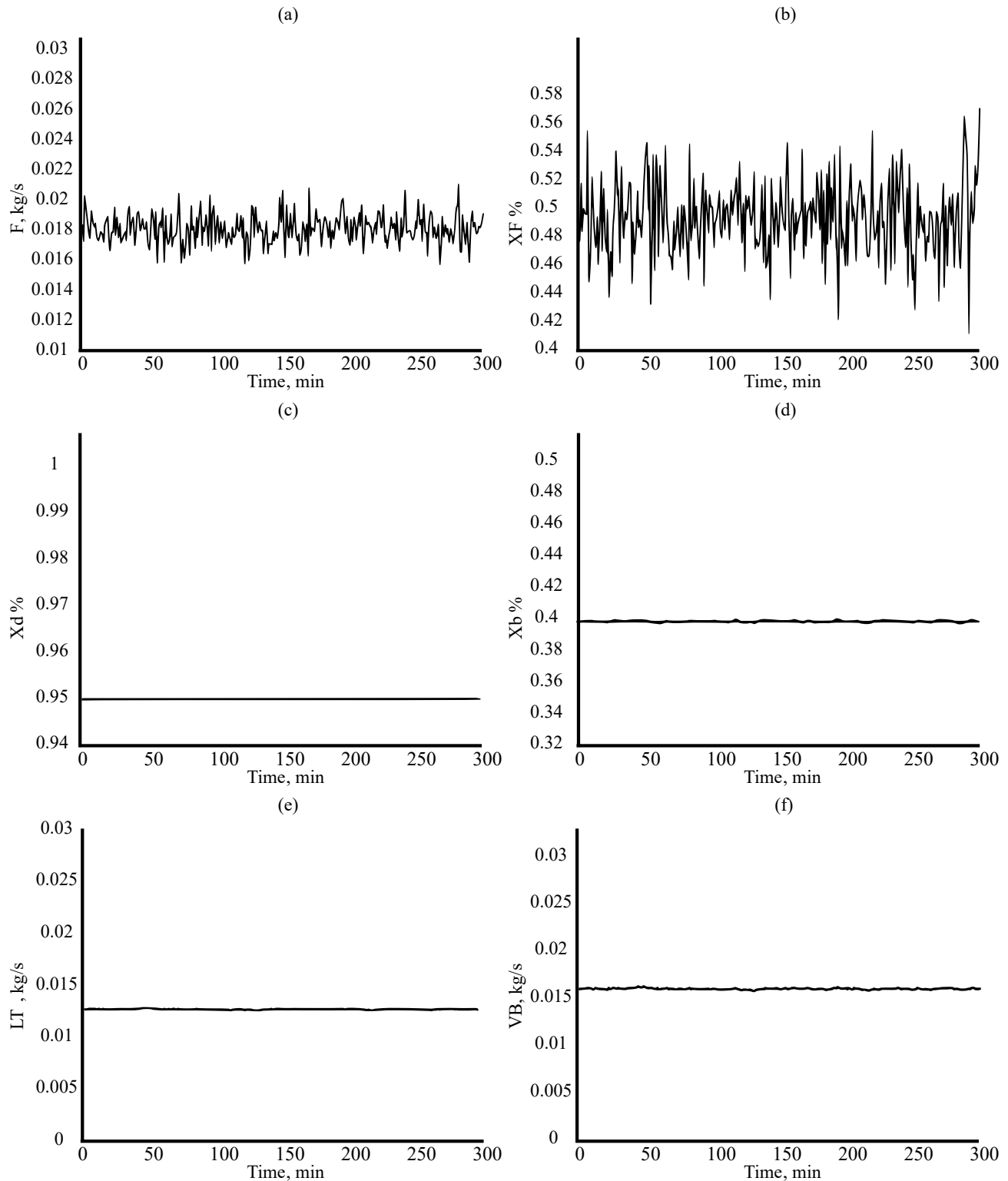


Figure 7. Transient response subjected to noise effect of $\sigma = 5\%$ in the average of F and x_F with constant reference inputs: (a) F , (b) x_F , (c) x_d , (d) x_b , (e) LT and (f) VB .

5.2. Modelling and Control of MIMO Thermal Mixing Process Using FDEMM

The problem of dosing hot and cold fluids to produce a mixture with a desired temperature and flow rate is widely recognized, even outside the field of control engineering. Beyond the common example of a household shower, this problem is frequently encountered in industrial processes as can be seen in [25,26].

Consider the MIMO thermal mixing process illustrated in Figure 8. The setup consists of a mixing chamber, hot and cold fluid inlet valves, and an outlet valve. The main control objective is to maintain both the output flow rate Q_m and the outlet temperature T at their desired reference values. The system assumes a hot water inlet at temperature T_1 and a cold water inlet at temperature T_2 , both considered constant. The two inlet flow rates Q_1 and Q_2 serve as manipulated variables, controlled by motorized valves with simplified dynamics:

$$Q_1(s) = \frac{K}{1+T_1s} e^{-\tau_1s} U_1(s) \quad (44)$$

$$Q_2(s) = \frac{K}{1+T_2s} e^{-\tau_2s} U_2(s) \quad (45)$$

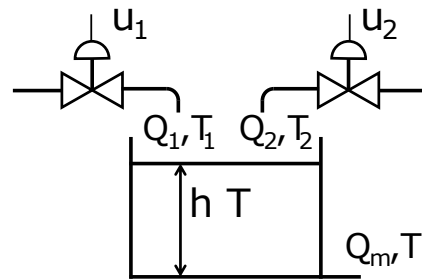


Figure 8. MIMO thermal mixing process.

The system includes two measurements, the outlet temperature T , which is assumed to be measured without delay, and the output flow rate, whose measurement Q_m is delayed due to flowmeter placement and sensing technology:

$$Q_m(t) = Q_s(t - \tau_s) \quad (46)$$

Thermal losses are also considered, modeled as being proportional to the temperature difference between the tank and the environment. These act as disturbances, yielding the full model:

$$Q_1 + Q_2 - Q_s = A \frac{dh}{dt} \quad (47)$$

$$T_1 Q_1 + T_2 Q_2 - T Q_s + K_p(T - T_e) = A \frac{dT}{dt} \quad (48)$$

$$Q_s = \alpha \sqrt{h} \quad (49)$$

Figure 9 presents the block diagram of the proposed FDEMM controller applied to the thermal mixing tank.

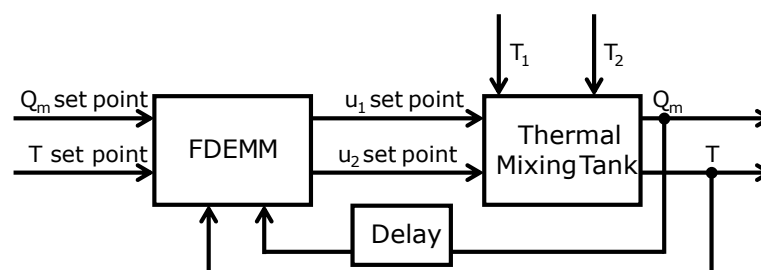


Figure 9. Control loop of the thermal mixing tank applying the proposed FDEMM.

Assume the following process parameters:

- $A = 1 \text{ m}^2$;
- $\alpha = 1$;

- $T_1 = 90\text{ }^{\circ}\text{C}$;
- $T_2 = 10\text{ }^{\circ}\text{C}$;
- $K = 0.1$;
- $\tau_1 = 1\text{ s}$;
- $\tau_2 = 0.5\text{ s}$;
- $\tau_5 = 1\text{ s}$.

The central operating point is:

$$u_{1_0} = 2.0044, u_{2_0} = 1.9956, T_{e_0} = 15\text{ }^{\circ}\text{C}, Q_{S_0} = 4 \frac{m^3}{sec}, T_0 = 50\text{ }^{\circ}\text{C}$$

A sampling time of 0.5 s is used.

Next, a linear model is identified around this operating point (parameters of P_0) as described in [23]:

$$Q_m(k+1) = 0.0010 + 1.8590Q_m(k) - 0.8651Q_m(k-1) + 0.0038u_1(k-4) + 0.0018u_1(k-5) + 0.0047u_2(k-3) + 0.0012u_2(k-4) \quad (50)$$

$$T(k+1) = 0.2984 + 2.4049T(k) - 1.9350T(k-1) + 0.5243T(k-2) + 0.1238u_1(k-2) - 0.00614u_1(k-3) - 0.1696u_2(k-1) + 0.1030u_2(k-2) \quad (51)$$

From the identified model, the following input–output delays are observed:

- Input $u_1 \rightarrow$ measured flow rate Q_m : 5 sampling units;
- Input $u_2 \rightarrow$ measured flow rate Q_m : 3 sampling units;
- Input $u_1 \rightarrow$ outlet temperature T : 3 units;
- Input $u_2 \rightarrow$ outlet temperature T : 2 units.

With this information, a T-S model is identified using input–output test data. Three triangular MFs are defined over the range [1, 7] for flow rate and [20, 80] for temperature; that makes 2 inputs with 3 MFs each resulting in a fuzzy model with $3^2 = 9$ rules ($S^{11}, S^{12}, S^{13}, S^{21}, S^{22}, S^{23}, S^{31}, S^{32}, S^{33}$):

$$\begin{aligned} S^{11} : & \text{if } Q_m(k) \text{ is } M_1^1 \text{ \& } T(k) \text{ is } M_2^1 \text{ then} \\ & Q_m(k+1) = 0.0023 + 1.8567Q_m(k) - 0.8670Q_m(k-1) \\ & \quad + 0.0061u_1(k-4) + 0.0029u_1(k-5) + 0.0079u_2(k-3) + 0.0022u_2(k-4) \\ & T(k+1) = 0.2984 + 2.4035T(k) - 1.9350T(k-1) + 0.5254T(k-2) \\ & \quad + 0.1225u_1(k-2) - 0.0622u_1(k-3) - 0.1693u_2(k-1) + 0.1030u_2(k-2) \\ & \vdots \\ S^{33} : & \text{if } Q_m(k) \text{ is } M_1^3 \text{ \& } T(k) \text{ is } M_2^3 \text{ then} \\ & Q_m(k+1) = 0.0011 + 1.8600Q_m(k) - 0.8640Q_m(k-1) \\ & \quad + 0.0062u_1(k-4) + 0.0010u_1(k-5) + 0.0034u_2(k-3) + 0.0008u_2(k-4) \\ & T(k+1) = 0.2984 + 2.4048T(k) - 1.9349T(k-1) + 0.5244T(k-2) \\ & \quad + 0.1225u_1(k-2) - 0.0622u_1(k-3) - 0.1693u_2(k-1) + 0.1030u_2(k-2) \end{aligned}$$

where S^{xy} refers to the rule pertaining to the x 's MF of the first input (flow rate) and the y 's MF of the second one (temperature).

The identification procedure follows the methodology described in Section 2, and the FDEMM controller is applied according to the approach outlined in Section 4. The tuning parameters are:

$$h = 20, Q = \begin{bmatrix} 50I & 0 \\ 0 & 0.5I \end{bmatrix}, R = \begin{bmatrix} 1000I & 0 \\ 0 & 1000I \end{bmatrix}$$

Figure 10 shows the transient response of the system to step changes in the flow and temperature reference signals, using a constant reference vector $y(k+1)$.

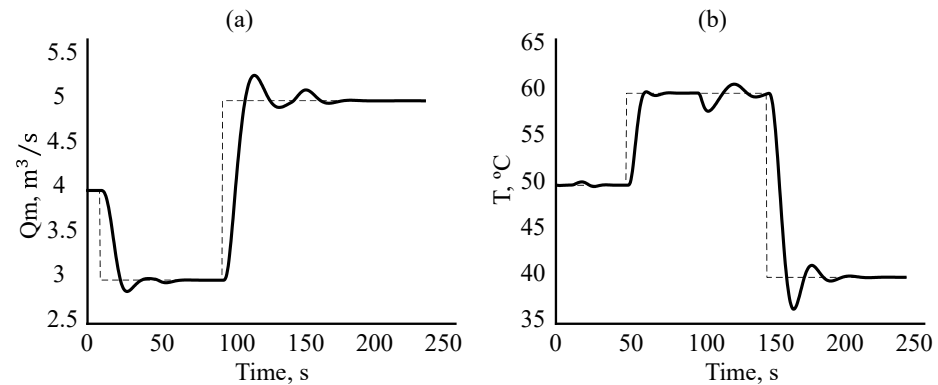


Figure 10. Transient response of the thermal mixing process when it is subjected to: (a) changes in the flow rate reference signal and (b) changes in the temperature reference signal.

Figure 11 presents the system response to the same reference changes, assuming the reference vector includes future values over a prediction horizon h . The predictive capability of the controller allows the system response to lead the reference signal, improving performance.

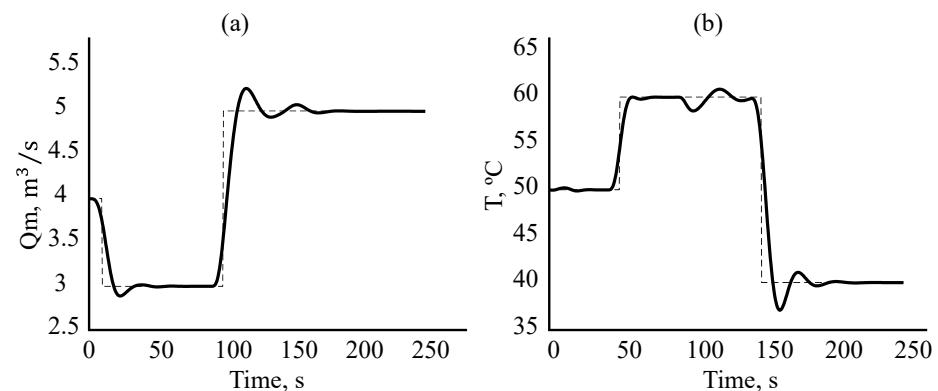


Figure 11. Transient response when the reference vector has future references with prediction horizon h : (a) changes in the flow rate reference signal and (b) changes in the temperature reference signal.

Figure 12 shows the transient response of the system when subjected to successive ramp signals in both the flow and temperature references.

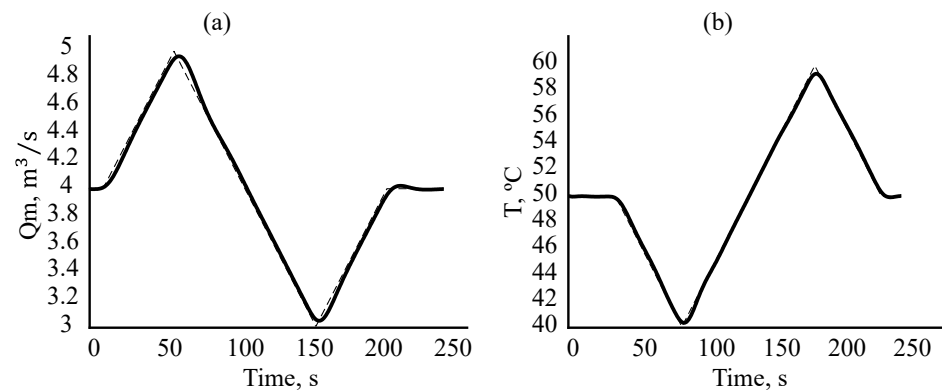


Figure 12. Transient response when the system is subjected to successive ramp signals in the (a) flowrate reference signal and (b) temperature reference signal.

Finally, Figure 13 illustrates the influence of the prediction horizon. In this case, the horizon is reduced to $h = 10$, and the resulting degradation in control performance is clearly observed.

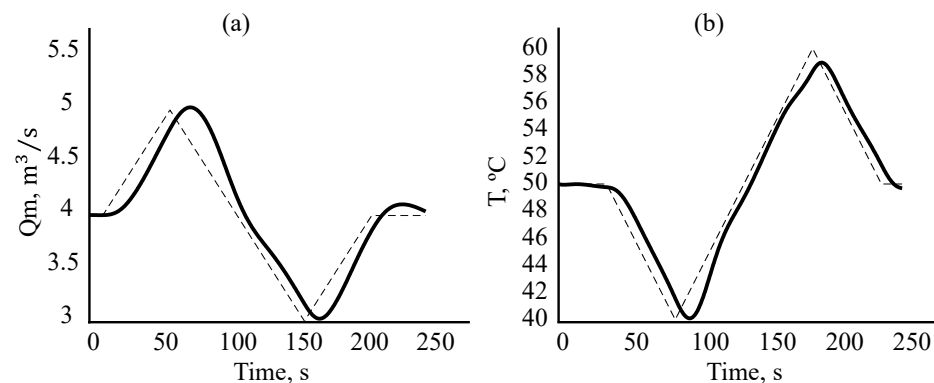


Figure 13. Transient response when the system is subjected to changes in prediction horizon in the (a) flowrate reference signal and (b) temperature reference signal.

5.3. Discussion

The simulation results presented in Sections 5.1 and 5.2 highlight the main characteristics and practical implications of the proposed FDEM control strategy when applied to nonlinear multivariable systems subject to delays, disturbances, and measurement noise. The selected experiments were designed to progressively stress the controller through step disturbances, stochastic noise, and demanding reference variations, providing a comprehensive assessment of its closed-loop performance.

For the distillation column case study, the results demonstrate that FDEM achieves accurate reference tracking for both top and bottom compositions while maintaining smooth and bounded control actions. The controller effectively handles the strong input–output interactions and inherent time delays characteristic of distillation processes. Robustness is explicitly evaluated through step disturbances in feed flow rate and feed composition, as well as through the injection of measurement noise. In all cases, the closed-loop system remains stable and returns to the desired operating point without retuning the controller parameters. The limited performance degradation observed under these perturbations confirms the ability of the fuzzy modeling layer to capture nonlinear behavior across a wide operating range, while the DEM-based predictive control law ensures consistent dynamic performance.

In the thermal mixing tank case study, the effectiveness of FDEMM becomes particularly evident under rapidly varying reference signals. The controller successfully tracks both step and ramp changes in flow rate and temperature, even in the presence of significant transport delays and nonlinear process dynamics. Ramp inputs represent a more demanding scenario than step changes, as they continuously excite the system dynamics; nevertheless, FDEMM maintains stable operation and acceptable tracking accuracy. When future reference information is included within the prediction horizon, the controller exhibits anticipative behavior, leading to reduced tracking error and improved transient response. The degradation observed when shortening the prediction horizon confirms the expected trade-off between control performance and computational effort, further validating the predictive nature of the proposed approach.

Compared to its structural prototype DEMM [22], FDEMM significantly broadens the applicability of the matrix-based predictive control framework to nonlinear operating regions without sacrificing the computational efficiency that makes DEMM attractive for real-time use. Unlike many T-S-based MPC strategies that rely on solving complex optimization problems or Linear Matrix Inequalities (LMIs) online, FDEMM employs a generalized fuzzy model with overlapping membership functions and offline-computed control gains. As a result, the online implementation is limited to weight evaluation and linear combinations, yielding an interpretable and computationally efficient control strategy suitable for real-time applications.

Despite these advantages, some limitations should be acknowledged. The performance of FDEMM depends on the selection of MFs, the accuracy of the initial T-S model identification, the prediction horizon, and the weighting matrices, which currently require engineering judgment or empirical tuning. Moreover, although the number of fuzzy rules was kept moderate in the presented case studies, extending the approach to very large-scale systems may require careful rule-based design to balance modeling accuracy and computational effort. Addressing these aspects through automated tuning or adaptive fuzzy partitioning constitutes an interesting direction for future work.

Overall, the presented results confirm that FDEMM provides a robust and effective predictive control solution, capable of handling disturbances, measurement noise, and demanding reference trajectories, while maintaining a favorable compromise between modeling accuracy, control performance, and computational efficiency. This makes FDEMM a promising candidate for nonlinear multivariable control problems where real-time implementation is required.

6. Conclusions

This work presented an FDEMM predictive control framework that combines the computational efficiency of DEMM with the modeling flexibility of T-S fuzzy systems. By embedding a generalized T-S fuzzy model into the DEMM structure and employing an LQR-based predictive control law, the proposed approach extends linear DEMM to nonlinear multivariable systems without significantly increasing online computational complexity.

The main contribution of FDEMM lies in its ability to handle nonlinear dynamics through a convex combination of local linear models while preserving the lightweight structure of DEMM. Compared with classical DMC, FDEMM avoids step response modeling and large online matrix operations, and compared with standard DEMM, it enables accurate prediction and control over a wider operating range. In contrast to many existing T-S-based MPC schemes, the proposed framework relies on offline computation of control gains and simple online weighting, making it particularly well suited for real-time implementation.

The effectiveness and robustness of FDEMM were demonstrated through two representative nonlinear case studies designed to evaluate performance under disturbances, measurement noise, and demanding reference trajectories. In the distillation column example, the controller achieved stable and accurate regulation of top and bottom compositions under input flux disturbances, feed composition step changes, and injected measurement noise, while maintaining smooth and bounded control actions. In the thermal mixing tank system, FDEMM successfully tracked both step and ramp reference changes in flow rate and temperature, even in the presence of significant time delays and nonlinear process dynamics. Ramp inputs, which continuously excite the system dynamics, were handled without loss of stability, confirming the controller's robustness under challenging operating conditions. Moreover, the inclusion of future reference information within the prediction horizon resulted in improved transient response and reduced tracking error, highlighting the anticipative nature of the proposed predictive strategy.

From a practical perspective, FDEMM is particularly suitable for industrial processes characterized by nonlinear behavior, time delays, and multivariable interactions, such as chemical and petrochemical processes, thermal systems, energy systems, and fluid transport applications, where robustness to disturbances and real-time feasibility are essential.

Despite these advantages, some limitations should be noted. Control performance depends on the selection of fuzzy MFs, the prediction horizon, and weighting matrices, which currently rely on engineering judgment or empirical tuning. In addition, although the number of fuzzy rules was kept moderate in the presented applications, extending the approach to very large-scale systems may require careful rule-based design to balance modeling accuracy and computational effort.

Future research will focus on automated and adaptive tuning of membership functions and weighting parameters, systematic robustness analysis under broader disturbance scenarios, and experimental validation on real-time industrial platforms. Extensions of the proposed framework to adaptive and learning-based fuzzy predictive control structures also represent promising directions for further investigation.

Author Contributions: Conceptualization, B.M.A.-H. and A.J.; methodology, B.M.A.-H. and A.J.; software, A.J.; validation, B.M.A.-H. and A.J.; formal analysis, B.M.A.-H. and A.J.; investigation, B.M.A.-H. and A.J.; resources, B.M.A.-H. and A.J.; original draft preparation, B.M.A.-H. and A.J.; review and editing, B.M.A.-H., J.B.R. and A.J.; visualization, B.M.A.-H., J.B.R. and A.J.; supervision, B.M.A.-H., A.J. and A.J.; project administration, B.M.A.-H. All authors have read and agreed to the published version of the manuscript.

Funding: This work is a result of grant PID2024-155948OB-C54 funded by MICIU/AEI/10.13039/501100011033 and ERDF/EU.

Data Availability Statement: The original contributions presented in the study are included in the article; further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Schwenzer, M.; Ay, M.; Bergs, T.; Abel, D. Review on model predictive control: An engineering perspective. *Int. J. Adv. Manuf. Technol.* **2021**, *117*, 1327–1349. <https://doi.org/10.1007/s00170-021-07682-3>.
2. Yao, Y.; Shekhar, D. K. State of the art review on model predictive control (MPC) in Heating Ventilation and Air-conditioning (HVAC) field. *Build. Environ.* **2021**, *200*, 107952. <https://doi.org/10.1016/j.buildenv.2021.107952>.
3. Wang, Y.; Cheng, Y.; Xiong, Y.; Yan, Q. Estimation of battery open-circuit voltage and state of charge based on dynamic matrix control-extended Kalman filter algorithm. *J. Energy Storage* **2022**, *52*, 104860. <https://doi.org/10.1016/j.est.2022.104860>.
4. Sun, Z.; Wu, B.; Wang, D.; Chen, J. Event-triggered model predictive control of spacecraft formation. *IEEE Trans. Autom. Sci. Eng.* **2024**, *22*, 7696–7711. <https://doi.org/10.1109/TASE.2024.3468459>.

5. Qi, X.; Zhang, L.; Wang, P.; Yang, J.; Zou, T.; Liu, W. Learning-based mpc for autonomous motion planning at freeway off-ramp diverging. *IEEE Trans. Intell. Veh.* **2024**, *9*, 6183–6194. <https://doi.org/10.1109/TIV.2024.3363239>.
6. Dubay, S.; Pan, Y. J. Distributed dynamic matrix control with constrained optimization for collision and obstacle avoidance of simulated multiple quadcopters. *Asian J. Control* **2023**, *25*, 4183–4200. <https://doi.org/10.1002/asjc.3120>.
7. Takagi, T.; Sugeno, M. Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans. Syst. Man, Cybern.* **1985**, *SMC-15*, 116–132. <https://doi.org/10.1109/TSMC.1985.6313399>.
8. Yan, W.; Zhao, T.; Niu, B.; Wang, X. Adaptive TS fuzzy control for an unknown structure system with a self-adjusting control accuracy. *IEEE Trans. Autom. Sci. Eng.* **2024**, *22*, 944–957. <https://doi.org/10.1109/TASE.2024.3356752>.
9. Hao, L.-Y.; Zhang, H.; Li, T.-S.; Lin, B.; Chen, C.-P. Fault tolerant control for dynamic positioning of unmanned marine vehicles based on TS fuzzy model with unknown membership functions. *IEEE Trans. Veh. Technol.* **2021**, *70*, 146–157. <https://doi.org/10.1109/TVT.2021.3050044>.
10. Ning, J.; Hua, C. $\mathcal{H}_\infty$ output feedback control for fractional-order TS fuzzy model with time-delay. *Appl. Math. Comput.* **2022**, *416*, 126736. <https://doi.org/10.1016/j.amc.2021.126736>.
11. Li, Y.; Wang, M.; Gu, J. An Approach to Finite-Time $\mathcal{H}_\infty$ Control of TS Fuzzy Systems via Piecewise Lyapunov Functions. In Proceedings of the International Conference on Intelligent Autonomous Systems, Qinhuangdao, China, 22–24 September 2023; pp. 291–296. <https://doi.org/10.1109/ICoIAS61634.2023.00055>.
12. Sunori, S. K.; Mittal, A.; Sharma, S.; Saxena, A. K.; Nainwal, D.; Juneja, P. Controller Design with Dead Time Compensation for Distillation Column. In Proceedings of the International Conference on Communication and Electronics Systems, Coimbatore, India, 22–24 June 2022; pp. 16–20. <https://doi.org/10.1109/ICCES54183.2022.9835789>.
13. Kwon, H.; Oh, K. C.; Choi, Y.; Chung, Y. G.; Kim, J. Development and application of machine learning-based prediction model for distillation column. *Int. J. Intell. Syst.* **2021**, *36*, 1970–1997. <https://doi.org/10.1002/int.22368>.
14. He, W.; Pan, Y.; Bo, C. Iterative Learning Predictive Control Method Based on Reinforcement Learning for Rubber Mixing Batch Process. *IEEE Trans. Autom. Sci. Eng.* **2025**, *22*, 19266–19278. <https://doi.org/10.1109/TASE.2025.3591540>.
15. Herra, E. V.; Vásquez, M. C. A DMC implementation for a binary distillation column: A MIMO approach. In Proceedings of the IEEE Biennial Congress of Argentina, San Juan, Argentina, 7–9 September 2022; pp. 1–5. <https://doi.org/10.1109/ARGENCON55245.2022.9939777>.
16. Sagi, B.; Thyagarajan, T. Model Independent Dynamic Predictive Controller Design Using Differential Extreme Learning Machine for Composition Control in Binary Distillation Column. *Int. J. Adapt. Control Signal Process.* **2025**, *39*, 332–343. <https://doi.org/10.1002/acs.3940>.
17. Lou, W.; Xie, B.; Aubril, J.; Fan, Y.; Luo, L.; Arrivé, A. Optimized flow distributor for stabilized thermal stratification in a single-medium thermocline storage tank: A numerical and experimental study. *Energy* **2023**, *263*, 125709. <https://doi.org/10.1016/j.energy.2022.125709>.
18. Moazeni, F.; Khazaei, J.; Asrari, A. Step towards energy-water smart microgrids; buildings thermal energy and water demand management embedded in economic dispatch. *IEEE Trans. Smart Grid* **2021**, *12*, 3680–3691. <https://doi.org/10.1109/TSG.2021.3068053>.
19. Yamamoto, S.; Furukakoi, M.; Uehara, A.; Takahashi, H.; Senjyu, T. Optimal Operation of Smart House Considering Water Temperature in Hot Water Storage Tank. In Proceedings of the International Conference on Power and Energy Systems Engineering, Nara City, Japan, 6–8 September 2024; pp. 73–78. <https://doi.org/10.1109/CPESE62584.2024.10841272>.
20. Tunjung, D.; Prajitno, P.; Handoko, D. Temperature and water level control system in water thermal mixing process using adaptive fuzzy PID controller. *J. Phys. Conf. Ser.* **2021**, *1816*, 012032. <https://doi.org/10.1088/1742-6596/1816/1/012032>.
21. Peng, Q.; Rockstroh, T.; Pamminger, M.; Hall, C. Model predictive control of mixing controlled compression ignition operation for low reactivity fuels. *Control Eng. Pract.* **2023**, *139*, 105631. <https://doi.org/10.1016/j.conengprac.2023.105631>.
22. Avello, A.J.; Al-Hadithi, B.M.; Garcia, M.I.G.; Rubio, J.M.L. Difference equation matrix model (DEMM) for the control of wind turbines. *Wind Energy* **2014**, *17*, 57–74. <https://doi.org/10.1002/we.1556>.
23. Al-Hadithi, B.M.; Jiménez, A.; Matía, F. New methods for the estimation of Takagi–Sugeno model based extended Kalman filter and its applications to optimal control for nonlinear systems. *Optimal Control Applications and Methods*, **2012**, *33*, 552–575. <https://doi.org/10.1002/oca.1014>.
24. Al-Hadithi, B.M.; Jiménez, A.; Matía, F. A new approach to fuzzy estimation of Takagi–Sugeno model and its applications to optimal control for nonlinear systems. *Appl. Soft Comput.* **2012**, *12*, 280–290. <https://doi.org/10.1016/j.asoc.2011.08.044>.

25. Richter, H.; Barbieri, E.; Figueroa, F. Modelling and validation of a propellant mixer for controller design. *Appl. Math. Model.* **2005**, *29*, 195–210. <https://doi.org/10.1016/j.apm.2004.07.012>.
26. Richter, H. Robust nonlinear control of a thermal mixing process. In Proceedings of the International Conference on Control Applications, Taipei, China, 2–4 September 2004; Volume 2, pp. 1376–1381. <https://doi.org/10.1109/CCA.2004.1387566>.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.