An International Journal of Optimization and Control: Theories & Applications
ISSN: 2146-0957 eISSN: 2146-5703
Vol.x, No.x, pp.xx-xx (2026)
https://doi.org/10.36922/IJOCTA026330214

RESEARCH ARTICLE

Adaptive MIMO fuzzy PID controller based on Runge-Kutta model

İsmail Koray Gündoğan[1*] and Kemal Uçak[2,3*]

[1]Department of Mechatronics Engineering, Istanbul Technical University, İstanbul, Türkiye
[2]Department of Control and Automation Engineering, Istanbul Technical University, İstanbul, Türkiye
[3]Network-based and Data-Driven Control Systems Laboratory, Istanbul Technical University, İstanbul, Türkiye
gundogan23@itu.edu.tr; ucakk@itu.edu.tr

ARTICLE INFO

Article history:
Received: August 14, 2026
Revised: October 10, 2026
Accepted: October 15, 2026
Published Online: October 22, 2026

Keywords:
Adaptive control
Levenberg-Marquardt
Lyapunov stability
MIMO fuzzy PID controller
Parameter estimation
Runge–Kutta model

AMS Classification 2010:
93C10; 93C40; 93C42; 65L06

ABSTRACT

Nonlinear multi-input multi-output (MIMO) systems are difficult to control because of input–output coupling, measurement noise, and parameter variations. This study proposes a Runge–Kutta (RK) model-based predictive adaptive MIMO fuzzy proportional–integral–derivative (PID) controller for nonlinear dynamical systems. A fourth-order Runge–Kutta (RK4) model predicts future outputs and provides the sensitivity information required to update the fuzzy PID scaling parameters online using the Levenberg–Marquardt (LM) algorithm. An additive correction mechanism provides supplementary control action during transient conditions and plant–model mismatch. Furthermore, a discrete-time Lyapunov-based stability analysis is established, guaranteeing the uniform ultimate boundedness (UUB) of the tracking errors and the boundedness of adaptive parameters under aggregate disturbances. The proposed controller is evaluated on a nonlinear three-tank system under nominal conditions, measurement noise, and time-varying parametric uncertainty. For comparative evaluation, an existing RK model-based adaptive MIMO PID controller is employed as a benchmark under the same operating scenarios. Both controllers maintain bounded tracking responses, control inputs, and adaptive parameters. The composite performance-index values obtained with the proposed fuzzy PID controller are 0.2660, 0.2701, and 1.0739, corresponding to improvements of 6.11%, 6.12%, and 3.59%, respectively, over the benchmark controller. In the parametric-uncertainty case, the RK-based parameter estimator follows the variations in the uncertain inter-tank flow coefficient and reduces the resulting plant–model mismatch. Although the benchmark adaptive PID controller has lower computational requirements, the average and maximum cycle times of both controllers remain substantially below the sampling period. The results demonstrate that the proposed adaptive MIMO fuzzy PID controller improves the combined tracking and control-variation performance while retaining computational feasibility.

(cc) BY

1. Introduction

Nonlinear MIMO systems are encountered in various industrial and engineering applications, including interconnected liquid-level processes, chemical reactors, and electromechanical systems. The control of such systems is challenging because their manipulated inputs and controlled outputs are dynamically coupled. Consequently, a variation in one control input may affect several controlled outputs, and the control action generated for one output may influence the behavior of the remaining output channels. Such nonlinear interactions have been demonstrated in interconnected three-tank systems, Van de Vusse chemical reactors, magnetic-levitation systems, and other nonlinear MIMO control applications.[1–3] In addition to input–output coupling, measurement

*Corresponding Authors

noise, modeling inaccuracies, disturbances, and variations in plant parameters may significantly deteriorate closed-loop control performance.[1,2,4] Therefore, adaptive control mechanisms capable of modifying the controller parameters according to changes in the plant dynamics and operating conditions are required.

PID-type controllers are commonly preferred in industrial control applications because of their simple structure, relatively low computational requirements, and straightforward implementation.[2,5] Nevertheless, the performance of a PID controller strongly depends on the proper selection of its PID parameters, and the simultaneous satisfaction of potentially conflicting transient-response, stability, and robustness requirements makes PID tuning a nontrivial design problem.[5] Fixed controller parameters may provide satisfactory performance only within a limited operating region and may become inadequate when the operating point, reference trajectory, plant parameters, or nonlinear system dynamics change. Adaptive control addresses this limitation by updating the controller or model parameters online according to the observed behavior of the controlled system.[2,6] Such adaptation is particularly relevant in the presence of plant-parameter uncertainty and changing operating conditions. However, adaptation mechanisms relying only on current or previously measured tracking errors cannot explicitly consider the future effects of the generated control signals.

In recent literature, additional progress has been made in understanding and designing PID-type controllers for uncertain nonlinear systems from a theoretical perspective. Zhao and Guo[7] derived explicit regions of stabilizing PID parameters for a particular class of second-order MIMO non-affine uncertain nonlinear systems, which led to the achievement of global stability and exponential regulation of the closed-loop system with some upper limits of the system Jacobian and a positive lower limit of control effectiveness. Zhao[8] extended the previous work by considering semiglobal stabilization and asymptotic regulation when the nonlinearities of the systems are allowed to be super-linear with respect to the state variables and introduced a PD controller with prescribed steady-state compensation for known required input at an equilibrium point. Zhao *et al.*[9] studied the limitation of classical constant-gain PID, PD, and PI controllers and designed robust nonlinear PID-type controllers with state- or error-dependent gains for classes of highly nonlinear uncertain systems not capable of global stability by classical PID control. A comprehensive overview of recent nonlinear PID structures, adaptive tuning methods, intelligent optimization approaches, and applications is provided in.[10]

Fuzzy control provides a useful alternative for nonlinear systems by generating the control action through a nonlinear rule-based mapping rather than a fixed linear input–output relationship. Qiao and Mizumoto analyzed a two-input product-sum crisp-type fuzzy controller and showed that its local behavior resembles that of a time-varying PD controller.[11] To eliminate the steady-state error associated with the PD-type structure, they introduced PI- and PID-type fuzzy controllers by incorporating integral action into the fuzzy -controller output. They also developed a peak-observer-based adaptation mechanism to adjust selected input and output scaling parameters according to the extrema observed in the tracking-error response.[11]

Several fuzzy PID structures and design methodologies have subsequently been investigated. Mann *et al.* analyzed different direct-action fuzzy PID structures and examined their equivalent linear and nonlinear PID characteristics.[12] Carvajal *et al.* developed an analytically implementable nonlinear fuzzy PID controller and evaluated its control performance and bounded-input bounded-output stability properties.[13] These studies indicate that fuzzy inference can be combined with PID-type control characteristics to obtain nonlinear controller structures suitable for systems that cannot be adequately controlled using fixed linear relationships.

The behavior of a fuzzy PID controller can be tuned through the scaling factors of its input and output values. Input scaling factors define the range of operation of the tracking error and difference of errors for which the signals are fed into the fuzzy inference engine. Output scaling factors define the magnitude of contribution in terms of PID control actions. Thus, tuning of these factors online becomes an effective way to tune the behavior of the fuzzy controller.

Different adaptation mechanisms have been proposed to adjust the scaling parameters of fuzzy PID controllers. Woo *et al.* introduced a PID-type fuzzy controller with self-tuning scaling factors, in which the scaling parameters are adjusted online to improve both transient and steady-state control performance.[14] This study demonstrated the importance of scaling-parameter adaptation but did not employ an explicit multi-step plant prediction model in the parameter-update mechanism. Uçak and Öke introduced an adaptive fuzzy PID controller based on an online least-squares

support vector regression model.[15] In their structure, the online regression model predicts the future behavior of the nonlinear plant and approximates the system Jacobian required for parameter adaptation. The input and output scaling parameters of the fuzzy PID controller are then adjusted by minimizing a multi-step prediction-error objective through the LM algorithm. Although this structure incorporates predictive adaptation into fuzzy PID control, it is formulated for a single-input single-output (SISO) system and employs a data-driven regression model rather than the known nonlinear differential equations of the plant.

Uçak and Arslantürk extended the peak-observer-based scaling-parameter adaptation mechanism from SISO systems to MIMO systems.[16] The proposed MIMO fuzzy PID controller is composed of main and coupling fuzzy subcontrollers, and the input and output scaling parameters of these subcontrollers are updated according to the peak values observed in the tracking-error responses. In this manner, both direct control contributions and cross-coupling effects are considered. However, the adaptation mechanism is activated according to previously observed tracking-error extrema and does not explicitly incorporate predictions of the future plant behavior.

Adaptive fuzzy and adaptive fuzzy PID control structures have been continuously studied by recent researchers on nonlinear and multivariable systems. Zhou and Wang[17] designed a fuzzy command filtered adaptive controller for constrained nonlinear MIMO systems and proved the closed-loop signals' semiglobal uniform ultimate boundedness. In the case of interacting liquid level systems, Puviyarasi and Murukesh[18] introduced an adaptive mixed fuzzy controller with distinct direct and coupling controllers.

Other recent hybrid designs include an adaptive fuzzy finite-time PID backstepping controller for uncertain nonlinear systems with unmodeled dynamics[19] and a fuzzy-augmented model-reference adaptive PID controller with online adjustment of its adaptation rates.[20] These studies confirm the continuing interest in adaptive fuzzy PID control while also illustrating that the underlying adaptation mechanisms, control architectures, and stability objectives vary considerably among existing approaches, ranging from finite-time backstepping-based designs to model-reference adaptive frameworks with fuzzy online gain regulation.

Model-based predictive adaptation provides a systematic way of accounting for the future consequences of the applied control signal. In this approach, a dynamical model of the controlled system is propagated over a finite prediction horizon to estimate future outputs, construct a multi-step tracking objective, and determine the control or parameter adjustments through model-based optimization.[21] The prediction objective can incorporate both future tracking errors and penalties on excessive control variations, thereby allowing tracking performance and control effort to be treated within a unified optimization framework.[2,21] For continuous-time nonlinear systems represented by known differential equations, numerical integration offers a direct means of constructing a discrete-time prediction model while preserving the nonlinear physical structure of the plant.[1]

In recent works on adaptive predictive control, model mismatch and time-varying parameter issues have been resolved using different model representations. Chen *et al.*[22] presented an adaptive Koopman operator-based MPC technique, wherein the nonlinear system model is updated online by accounting for the unknown time-varying parameters. Asvadi-Kermani *et al.*[23] used the LM-based online model updates in the context of adaptive sliding mode predictive control. Unlike these approaches using data-based or hybrid model updates, the current approach involves propagation of the plant equations through the RK4 algorithm with the resultant analytical sensitivities used for updating the fuzzy PID tuning parameters.

İplikçi introduced an RK model-based adaptive predictive control mechanism for nonlinear continuous-time processes.[1] In this method, the classical RK4 algorithm is used to construct a discrete-time approximation of the nonlinear plant dynamics. The resulting RK model is employed for multi-step output prediction and derivative calculation. The framework also includes an extended Kalman filter (EKF) based on the RK model for online state estimation and an RK-based estimator for unknown or varying plant parameters. The effectiveness of the method was demonstrated on nonlinear MIMO processes, including an interconnected three-tank system and a Van de Vusse chemical reactor.[1]

Çetin and İplikçi extended the RK model-based predictive framework to the online adjustment of PID-controller parameters.[2] In their RK-PID mechanism, the RK model generates multi-step future output predictions, and the corresponding

prediction errors are used to construct a finite-horizon objective function. The sensitivities of the predicted outputs with respect to the control signals and PID parameters are calculated through the RK model. The PID parameters are then updated using the LM algorithm. An additive correction mechanism is also incorporated to supplement the PID control action when the currently adjusted controller parameters are inadequate because of changes in the reference trajectory, disturbances, or modeling errors.[2] The proposed method was experimentally evaluated on both SISO and MIMO nonlinear systems.

RK structures have also been combined with intelligent nonlinear controllers. Uçak proposed a model-predictive RK neural-network controller for nonlinear MIMO systems.[3] In this structure, the nonlinear learning capability of a radial-basis-function neural network is integrated with RK-based state propagation, while a system model predicts the future behavior of the controlled plant. The controller parameters are adjusted according to a multi-step prediction-error objective. The results demonstrate that RK-based prediction and intelligent nonlinear control structures can be integrated for the control of coupled nonlinear MIMO systems.[3]

Another direct application of RK integration within adaptive control is due to Uçak,[24] where, in addition to using RK integration for plant modeling, the RK4 discretization of a certain ODE was used to parametrize the control law. Inasmuch as the controller function itself was not known a priori, the stage-slopes obtained through RK integration were used as parameters of the controller. The predictor model based on RK integration was applied to obtain finite-–horizon predictions and their corresponding system sensitivities. Using such predictions, the LM algorithm updated the controller parameters, minimizing an objective that includes multi-step tracking errors and control increments. Therefore, RK integration was used both within the controller and predictor models, with the latter also supplying Jacobians needed for the controller update.

In adaptive control, a critical fundamental step toward achieving accurate control-system performance is to obtain an accurate system model. For this purpose, data-driven, LM-based neural network, and physics-informed neural network structures have been proposed in the literature for estimating system dynamics by considering the system ODEs and input–output data. Haring *et al.*[25] proposed a Levenberg–Marquardt-based method for the sparse identification of nonlinear dynamical systems described by ordinary differential equations. Waseem *et al.*[26] employed a physics-informed neural network approach for the data-driven solution of a nonlinear infectious-disease model. Khan *et al.*[27] employed a neural network trained using the Levenberg–Marquardt back-propagation algorithm for the numerical solution of a nonlinear electromagnetohydrodynamic bio-convection flow model. These studies represent selected examples from a broader body of related literature.

In contrast to the above-mentioned approaches, this research uses the nonlinear dynamical equations of the plant in the RK4 prediction model and employs the LM algorithm for online tuning of the scaling factors of the MIMO fuzzy PID controller based on multi-step prediction errors.

The existing literature therefore presents three complementary research directions: RK model-based multi-step prediction, state estimation, parameter estimation, and PID tuning;[1,2] prediction-model-based adaptation of fuzzy PID scaling parameters;[15] and adaptive MIMO fuzzy PID control based on direct and cross-coupling subcontrollers.[16] Nevertheless, these components have not been integrated into a unified architecture in which an RK model constructed from the known nonlinear plant equations provides multi-step predictions and structured sensitivity information for the online adjustment of the input and output scaling parameters of a MIMO fuzzy PID controller.

To address this need, this study proposes an RK model-based predictive adaptive MIMO fuzzy PID controller for nonlinear and coupled systems. In the RK-based framework introduced by İplikçi[1] and subsequently extended by Çetin and İplikçi for predictive PID tuning,[2] the RK model is used to generate future output predictions and the sensitivity information required for controller-parameter optimization. The complete RK framework is structured around three fundamental subblocks: the $\mathrm{RK_{EKF}}$, $\mathrm{RK_{estimator}}$, and the raw $\mathrm{RK_{model}}$. Specifically, the $\mathrm{RK_{EKF}}$ estimates the system states from measured input–output data, while the $\mathrm{RK_{estimator}}$ identifies unknown or slowly varying system parameters that cannot be explicitly derived from the mathematical model. Thus, the RK4 model of the controlled system predicts the future outputs over a finite horizon and provides the sensitivity information required for controller adaptation. The input and output scaling parameters of the direct and cross-coupling fuzzy PID subcontrollers are updated online using the LM algorithm.[28] An additive correction mechanism is also included to provide supplementary control action during

initial transients, abrupt reference changes, and plant–model mismatch.

The liquid levels of two tanks are controlled by manipulating their inlet flow rates in the three-tank system.[29] The controller is examined under nominal operation, measurement noise, and time-varying parametric uncertainty.[3,30] In the uncertainty case, the varying inter-tank flow coefficient is estimated online using the RK-based parameter estimator ($\mathrm{RK_{estimator}}$).

For comparative evaluation, the existing RK model-based adaptive MIMO PID controller reported in[2] is employed solely as a benchmark in the simulation studies. Both controllers are evaluated under the same nominal, measurement-noise, and parametric-uncertainty scenarios to compare their control performance and computational requirements.

The main contributions of this study can be summarized as follows:

- To the best of the authors' knowledge, this study presents the first application of the $\mathrm{RK_{model}}$ framework proposed by İplikçi to the design and optimization of a fuzzy PID controller. This work extends the applicability of the original RK framework to intelligent control systems.
- A novel RK-based fuzzy PID controller is developed by integrating the $\mathrm{RK_{EKF}}$, the $\mathrm{RK_{estimator}}$, and the raw $\mathrm{RK_{model}}$ into a unified control framework. The $\mathrm{RK_{EKF}}$ is employed for online state estimation, while the $\mathrm{RK_{estimator}}$ estimates unknown or time-varying system parameters, enabling adaptive tuning of the parameters of the fuzzy PID controller.
- The proposed controller is comprehensively evaluated on the nonlinear three-tank benchmark system under three representative operating scenarios, namely nominal conditions, measurement noise, and parametric uncertainty.
- Extensive simulation results demonstrate that the proposed RK-based fuzzy PID controller achieves accurate reference tracking while maintaining strong robustness against disturbances and model uncertainties, confirming its effectiveness and practical applicability in nonlinear process control.
- The mathematical novelty of the proposed framework lies in replacing the conventional PID structure of the existing RK-based adaptive PID framework with a nonlinear fuzzy PID control structure and formulating the online optimization of its input and output scaling parameters. Specifically, the scaling parameters K_e, K_{de}, α, and β are adapted through the LM algorithm using the prediction and sensitivity information provided by the RK model. This formulation enables the nonlinear behavior of the fuzzy PID controller to be adjusted online according to the predicted closed-loop response.
- The proposed RK-based adaptive MIMO fuzzy PID controller is systematically compared with the previously developed RK-based adaptive MIMO PID controller[2] under identical operating conditions. The comparison evaluates reference-tracking performance, control signals, adaptive parameter variations, correction terms, quantitative performance indices, and computational requirements.
- A discrete-time Lyapunov-based stability analysis is established for the closed-loop system, proving the uniform ultimate boundedness (UUB) of the tracking errors and the boundedness of the parameters in the presence of aggregate disturbances, RK truncation errors, and parameter-estimation mismatch.
- The mathematical novelty of the design is the substitution of the classical PID structure in the existing RK-based adaptive PID framework with a nonlinear fuzzy PID control structure, followed by online optimization of the input and output scaling factors of the control structure. In particular, the scaling factors K_e, K_{de}, α, and β are tuned via the LM algorithm based on the predictions and sensitivity information provided by the RK model.
- A systematic comparison of the newly designed RK-based adaptive MIMO fuzzy PID controller with the existing RK-based adaptive MIMO PID controller[2] is performed under the same operating conditions. The aspects being considered for comparison include reference tracking, control inputs, adaptive parameter changes, correction terms, performance indices, and computational requirements.

The remainder of the paper is structured as follows. Section 2 provides a detailed description of the RK-based system identification framework proposed by İplikçi.[1] Section 3 presents the development of the RK-based adaptation mechanism for the fuzzy PID controller along with its discrete-time Lyapunov stability analysis. The performance of the proposed RK-based fuzzy PID controller is evaluated in Section 4. Finally, Section 5 concludes the paper and outlines potential

directions for future research.

2. Runge–Kutta model-based system identification

This section presents the RK model-based identification framework employed to obtain a discrete-time representation of the continuous-time nonlinear system. The resulting model is used for state estimation and the online identification of uncertain system parameters.[1] This identification structure provides the state, parameter, and model information required for the fuzzy PID control system considered in this study.

Section 2.1 briefly introduces the fundamentals of the RK integration method. Subsequently, Section 2.2 describes the RK-based system identification framework in detail, including its two principal functional modules: the $\mathrm{RK_{EKF}}$ for online state estimation and the $\mathrm{RK_{estimator}}$ for estimating unknown or time-varying system parameters.

2.1. An overview of Runge–Kutta integration

Consider a nonlinear continuous-time system represented by

$$\dot{\mathbf{x}}(t) = \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t), \theta\big), \tag{1}$$

where $\mathbf{x}(t)$ is the state vector, $\mathbf{u}(t)$ represents the input vector, θ denotes an uncertain system parameter, and $\mathbf{f}(\cdot)$ defines the nonlinear system dynamics.[1] A similar continuous-time MIMO representation without an explicit parameter term has also been adopted in.[2] Since estimation and control algorithms are implemented at discrete sampling instants, the RK4-based discretization approach is employed to obtain a suitable discrete-time representation.[1,2]

The classical RK4 method is widely regarded as providing an effective compromise between numerical accuracy and computational complexity, as higher-order methods generally achieve greater accuracy at a correspondingly higher per-step computational cost. Let T_s denote the sampling period. Assuming that the input remains constant over one sampling interval, the RK4 increments are calculated as[1–3]

$$\begin{aligned}
\mathbf{K}_1 &= T_s\mathbf{f}\big(\mathbf{x}[n], \mathbf{u}[n], \theta\big),\\
\mathbf{K}_2 &= T_s\mathbf{f}\Big(\mathbf{x}[n] + \frac{\mathbf{K}_1}{2}, \mathbf{u}[n], \theta\Big),\\
\mathbf{K}_3 &= T_s\mathbf{f}\Big(\mathbf{x}[n] + \frac{\mathbf{K}_2}{2}, \mathbf{u}[n], \theta\Big),\\
\mathbf{K}_4 &= T_s\mathbf{f}\Big(\mathbf{x}[n] + \mathbf{K}_3, \mathbf{u}[n], \theta\Big).
\end{aligned} \tag{2}$$

The one-step-ahead state is then approximated by[1–3]

$$\begin{aligned}
\mathbf{x}[n+1] = \mathbf{x}[n] + \frac{1}{6}\Big(&\mathbf{K}_1 + 2\mathbf{K}_2\\
&+ 2\mathbf{K}_3 + \mathbf{K}_4\Big).
\end{aligned} \tag{3}$$

Accordingly, the RK4-based discrete-time transition model can be written in compact form as:[1,3,30]

$$\mathbf{x}[n+1] = \mathbf{f}_{\mathrm{RK}}\big(\mathbf{x}[n], \mathbf{u}[n], \theta\big), \tag{4}$$

where $\mathbf{f}_{\mathrm{RK}}(\cdot)$ denotes the nonlinear state transition obtained through RK4 integration.[1] This representation forms the basis of the state- and parameter-estimation procedures used in the identification framework.[1,3,30]

2.2. System identification based on Runge–Kutta

The overall structure of the RK model-based system identification framework is illustrated in **Figure 1**. The framework consists of three principal components: the RK prediction model ($\mathrm{RK_{model}}$), the RK model-based extended Kalman filter ($\mathrm{RK_{EKF}}$), and the RK model-based online parameter estimator ($\mathrm{RK_{estimator}}$).[3,30]

The available input-output measurements are processed by $\mathrm{RK_{EKF}}$ to reconstruct the system states that are not directly measured. The corrected state estimate is supplied to both $\mathrm{RK_{model}}$ and $\mathrm{RK_{estimator}}$. Using the corrected state, the applied control input, and the current parameter estimate, $\mathrm{RK_{model}}$ generates a one-step-ahead state prediction. The estimator subsequently compares this prediction with the corrected state available at the next sampling instant and updates the uncertain parameter according to the resulting state-prediction error.

2.2.1. MIMO systems and the Runge–Kutta method

Consider a continuous-time nonlinear MIMO system with N states, R inputs, and Q outputs:[1,3,30]

$$\begin{aligned}
\dot{\mathbf{x}}(t) &= \mathbf{f}\big(\mathbf{x}(t), \mathbf{u}(t), \theta\big),\\
\mathbf{y}(t) &= \mathbf{g}\big(\mathbf{x}(t), \mathbf{u}(t)\big),\\
\mathbf{x}(t) &= \begin{bmatrix} x_1(t) & \cdots & x_N(t)\end{bmatrix}^{\mathrm{T}},\\
\mathbf{u}(t) &= \begin{bmatrix} u_1(t) & \cdots & u_R(t)\end{bmatrix}^{\mathrm{T}},\\
\mathbf{y}(t) &= \begin{bmatrix} y_1(t) & \cdots & y_Q(t)\end{bmatrix}^{\mathrm{T}}.
\end{aligned} \tag{5}$$

The nonlinear state and output mappings $\mathbf{f}(\cdot)$ and $\mathbf{g}(\cdot)$ are assumed to be known and continuously differentiable with respect to their relevant arguments.[1] This modeling assumption enables the analytical evaluation of the Jacobian matrices required by $\mathrm{RK_{EKF}}$ and $\mathrm{RK_{estimator}}$.[1,3,30]

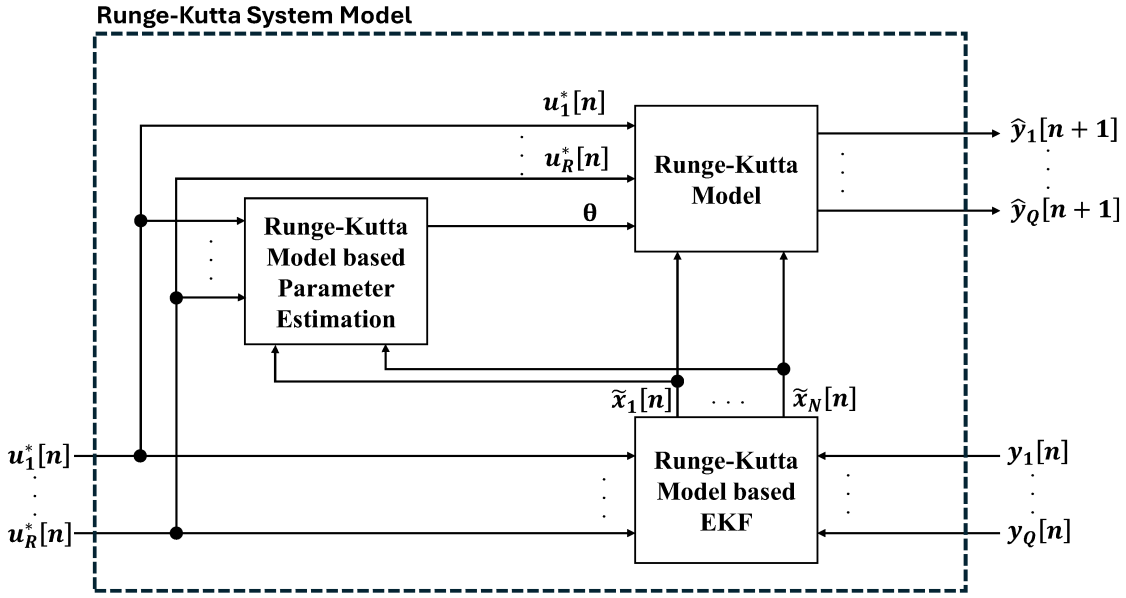


Figure 1. General structure of the Runge–Kutta model-based system identification framework, adapted from[1,3,30]

The state estimate corrected by $\mathrm{RK_{EKF}}$ at sampling instant n is represented by

$$\tilde{\mathbf{x}}[n] = \begin{bmatrix} \tilde{x}_1[n] & \tilde{x}_2[n] & \cdots & \tilde{x}_N[n] \end{bmatrix}^{\mathrm{T}}. \tag{6}$$

The corrected state estimate provides the initial condition for the RK prediction model, while the current parameter estimate determines the model dynamics. The nonlinear MIMO formulation of the resulting discrete-time prediction model is presented next.

Applying the RK4 method introduced in Section 2.1 to the nonlinear MIMO system in (5) yields the following one-step-ahead discrete-time prediction model:[1,3,30]

$$\begin{aligned} \hat{\mathbf{x}}[n+1] &= \mathbf{f}_{\mathrm{RK}}\big(\tilde{\mathbf{x}}[n], \mathbf{u}[n], \hat{\theta}[n]\big), \\ \hat{\mathbf{y}}[n+1] &= \mathbf{g}\big(\hat{\mathbf{x}}[n+1], \mathbf{u}[n]\big). \end{aligned} \tag{7}$$

The discrete-time model in (7) preserves the nonlinear state dynamics and the coupled influence of the multiple inputs on the predicted states and outputs. Within the identification framework, this model generates the nonlinear prior state prediction required by $\mathrm{RK_{EKF}}$. It is also used to calculate the state-prediction error and parameter sensitivities required by $\mathrm{RK_{estimator}}$.[1,3,30]

2.2.2. The Runge–Kutta model-based EKF

$\mathrm{RK_{EKF}}$ estimates the unavailable system states by combining the nonlinear state prediction generated by $\mathrm{RK_{model}}$ with the available output measurements. The discrete-time system is given by[1,3,30]

$$\begin{aligned} \mathbf{x}[n] &= \mathbf{f}_{\mathrm{RK}}\big(\mathbf{x}[n-1], \mathbf{u}[n-1], \\ &\qquad \hat{\theta}[n-1]\big) + \mathbf{w}[n-1], \\ \mathbf{y}[n] &= \mathbf{g}\big(\mathbf{x}[n], \mathbf{u}[n-1]\big) + \mathbf{v}[n], \end{aligned} \tag{8}$$

where $\mathbf{w}[n-1]$ and $\mathbf{v}[n]$ denote zero-mean process- and measurement-noise vectors with covariance matrices $\mathbf{Q}$ and $\mathbf{R}$, respectively.[1] The process noise represents modeling errors and unmodeled disturbances, whereas the measurement noise represents uncertainty in the available output measurements.

The recursive operation of $\mathrm{RK_{EKF}}$ consists of two principal stages: the time-update stage and the measurement-update stage. During the time update, $\mathrm{RK_{model}}$ is used to calculate the prior state estimate and its error covariance. During the measurement update, the prior estimate is corrected using the output innovation and the Kalman gain.[1,3,30]

Time-update (prediction) stage: $\mathrm{RK_{EKF}}$ combines the prior state and covariance estimates as:[1,3,30]

$$\begin{aligned} \tilde{\mathbf{x}}^{-}[n] &= \mathbf{f}_{\mathrm{RK}}\big(\tilde{\mathbf{x}}[n-1], \mathbf{u}[n-1], \hat{\theta}[n-1]\big), \\ \mathbf{P}_n^{-} &= \mathbf{A}_n \mathbf{P}_{n-1} \mathbf{A}_n^{\mathrm{T}} + \mathbf{Q}, \end{aligned} \tag{9}$$

where the RK state-transition Jacobian is

$$\mathbf{A}_n = \left.\frac{\partial \mathbf{f}_{\mathrm{RK}}}{\partial \mathbf{x}}\right|_{\substack{\mathbf{x}=\tilde{\mathbf{x}}[n-1] \\ \mathbf{u}=\mathbf{u}[n-1] \\ \theta=\hat{\theta}[n-1]}}. \tag{10}$$

Here, $\mathbf{P}_n^{-}$ denotes the prior state-estimation error covariance, and $\mathbf{A}_n$ describes the local sensitivity of the RK state-transition mapping to the

state vector. **Equations (9)–(10)** constitute the time-update stage of $\mathrm{RK_{EKF}}$.[1,3,30]

Measurement-update stage: The measurement Jacobian is given by[1,3,30]

$$\mathbf{H}_n = \left.\frac{\partial \mathbf{g}}{\partial \mathbf{x}}\right|_{\substack{\mathbf{x}=\tilde{\mathbf{x}}[n-1] \\ \mathbf{u}=\mathbf{u}[n-1]}}. \tag{11}$$

Using $\mathbf{H}_n$ in (11), the measurement-update (correction) stage jointly computes the Kalman gain, the corrected state estimate, and the updated covariance matrix as:[1,3,30]

$$\begin{aligned} \mathbf{K}_n &= \mathbf{P}_n^- \mathbf{H}_n^{\mathrm{T}} \Big(\mathbf{H}_n \mathbf{P}_n^- \mathbf{H}_n^{\mathrm{T}} + \mathbf{R}\Big)^{-1}, \\ \tilde{\mathbf{x}}[n] &= \tilde{\mathbf{x}}^-[n] + \mathbf{K}_n \Big[\mathbf{y}[n] - \mathbf{g}\big(\tilde{\mathbf{x}}^-[n], \mathbf{u}[n-1]\big)\Big], \\ \mathbf{P}_n &= \Big(\mathbf{I} - \mathbf{K}_n \mathbf{H}_n\Big)\mathbf{P}_n^-. \end{aligned} \tag{12}$$

The term enclosed in square brackets in (12) is the output innovation, namely, the difference between the measured output and the output predicted from the prior state estimate. Thus, $\mathrm{RK_{EKF}}$ recursively reconstructs the current state vector by correcting the nonlinear RK prediction according to the available output measurements.[1] The corrected state estimate is subsequently supplied to $\mathrm{RK_{model}}$ and $\mathrm{RK_{estimator}}$.[3,30]

2.2.3. The Runge–Kutta model-based online parameter estimation

The accuracy of the RK prediction model is fundamentally determined by the fidelity of the underlying ordinary differential equation (ODE) model, which, in turn, depends on the accuracy of the physical system parameters. Therefore, $\mathrm{RK_{estimator}}$ updates the uncertain parameter θ online using the state-prediction error, defined as the difference between the state corrected by $\mathrm{RK_{EKF}}$ and the corresponding one-step-ahead state predicted by $\mathrm{RK_{model}}$.[1,3,30] Accordingly, parameter estimation is driven by the state-prediction error rather than directly by the output-tracking error.

The one-step-ahead state prediction used for parameter estimation is obtained from the RK model in (7). The corresponding state-prediction error is defined as[1,3,30]

$$\mathbf{e}_\theta[n] = \tilde{\mathbf{x}}[n+1] - \hat{\mathbf{x}}[n+1]. \tag{13}$$

For the scalar uncertain parameter considered here, the Jacobian for $\mathrm{RK_{estimator}}$ is obtained from the sensitivity of the state-prediction error to θ[3,30]

$$\begin{aligned} \mathbf{J}_\theta[n] &= \frac{\partial \mathbf{e}_\theta[n]}{\partial \theta[n]} \\ &= -\frac{\partial \hat{\mathbf{x}}[n+1]}{\partial \theta}. \end{aligned} \tag{14}$$

In (14), the corrected state $\tilde{\mathbf{x}}[n+1]$ is treated as independent of the current parameter update. Consequently, the Jacobian is determined by the negative sensitivity of the one-step-ahead RK prediction to θ.

Because the RK4 update in (7) advances the state recursively, the sensitivity of the one-step-ahead prediction to θ also depends on the sensitivity of the initializing state $\hat{\mathbf{x}}[n]$ to θ. Accordingly, the RK prediction sensitivity is expressed as[1,3,30]

$$\begin{aligned} \frac{\partial \hat{\mathbf{x}}[n+1]}{\partial \theta[n]} = \frac{\partial \hat{\mathbf{x}}[n]}{\partial \theta[n]} + \frac{1}{6}\Bigg(&\frac{\partial \mathbf{K}_1[n]}{\partial \theta[n]} + 2\frac{\partial \mathbf{K}_2[n]}{\partial \theta[n]} \\ &+ 2\frac{\partial \mathbf{K}_3[n]}{\partial \theta[n]} + \frac{\partial \mathbf{K}_4[n]}{\partial \theta[n]}\Bigg). \end{aligned} \tag{15}$$

Under the independence assumption stated above, the term $\partial \hat{\mathbf{x}}[n]/\partial \theta[n]$ is set to zero, so that (15) reduces to the form used in (14). The individual increment sensitivities $\partial \mathbf{K}_1[n]/\partial \theta[n], \ldots, \partial \mathbf{K}_4[n]/\partial \theta[n]$ are evaluated recursively by applying the chain rule to the four RK4 stages, following the same recursive differentiation strategy used for Jacobian computation in RK-based model-predictive frameworks.[1–3,30]

Using the state-prediction error and its Jacobian, the parameter estimate is updated according to[1,3,30]

$$\hat{\theta}[n+1] = \hat{\theta}[n] - \frac{\mathbf{J}_\theta^{\mathrm{T}}[n]\mathbf{e}_\theta[n]}{\mathbf{J}_\theta^{\mathrm{T}}[n]\mathbf{J}_\theta[n]}. \tag{16}$$

Equation (16) has the structural form of a Gauss–Newton-type correction for the scalar uncertain parameter, in which the Hessian of the least-squares objective is approximated by $\mathbf{J}_\theta^{\mathrm{T}}[n]\mathbf{J}_\theta[n]$.[31] The numerator projects the state-prediction error onto the parameter-sensitivity direction, while the denominator scales the correction by the squared magnitude of this sensitivity.

The recursive parameter update enables the $\mathrm{RK_{model}}$ to continuously adapt to uncertain or time-varying system parameters while reducing the state prediction error.[1–3,30] This adaptive mechanism ensures accurate estimation of the physical system parameters, thereby improving the accuracy of the underlying ODE model.[1–3,30] Consequently, future system trajectories and the corresponding Jacobian matrix can be predicted more precisely.[1–3,30] The updated state estimates provided by the $\mathrm{RK_{EKF}}$, together with the parameter estimates generated by the $\mathrm{RK_{estimator}}$, provide the state and model information required by the RK-based fuzzy PID controller ($\mathrm{RK_{Fuzzy-PID}}$) presented in the following section.

3. Adaptation mechanism for the fuzzy PID controller

This section presents the RK model-based adaptation mechanism developed for the MIMO fuzzy PID controller. First, the nonlinear fuzzy PID control law and its adaptive scaling parameters are formulated in Section 3.1. The predictive adaptation mechanism, which uses the RK output predictions, is then derived in Section 3.2. Finally, Section 3.3 provides a discrete-time Lyapunov stability analysis to prove the uniform ultimate boundedness of tracking errors and parameter boundedness under aggregate modeling and disturbance effects.

3.1. An overview of the fuzzy PID controller

The general structure of the RK model-based adaptive MIMO fuzzy PID controller is illustrated in **Figure 2**. The fuzzy PID block generates the nominal control vector, following the fuzzy PID controller structure of;[15] $\mathrm{RK}_{\mathrm{model}}$ provides the required predictions, as in the RK-model-based prediction framework of;[3,30] and the adaptation mechanism updates the controller parameters online based on the LM optimization algorithm, consistent with.[2,3,30,32]

Consider a nonlinear MIMO system with R control inputs and Q controlled outputs. The tracking error vector is defined as

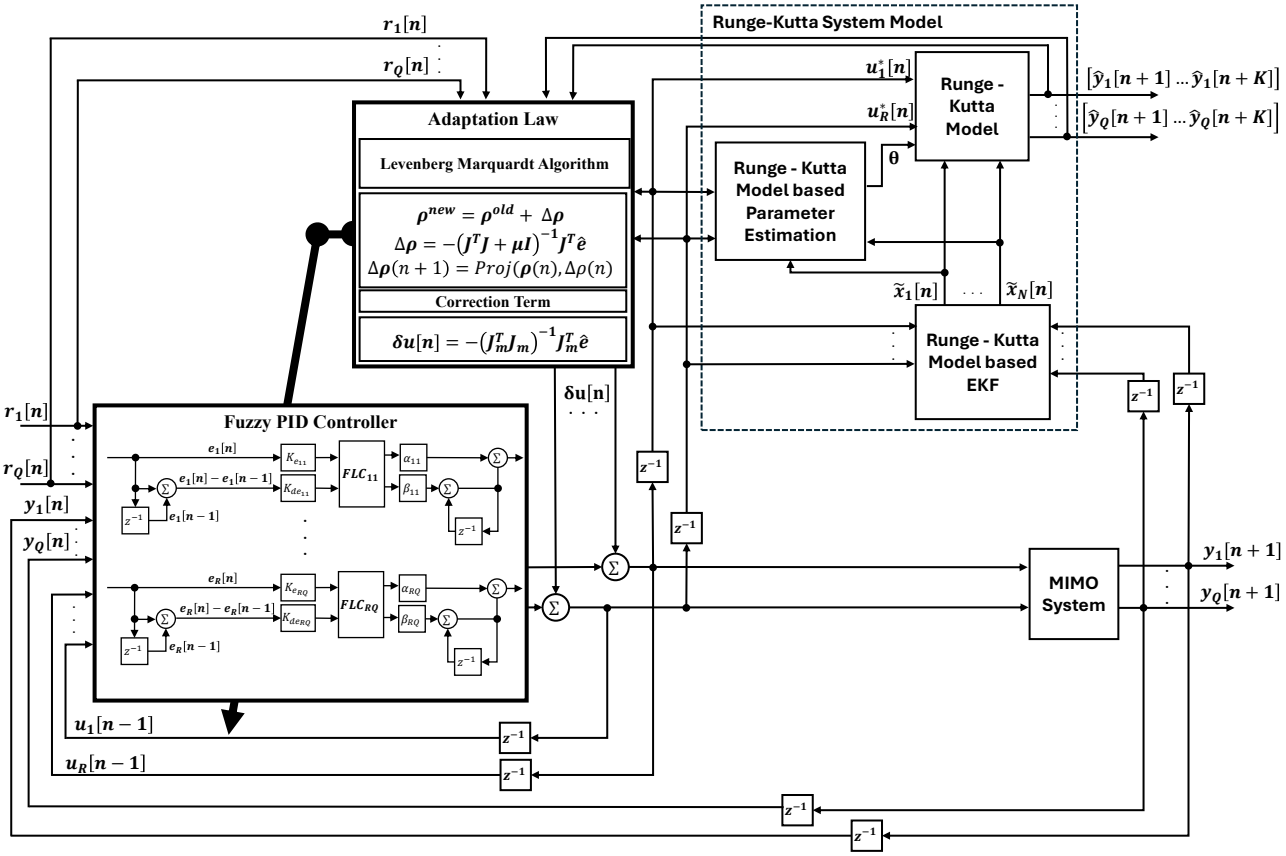


Figure 2. Runge–Kutta model-based adaptive MIMO fuzzy PID controller structure[3,16]

$$\mathbf{e}[n] = \mathbf{y}_{\mathrm{ref}}[n] - \mathbf{y}[n], \tag{17}$$

where $\mathbf{y}_{\mathrm{ref}}[n]$ and $\mathbf{y}[n]$ denote the reference and measured output vectors, respectively. The tracking error and its discrete-time difference associated with the q-th controlled output are defined analogously to the scalar tracking error of,[15] extended to each output channel of the MIMO system, as

$$\begin{aligned} e_q[n] &= y_{ref_q}[n] - y_q[n], \\ \Delta e_q[n] &= e_q[n] - e_q[n-1], \\ q &= 1, \ldots, Q. \end{aligned} \tag{18}$$

For each input–output pair of subcontrollers, the normalized fuzzy controller inputs are calculated by extending the SISO input-scaling formulation of[15] to every (r, q) pair of the MIMO structure, consistent with the multiple-input–multiple-output fuzzy PID architecture of,[16] as

$$\begin{aligned} e_{n,rq}[n] &= K_{e,rq}[n]e_q[n], \\ \dot{e}_{n,rq}[n] &= K_{de,rq}[n]\Delta e_q[n]. \end{aligned} \tag{19}$$

where $K_{e,rq}[n]$ and $K_{de,rq}[n]$ are the input scaling parameters of the fuzzy subcontroller connecting the q-th controlled output to the r-th control input, extending the SISO scaling-coefficient definition of[15] to each (r, q) pair of the MIMO structure introduced in this work. The notation $\dot{e}_{n,rq}[n]$ represents the normalized *discrete-time error difference* rather than a continuous-time derivative. Since the scaling parameters are adapted online, the normalized inputs are saturated to the universe of discourse before defuzzification,

$$\begin{aligned} e_{n,rq}[n] &\leftarrow \mathrm{sat}\big(e_{n,rq}[n]\big), \\ \dot{e}_{n,rq}[n] &\leftarrow \mathrm{sat}\big(\dot{e}_{n,rq}[n]\big), \\ \mathrm{sat}(z) &= \max\big(-1,\ \min(1,\ z)\big), \end{aligned} \tag{20}$$

so that the membership degrees remain well defined during large reference transitions, consistent with the fundamental requirement that fuzzy membership functions be defined over a bounded universe of discourse,[33,34] and with the normalized $[-1, 1]$ universe of discourse adopted for the triangular membership functions in.[15,16] Outside the saturation region, the corresponding derivatives in (19) vanish, which prevents unbounded growth of the input scaling parameters.

Triangular membership functions are defined over the normalized universe of discourse using the core points $(-1, -0.4, 0, 0.4, 1)$, extending the SISO core-point definition of[15] to each fuzzy subcontroller of the MIMO structure, consistent with.[16] Assume that $e_{n,rq}[n]$ lies between the neighboring core points e_i and e_{i+1}. The corresponding membership degrees are obtained by extending the SISO membership-degree formulation of[15] to each (r, q) pair, as

$$\begin{aligned} A_i\big(e_{n,rq}[n]\big) &= \frac{e_{i+1} - e_{n,rq}[n]}{e_{i+1} - e_i}, \\ A_{i+1}\big(e_{n,rq}[n]\big) &= \frac{e_{n,rq}[n] - e_i}{e_{i+1} - e_i}. \end{aligned} \tag{21}$$

Similarly, if $\dot{e}_{n,rq}[n]$ lies between the neighboring core points $\dot{e}_j$ and $\dot{e}_{j+1}$, the membership degrees are, following the same SISO-to-MIMO extension of,[15]

$$\begin{aligned} B_j\big(\dot{e}_{n,rq}[n]\big) &= \frac{\dot{e}_{j+1} - \dot{e}_{n,rq}[n]}{\dot{e}_{j+1} - \dot{e}_j}, \\ B_{j+1}\big(\dot{e}_{n,rq}[n]\big) &= \frac{\dot{e}_{n,rq}[n] - \dot{e}_j}{\dot{e}_{j+1} - \dot{e}_j}. \end{aligned} \tag{22}$$

Since the triangular membership functions overlap only pairwise, at most two membership functions are active for each fuzzy input, and hence exactly four neighboring rules are fired at any sampling instant, as in the original PID-type fuzzy controller of.[11,16] Furthermore, the neighboring membership degrees sum to unity within each active interval, so that the center-of-gravity defuzzification reduces to the plain weighted sum of the four active rule outputs, extending the SISO defuzzification formulation of[15] to each (r, q) fuzzy subcontroller of the MIMO structure,

$$\begin{aligned} u_{FC,rq}[n] = A_i B_j u_{ij} + A_{i+1} B_j u_{i+1,j} \\ + A_i B_{j+1} u_{i,j+1} + A_{i+1} B_{j+1} u_{i+1,j+1}, \end{aligned} \tag{23}$$

where the arguments of the membership functions have been omitted for compactness. **Equation (23)** represents the weighted interpolation of the four active singleton rule outputs, consistent with.[15]

The rule values are generated from a sigmoid-based nonlinear control surface as given in[15] instead of a conventional rule table in.[11,16] The rule value at the core point $(e_i, \dot{e}_j)$ is defined as[15]

$$u_{ij} = S_i + S_j - 1, \tag{24}$$

where

$$\begin{aligned} S_i &= \frac{1}{1 + \mathrm{e}^{-k_1 e_i}}, \\ S_j &= \frac{1}{1 + \mathrm{e}^{-k_2 \dot{e}_j}}. \end{aligned} \tag{25}$$

The sigmoid slope parameters have been empirically selected as $k_1 = k_2 = 3$[15] based on an observational evaluation of the resulting fuzzy rule surface, as these values provided an appropriate nonlinear characteristic while maintaining smooth transitions between the fuzzy rule regions.

As illustrated in **Figure 3**, the resulting control surface exhibits a smooth, continuous, and non-singular transition across the input space, confirming that the selected sigmoid slope parameters yield a well-behaved fuzzy mapping without abrupt switching discontinuities.

The fuzzy mapping can consequently be represented in compact form as

$$u_{FC,rq}[n] = f_F\big(e_{n,rq}[n], \dot{e}_{n,rq}[n]\big), \tag{26}$$

where $f_F(\cdot)$ represents the nonlinear mapping produced by the membership functions and the sigmoid-based control surface.[15] The subcontroller-wise fuzzy PID control contribution is calculated as

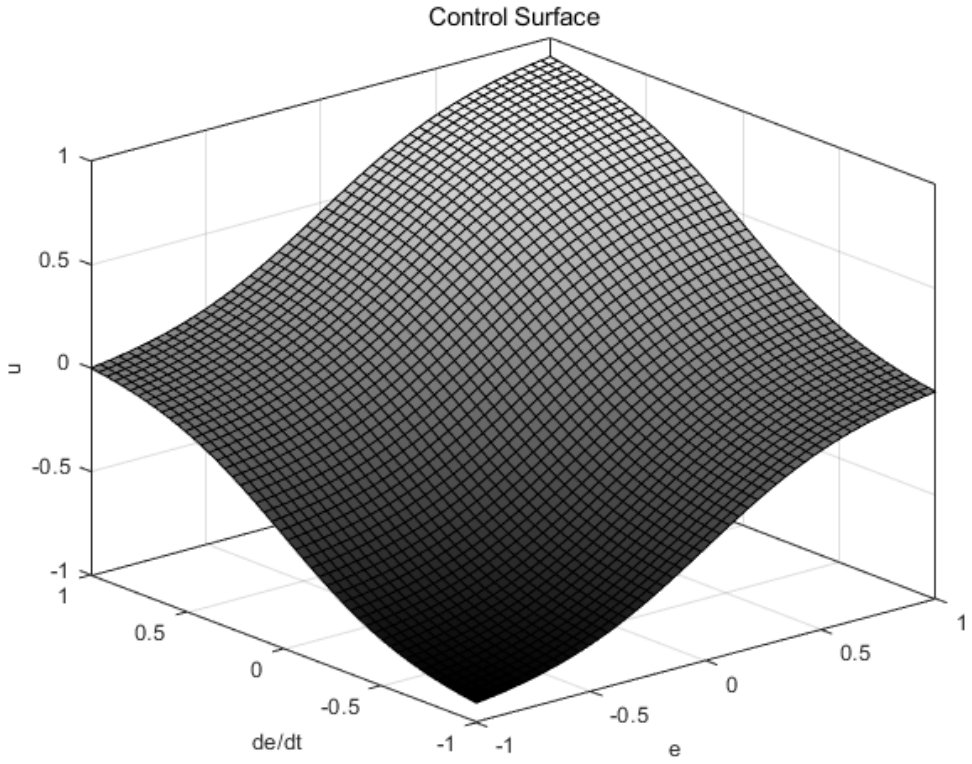


Figure 3. Nonlinear fuzzy control surface[15]

$$u_{c,rq}[n] = \alpha_{rq}[n]u_{FC,rq}[n] + \beta_{rq}[n]\Big(u_{FC,rq}[n] + u_{FC,rq}[n-1]\Big), \tag{27}$$

where $\alpha_{rq}[n]$ and $\beta_{rq}[n]$ are the output scaling parameters of the fuzzy subcontroller connecting the q-th controlled output to the r-th control input, extending the SISO output-scaling formulation of[15] and the main/coupling subcontroller structure of[16] to each (r, q) pair of the MIMO system.

As established for PID-type fuzzy controller structures in,[11,15,16] the two terms in (27) provide complementary control actions: the first term, $\alpha_{rq}[n]u_{FC,rq}[n]$, provides a nonlinear PD-type control contribution because the fuzzy controller output is generated from the scaled tracking error and its discrete-time difference, whereas the second term, $\beta_{rq}[n]\big(u_{FC,rq}[n] + u_{FC,rq}[n-1]\big)$, introduces an integral-like contribution through the incremental implementation of the control law; therefore, the combination of these two contributions yields the PID-type behavior of the fuzzy subcontroller associated with the (r, q) input–output pair. This formulation extends the SISO PID-type fuzzy controller structure in[15] to the incremental MIMO fuzzy PID architecture considered in.[16] The r-th control input is obtained by combining the direct control contribution with the cross-coupling contributions associated with the remaining controlled outputs. Accordingly, the contributions of all output-related subcontrollers are combined to generate the r-th control input, consistent with the main-and-coupling-subcontroller architecture of:[16]

$$u_r[n+1] = u_r[n] + \sum_{q=1}^{Q}\Big[\alpha_{rq}[n]u_{FC,rq}[n] + \beta_{rq}[n]\Big(u_{FC,rq}[n] + u_{FC,rq}[n-1]\Big)\Big], \quad r = 1, \ldots, R. \tag{28}$$

Thus, all Q controlled outputs contribute to the increment of each control input, allowing the controller to represent the coupled behavior of the MIMO system.[16]

In vector form, the nominal MIMO fuzzy PID control law is

$$\mathbf{u}[n+1] = \mathbf{u}[n] + \Delta\mathbf{u}_F[n+1], \tag{29}$$

where

$$\Delta\mathbf{u}_F[n+1] = \begin{bmatrix}\Delta u_1[n+1] & \cdots & \Delta u_R[n+1]\end{bmatrix}^{\mathrm{T}}. \tag{30}$$

Motivated by the $R \times 3Q$ MIMO PID parameter matrix structure introduced in,[2] in which each (r, q)-indexed control-input/controlled-output pair is assigned a dedicated set of PID gains, the adaptive input scaling parameter matrices of the MIMO fuzzy PID controller are defined analogously as

$$\mathbf{K}_e[n] = \begin{bmatrix} K_{e,11}[n] & \cdots & K_{e,1Q}[n] \\ \vdots & \ddots & \vdots \\ K_{e,R1}[n] & \cdots & K_{e,RQ}[n] \end{bmatrix}, \quad \mathbf{K}_{de}[n] = \begin{bmatrix} K_{de,11}[n] & \cdots & K_{de,1Q}[n] \\ \vdots & \ddots & \vdots \\ K_{de,R1}[n] & \cdots & K_{de,RQ}[n] \end{bmatrix}. \tag{31}$$

Similarly, the output scaling parameter matrices are defined as

$$\boldsymbol{\alpha}[n] = \begin{bmatrix} \alpha_{11}[n] & \cdots & \alpha_{1Q}[n] \\ \vdots & \ddots & \vdots \\ \alpha_{R1}[n] & \cdots & \alpha_{RQ}[n] \end{bmatrix}, \quad \boldsymbol{\beta}[n] = \begin{bmatrix} \beta_{11}[n] & \cdots & \beta_{1Q}[n] \\ \vdots & \ddots & \vdots \\ \beta_{R1}[n] & \cdots & \beta_{RQ}[n] \end{bmatrix}. \tag{32}$$

The adaptive parameters of the MIMO fuzzy PID controller are collected into the composite parameter matrix $\boldsymbol{\Theta}[n] \in \mathbb{R}^{4\times RQ}$ or, equivalently, into the stacked parameter vector $\boldsymbol{\rho}[n] \in \mathbb{R}^{4RQ}$:

$$\boldsymbol{\Theta}[n] = \begin{bmatrix} \mathrm{vec}^{\mathrm{T}}\mathbf{K}_e[n] \\ \mathrm{vec}^{\mathrm{T}}\mathbf{K}_{de}[n] \\ \mathrm{vec}^{\mathrm{T}}\boldsymbol{\alpha}[n] \\ \mathrm{vec}^{\mathrm{T}}\boldsymbol{\beta}[n] \end{bmatrix}^{\mathrm{T}}, \tag{33}$$

$$\boldsymbol{\rho}[n] = \mathrm{vec}(\boldsymbol{\Theta}[n]) = [\rho_1[n] \quad \rho_2[n] \quad \cdots \quad \rho_{4RQ}[n]]^{\mathrm{T}},$$

where $\mathrm{vec}(\cdot)$ denotes the column-stacking vectorization operator. For the considered 2×2 MIMO three-tank system ($R = Q = 2$), $\boldsymbol{\rho}[n]$ contains 16 online-tunable parameters.

To guarantee parameter boundedness and maintain numerical well-posedness, each adaptive parameter is confined to a strictly defined compact admissible interval:

$$\theta_{\min,l} \le \rho_l[n] \le \theta_{\max,l}, \qquad l = 1,\ldots,4RQ, \tag{34}$$

where $\theta_{\min,l}$ and $\theta_{\max,l}$ denote the lower and upper bounds of the l-th parameter component, respectively. Consequently, the admissible parameter space forms the compact hyper-rectangle:

$$\mathcal{R} = \prod_{l=1}^{4RQ} [\theta_{\min,l},\, \theta_{\max,l}] \subset \mathbb{R}^{4RQ}. \tag{35}$$

3.2. Predictive controller adaptation mechanism

$\mathrm{RK}_{\mathrm{model}}$, introduced in Section 2, predicts the system outputs over the prediction horizon κ. These predictions are used to calculate the future tracking errors and model sensitivities required for controller adaptation.[1,2]

The predicted output sequence over the prediction horizon can accordingly be expressed as

$$\hat{\mathbf{Y}}[n] = \begin{bmatrix} \hat{\mathbf{y}}^{\mathrm{T}}[n+1] & \cdots & \hat{\mathbf{y}}^{\mathrm{T}}[n+\kappa] \end{bmatrix}^{\mathrm{T}}, \tag{36}$$

following the multi-step prediction structure employed in.[1,2]

In the iterative computation of (36), the candidate control vector $\mathbf{u}[n+1]$ is assumed to remain unchanged over the prediction interval $[t+T_s,\ t+\kappa T_s]$, so that the κ-step-ahead outputs are obtained by applying $\mathbf{u}[n+1]$ repeatedly to $\mathrm{RK}_{\mathrm{model}}$.[2] This assumption renders the predicted outputs explicit functions of a single control vector and therefore makes the model sensitivities in (47) well defined.

The predicted tracking-error vector at the p-th prediction step is defined as[1,15]

$$\hat{\mathbf{e}}_y[n+p] = \mathbf{y}_{\mathrm{ref}}[n+p] - \hat{\mathbf{y}}[n+p], p = 1,\ldots,\kappa. \tag{37}$$

Its q-th component is

$$\hat{e}_q[n+p] = y_{ref_q}[n+p] - \hat{y}_q[n+p], \quad p = 1,\ldots,\kappa, \qquad q = 1,\ldots,Q. \tag{38}$$

The control objective includes the predicted tracking errors and a quadratic penalty on the control increments:[1,3,32]

$$F\big(\mathbf{u}[n+1]\big) = \frac{1}{2}\sum_{p=1}^{\kappa}\sum_{q=1}^{Q}\hat{e}_q^2[n+p] + \frac{1}{2}\sum_{r=1}^{R}\lambda_r\big(u_r[n+1]-u_r[n]\big)^2, \tag{39}$$

where $\lambda_r > 0$ is the control-increment penalty associated with the r-th control input.[1,3,32] The first term ($\frac{1}{2}$) penalizes the predicted tracking errors, whereas the second term ($\frac{1}{2}\lambda_r$) limits excessive control-input variations.[35] For compact notation, the stacked prediction-error vector is defined as

$$\hat{\mathbf{e}}_Y[n] = \begin{bmatrix} \hat{\mathbf{e}}_y^{\mathrm{T}}[n+1] & \cdots & \hat{\mathbf{e}}_y^{\mathrm{T}}[n+\kappa] \end{bmatrix}^{\mathrm{T}} \in \mathbb{R}^{\kappa Q}. \tag{40}$$

The control increment and its weighting matrix are

$$\Delta\mathbf{u}[n+1] = \mathbf{u}[n+1] - \mathbf{u}[n], \tag{41}$$

and

$$\boldsymbol{\Lambda} = \mathrm{diag}\big(\lambda_1,\ldots,\lambda_R\big), \tag{42}$$

respectively.

The extended prediction-error vector is then defined as

$$\hat{\mathbf{e}}[n] = \begin{bmatrix} \hat{\mathbf{e}}_Y[n] \\ \boldsymbol{\Lambda}^{1/2}\Delta\mathbf{u}[n+1] \end{bmatrix} \in \mathbb{R}^{\kappa Q+R}. \tag{43}$$

Consequently, the objective function can be expressed as

$$F\big(\mathbf{u}[n+1]\big) = \frac{1}{2}\hat{\mathbf{e}}^{\mathrm{T}}[n]\hat{\mathbf{e}}[n]. \tag{44}$$

The Jacobian required for parameter adaptation is defined as[1–3]

$$\mathbf{J}[n] = \frac{\partial \hat{\mathbf{e}}[n]}{\partial \boldsymbol{\rho}_{\mathrm{FPID}}[n]} \in \mathbb{R}^{(\kappa Q+R)\times 4RQ}. \tag{45}$$

Since the predicted outputs depend on the fuzzy PID parameters through the control vector, the chain rule gives[1–3]

$$\begin{aligned}\mathbf{J}[n] &= \frac{\partial \hat{\mathbf{e}}[n]}{\partial \mathbf{u}[n+1]} \frac{\partial \mathbf{u}[n+1]}{\partial \boldsymbol{\rho}_{\mathrm{FPID}}[n]} \\ &= \mathbf{J}_m[n]\mathbf{J}_c^{\mathrm{FPID}}[n],\end{aligned} \tag{46}$$

where

$$\begin{aligned}\mathbf{J}_m[n] &= \frac{\partial \hat{\mathbf{e}}[n]}{\partial \mathbf{u}[n+1]} \in \mathbb{R}^{(\kappa Q+R)\times R}, \\ \mathbf{J}_c^{\mathrm{FPID}}[n] &= \frac{\partial \mathbf{u}[n+1]}{\partial \boldsymbol{\rho}_{\mathrm{FPID}}[n]} \in \mathbb{R}^{R\times 4RQ}.\end{aligned} \tag{47}$$

Here, $\mathbf{J}_m[n]$ contains the RK model sensitivities, whereas $\mathbf{J}_c^{\mathrm{FPID}}[n]$ represents the sensitivity of the control vector to the fuzzy PID parameters.[1–3]

Since the parameters associated with the s-th controller do not directly affect $u_r[n+1]$ for $s \neq r$, the corresponding derivatives are

$$\begin{aligned}\frac{\partial u_r[n+1]}{\partial K_{e,sq}} &= \frac{\partial u_r[n+1]}{\partial K_{de,sq}} \\ &= \frac{\partial u_r[n+1]}{\partial \alpha_{sq}} \\ &= \frac{\partial u_r[n+1]}{\partial \beta_{sq}} = 0, s \quad \neq r.\end{aligned} \tag{48}$$

Extending the SISO output- and input-scaling derivative formulations of the fuzzy PID controller in[15] to the MIMO case, the output-scaling derivatives for the parameters associated with the r-th control input and the q-th controlled output are obtained as

$$\begin{aligned}\frac{\partial u_r[n+1]}{\partial \alpha_{rq}} &= u_{FC,rq}[n], \\ \frac{\partial u_r[n+1]}{\partial \beta_{rq}} &= u_{FC,rq}[n] + u_{FC,rq}[n-1].\end{aligned} \tag{49}$$

Similarly, extending the SISO input-scaling derivative formulation of[15] to the MIMO case, the derivatives with respect to the input scaling parameters are

$$\begin{aligned}\frac{\partial u_r[n+1]}{\partial K_{e,rq}} &= \underbrace{\big(\alpha_{rq}[n] + \beta_{rq}[n]\big)}_{\dfrac{\partial u_r[n+1]}{\partial u_{FC,rq}[n]}} \frac{\partial u_{FC,rq}[n]}{\partial e_{n,rq}[n]} \\ &\quad \underbrace{e_q[n]}_{\dfrac{\partial e_{n,rq}[n]}{\partial K_{e,rq}}}, \\ \frac{\partial u_r[n+1]}{\partial K_{de,rq}} &= \big(\alpha_{rq}[n] + \beta_{rq}[n]\big)\frac{\partial u_{FC,rq}[n]}{\partial \dot{e}_{n,rq}[n]} \\ &\quad \Delta e_q[n].\end{aligned} \tag{50}$$

In these derivatives, the previous fuzzy output $u_{FC,rq}[n-1]$ is treated as stored data and is independent of the current parameter update.

The membership-function sensitivities required within the active membership interval follow directly from the triangular membership definitions of,[15] generalized to each input–output pair (r, q), as

$$\begin{aligned}\frac{\partial A_i}{\partial e_{n,rq}[n]} &= -\frac{1}{e_{i+1} - e_i}, \\ \frac{\partial A_{i+1}}{\partial e_{n,rq}[n]} &= \frac{1}{e_{i+1} - e_i}, \\ \frac{\partial B_j}{\partial \dot{e}_{n,rq}[n]} &= -\frac{1}{\dot{e}_{j+1} - \dot{e}_j}, \\ \frac{\partial B_{j+1}}{\partial \dot{e}_{n,rq}[n]} &= \frac{1}{\dot{e}_{j+1} - \dot{e}_j}.\end{aligned} \tag{51}$$

Consequently, applying the product rule to the fuzzy-controller output defined analogously to,[15] but extended to each (r, q) pair of the MIMO structure, the local fuzzy-controller sensitivities are obtained as

$$\begin{aligned}\frac{\partial u_{FC,rq}[n]}{\partial e_{n,rq}[n]} &= \sum_{a=i}^{i+1}\sum_{b=j}^{j+1} \frac{\partial A_a}{\partial e_{n,rq}[n]} B_b u_{ab}, \\ \frac{\partial u_{FC,rq}[n]}{\partial \dot{e}_{n,rq}[n]} &= \sum_{a=i}^{i+1}\sum_{b=j}^{j+1} A_a \frac{\partial B_b}{\partial \dot{e}_{n,rq}[n]} u_{ab}.\end{aligned} \tag{52}$$

These sensitivities follow directly from the piecewise-linear triangular membership functions. Using the total Jacobian in (46), the fuzzy PID parameter vector is updated by the LM method.[2,3,30,32]

Using the extended prediction error $\hat{\mathbf{e}}[n]$ and the total sensitivity Jacobian $\mathbf{J}[n] = \mathbf{J}_m[n]\mathbf{J}_c^{\mathrm{FPID}}[n]$ defined in (46), the raw parameter increment is computed using the Levenberg–Marquardt optimization algorithm:

$$\Delta\boldsymbol{\rho}^{\mathrm{LM}}[n] = -\left(\mathbf{J}^{\mathrm{T}}[n]\mathbf{J}[n] + \mu\mathbf{I}\right)^{-1}\mathbf{J}^{\mathrm{T}}[n]\hat{\mathbf{e}}[n], \tag{53}$$

where $\mu > 0$ is the LM damping factor used to improve numerical conditioning and to balance the Gauss–Newton and gradient-descent characteristics of the adaptation process.[28] The unconstrained parameter-adaptation increment is subsequently scaled by an adaptation learning rate $\gamma \in (0, 1]$:

$$\Delta\boldsymbol{\rho}[n] = \gamma\,\Delta\boldsymbol{\rho}^{\mathrm{LM}}[n]. \tag{54}$$

An additive correction term is included to improve the control action during transients and under model uncertainty:[2]

$$\delta\mathbf{u}[n+1] = -\eta\Big(\mathbf{J}_m^{\mathrm{T}}[n]\mathbf{J}_m[n]\Big)^{-1}\mathbf{J}_m^{\mathrm{T}}[n]\hat{\mathbf{e}}[n], \tag{55}$$

where $0 < \eta < 1$ is an adjustable correction coefficient, which is selected as the largest value within this interval that keeps the control signals, their rates of change, and the system outputs inside their admissible bounds.[2]

The corrected control vector before actuator saturation is given by:[2]

$$\begin{aligned}\mathbf{u}_{\text{corr}}[n+1] &= \mathbf{u}[n+1] + \delta\mathbf{u}[n+1] \\ &= \mathbf{u}[n+1] - \eta\Big(\mathbf{J}_m^{\mathrm{T}}[n]\mathbf{J}_m[n]\Big)^{-1} \\ &\quad \mathbf{J}_m^{\mathrm{T}}[n]\hat{\mathbf{e}}[n].\end{aligned} \tag{56}$$

To account for physical actuator limitations, the corrected control input is subjected to actuator saturation constraints:

$$\mathcal{U} = \Big\{\mathbf{u} \in \mathbb{R}^R : u_{r,\min} \le u_r \le u_{r,\max}, r = 1, \ldots, R\Big\}. \tag{57}$$

The control signal injected into the plant is therefore given by

$$\mathbf{u}_a[n+1] = \text{sat}_{\mathcal{U}}\big(\mathbf{u}_{\text{corr}}[n+1]\big), \tag{58}$$

where $\text{sat}_{\mathcal{U}}(\cdot)$ enforces componentwise saturation between $u_{r,\min}$ and $u_{r,\max}$.

To ensure that the controller parameters strictly remain within the compact admissible set $\mathcal{R}$ without introducing abrupt derivative discontinuities, an online projection operator with a boundary transition layer is employed, building upon the projection mechanisms utilized for robust adaptive control.[36] For each parameter component $l \in \{1, \ldots, 4RQ\}$, let the parameter range be $\Delta\theta_l = \theta_{\max,l} - \theta_{\min,l}$. A boundary layer thickness is specified as $\epsilon_l = \delta_\epsilon \Delta\theta_l$, where $\delta_\epsilon \in (0, 0.5)$ denotes the boundary layer ratio (chosen as 0.05 in the implementation).

The directional projection modifier adjusts the parameter update direction near the boundaries as follows:[36]

$$\Delta\rho_l^{\text{proj}}[n] = \begin{cases} \left(\dfrac{\theta_{\max,l} - \rho_l[n]}{\epsilon_l}\right)\Delta\rho_l[n], \\ \quad \text{if } \rho_l[n] > \theta_{\max,l} - \epsilon_l \\ \quad \text{and } \Delta\rho_l[n] > 0, \\ \left(\dfrac{\rho_l[n] - \theta_{\min,l}}{\epsilon_l}\right)\Delta\rho_l[n], \\ \quad \text{if } \rho_l[n] < \theta_{\min,l} + \epsilon_l \\ \quad \text{and } \Delta\rho_l[n] < 0, \\ \Delta\rho_l[n], \\ \quad \text{otherwise.} \end{cases} \tag{59}$$

Using the projected increment in (59), an intermediate candidate parameter is generated as:

$$\rho_l^{\text{cand}}[n] = \rho_l[n] + \Delta\rho_l^{\text{proj}}[n]. \tag{60}$$

Because discrete-time updates with finite sampling intervals may occasionally overshoot the boundary under severe transients, a discrete-time boundary guard is applied as a safeguard:

$$\begin{aligned}\rho_l[n+1] &= \text{sat}_{[\theta_{\min,l},\, \theta_{\max,l}]}\left(\rho_l^{\text{cand}}[n]\right) \\ &= \max\left(\theta_{\min,l},\, \min\left(\theta_{\max,l},\, \rho_l^{\text{cand}}[n]\right)\right).\end{aligned} \tag{61}$$

The combination of (59)–(61) ensures that the compact parameter set $\mathcal{R}$ is strictly forward-invariant:

$$\boldsymbol{\rho}[0] \in \mathcal{R} \implies \boldsymbol{\rho}[n] \in \mathcal{R}, \qquad \forall n \ge 0. \tag{62}$$

Moreover, the effective parameter increment applied to the controller at instant n satisfies $\Delta\boldsymbol{\rho}^{\text{applied}}[n] = \boldsymbol{\rho}[n+1] - \boldsymbol{\rho}[n]$.

At each sampling instant, the fuzzy PID controller generates a nominal control vector using the current tracking errors. RK_{model} then predicts the future outputs and calculates the required sensitivities, following the RK-model-based prediction and Jacobian computation framework of.[2] The fuzzy PID parameters are updated by the LM method, analogous to the LM-based fuzzy PID parameter adaptation of,[15] and the corrected control vector is applied to the nonlinear MIMO system. Thus, the controller parameters are adapted according to the predicted tracking behavior of the controlled system, extending both frameworks to the MIMO fuzzy-PID case addressed in this work.

3.3. Stability analysis

Motivated by the Lyapunov-based stability framework employed in the Runge–Kutta sparse identification (RK-SINDYc)-based MPC study,[37] a stability analysis tailored to the proposed adaptive controller is developed in this section. The stability of the introduced adaptive controller is examined using a discrete-time Lyapunov framework. Closed-loop adaptive control systems are affected by system modeling errors, numerical approximation errors, estimation errors, measurement noise, reference variations, and residual nonlinear effects; therefore, exact convergence of the tracking error to zero may not always occur. Thus, in this analysis, our aim is to establish uniform ultimate boundedness (UUB) of the tracking error while ensuring the boundedness of the adaptive fuzzy PID parameters.

The tracking-error vector can be defined as

$$\mathbf{e}[n] = \mathbf{y}_{\text{ref}}[n] - \mathbf{y}[n], \tag{63}$$

and let $\boldsymbol{\rho}[n] \in \mathbb{R}^{N_\rho}$ stand for the parameter vector composed of fuzzy PID scaling coefficients.

To prevent parameter drift, the controller parameters are adapted using the projection algorithm, so the parameter vector is bounded as shown below.

$$\|\boldsymbol{\rho}[n]\| \leq \bar{\rho} < \infty. \tag{64}$$

Moreover, since

$$\mathbf{J}^T[n]\mathbf{J}[n] + \mu\mathbf{I} \succeq \mu\mathbf{I}, \tag{65}$$

if $\|\mathbf{J}[n]\| \leq \bar{J}$, then

$$\|\Delta\boldsymbol{\rho}[n]\| \leq \frac{\bar{J}}{\mu}\|\hat{\mathbf{e}}[n]\|. \tag{66}$$

Hence, LM regularization limits the unconstrained parameter increment for bounded prediction errors, whereas the projection mechanism prevents unbounded parameter drift.

The local one-step closed-loop tracking-error dynamics can be defined as:

$$\mathbf{e}[n+1] = \mathbf{A}_c(\boldsymbol{\rho}[n], n)\mathbf{e}[n] + \mathbf{d}[n], \tag{67}$$

where $\mathbf{A}_c(\boldsymbol{\rho}[n], n)$ is the local closed-loop error-transition matrix and $\mathbf{d}[n]$ stands for the aggregate disturbance given as:

$$\begin{aligned}\mathbf{d}[n] = {} & \mathbf{d}_{\text{model}}[n] + \mathbf{d}_{\text{RK}}[n] \\ & + \mathbf{d}_{\text{estimation}}[n] + \mathbf{d}_{\text{noise}}[n] \\ & + \mathbf{d}_{\text{reference}}[n] + \mathbf{d}_{\text{nonlinear}}[n].\end{aligned} \tag{68}$$

Thus, in the error term, uncertainties arising from model mismatch, RK truncation error, state/parameter-estimation errors, measurement noise, bounded reference variations, and higher-order or unmodeled nonlinear effects can be represented by $\mathbf{d}[n]$.

The fourth-order Runge–Kutta method satisfies the following truncation error condition:

$$\|\mathbf{d}_{\text{RK}}[n]\| \leq c_{\text{RK}} T_s^5 \tag{69}$$

under the usual smoothness assumptions.

Assuming that all disturbance components are uniformly bounded, the aggregate disturbance satisfies

$$\|\mathbf{d}[n]\| \leq \bar{d} < \infty, \tag{70}$$

where

$$\begin{aligned}\bar{d} = {} & \bar{d}_{\text{model}} + c_{\text{RK}} T_s^5 + \bar{d}_{\text{estimation}} \\ & + \bar{d}_{\text{noise}} + \bar{d}_{\text{reference}} + \bar{d}_{\text{nonlinear}}.\end{aligned} \tag{71}$$

In the remainder of the analysis, $\bar{d}$ is used as the uniform upper bound representing all aggregate disturbance effects.

The quadratic Lyapunov candidate can be expressed as follows:

$$V[n] = \mathbf{e}^T[n]\mathbf{P}\mathbf{e}[n]. \tag{72}$$

Since $\mathbf{P} > 0$,

$$p_{\min}\|\mathbf{e}[n]\|^2 \leq V[n] \leq p_{\max}\|\mathbf{e}[n]\|^2. \tag{73}$$

where $p_{\min} = \lambda_{\min}(\mathbf{P})$ and $p_{\max} = \lambda_{\max}(\mathbf{P})$. Using (67), the one-step Lyapunov difference is

$$\begin{aligned}\Delta V[n] = {} & \mathbf{e}^T[n]\underbrace{\left(\mathbf{A}_c^T[n]\mathbf{P}\mathbf{A}_c[n] - \mathbf{P}\right)}_{-\mathbf{Q}}\mathbf{e}[n] \\ & + 2\mathbf{e}^T[n]\mathbf{A}_c^T[n]\mathbf{P}\mathbf{d}[n] \\ & + \mathbf{d}^T[n]\mathbf{P}\mathbf{d}[n].\end{aligned} \tag{74}$$

where

$$q_{\min} = \lambda_{\min}(\mathbf{Q}) > 0. \tag{75}$$

For stability, $\Delta V[n] \leq 0$ must be satisfied as follows:

$$\begin{aligned}\Delta V[n] \leq {} & -\mathbf{e}^T[n]\mathbf{Q}\mathbf{e}[n] \\ & + 2\mathbf{e}^T[n]\mathbf{A}_c^T[n]\mathbf{P}\mathbf{d}[n] \\ & + \mathbf{d}^T[n]\mathbf{P}\mathbf{d}[n].\end{aligned} \tag{76}$$

The upper bound for the first term in (76) can be obtained as follows:

$$-\mathbf{e}^T[n]\mathbf{Q}\mathbf{e}[n] \leq -q_{\min}\|\mathbf{e}[n]\|^2 \tag{77}$$

and, over the considered compact operating region,

$$\|\mathbf{A}_c(\boldsymbol{\rho}[n], n)\| \leq \bar{A}. \tag{78}$$

The cross term in (76) satisfies

$$\begin{aligned}2\mathbf{e}^T\mathbf{A}_c^T\mathbf{P}\mathbf{d} &\leq 2\|\mathbf{e}\|\|\mathbf{A}_c\|\|\mathbf{P}\|\|\mathbf{d}\| \\ &\leq 2\bar{A}p_{\max}\|\mathbf{e}\|\|\mathbf{d}\|.\end{aligned} \tag{79}$$

By defining $c = \bar{A}p_{\max}$ and using $\mathbf{d}^T\mathbf{P}\mathbf{d} \leq p_{\max}\|\mathbf{d}\|^2$, (76) can be rewritten as follows:

$$\begin{aligned}\Delta V[n] \leq {} & -q_{\min}\|\mathbf{e}[n]\|^2 \\ & + 2c\|\mathbf{e}[n]\|\|\mathbf{d}[n]\| + p_{\max}\|\mathbf{d}[n]\|^2.\end{aligned} \tag{80}$$

Using $\|\mathbf{d}[n]\| \leq \bar{d}$, (80) is given as:

$$\begin{aligned}\Delta V[n] \leq {} & -q_{\min}\|\mathbf{e}[n]\|^2 + 2c\bar{d}\|\mathbf{e}[n]\| \\ & + p_{\max}\bar{d}^2.\end{aligned} \tag{81}$$

The second term on the RHS of (81) can be written as follows via Young's inequality:

$$2c\bar{d}\|\mathbf{e}[n]\| \leq \varepsilon_L\|\mathbf{e}[n]\|^2 + \frac{c^2}{\varepsilon_L}\bar{d}^2, \tag{82}$$

where $0 < \varepsilon_L < q_{\min}$. Thus, (81) can be obtained as:

$$\begin{aligned}\Delta V[n] \leq {} & -\underbrace{(q_{\min} - \varepsilon_L)}_{\alpha}\|\mathbf{e}[n]\|^2 \\ & + \underbrace{\left(p_{\max} + \frac{c^2}{\varepsilon_L}\right)\bar{d}^2}_{\delta}.\end{aligned} \tag{83}$$

Then, (83) is reduced as:

$$\Delta V[n] \leq -\alpha\|\mathbf{e}[n]\|^2 + \delta. \tag{84}$$

In conclusion, $\Delta V[n] < 0$ is satisfied as long as

$$\|\mathbf{e}[n]\| > \sqrt{\frac{\delta}{\alpha}}, \tag{85}$$

and (85) indicates that the Lyapunov function decreases outside a bounded neighborhood of the origin.

To obtain an explicit UUB result, from (73), the following expression can be obtained:

$$\|\mathbf{e}[n]\|^2 \geq \frac{V[n]}{p_{\max}} \tag{86}$$

Hence, using (86), (84) can be given as:

$$\Delta V[n] \leq -\kappa V[n] + \delta. \tag{87}$$

where

$$\kappa = \frac{\alpha}{p_{\max}}, \qquad 0 < \kappa < 1, \tag{88}$$

Thus,

$$V[n+1] \leq (1-\kappa)V[n] + \delta. \tag{89}$$

As a result of recursive application of (89), the following is obtainedacquired:

$$V[n] \leq (1-\kappa)^n V[0] + \frac{\delta}{\kappa}\left[1-(1-\kappa)^n\right]. \tag{90}$$

Since $0 < \kappa < 1$, the upper limit can be derived as:

$$\limsup_{n\to\infty} V[n] \leq \frac{\delta}{\kappa} = \frac{p_{\max}\delta}{\alpha}. \tag{91}$$

Since $\|\mathbf{e}[n]\|^2 \leq \frac{1}{p_{\min}} V[n]$, the tracking error satisfies the following inequality:

$$\limsup_{n\to\infty} \|\mathbf{e}[n]\| \leq r_{\mathrm{UUB}}, \tag{92}$$

where r_{UUB} denotes the radius of the uniform ultimate bound (UUB) on the tracking error and is given as follows:

$$r_{\mathrm{UUB}} = \left[\frac{p_{\max}}{p_{\min}} \frac{(p_{\max} + c^2/\varepsilon_L)}{(q_{\min} - \varepsilon_L)}\right]^{1/2} \bar{d}. \tag{93}$$

Thus, the closed-loop tracking error is uniformly ultimately bounded by r_{UUB} in the presence of bounded aggregate disturbances $\bar{d}$.

For the nominal disturbance-free case, $\mathbf{d}[n] \equiv \mathbf{0}$,

$$\Delta V[n] \leq -\mathbf{e}^T[n]\mathbf{Q}\mathbf{e}[n] \leq -q_{\min}\|\mathbf{e}[n]\|^2 < 0 \tag{94}$$

for every $\mathbf{e}[n] \neq \mathbf{0}$. Hence, $\mathbf{e}[n] \to \mathbf{0}$.

4. Simulation results

In this study, the proposed adaptive MIMO fuzzy PID controller based on the RK model is comparatively analyzed with respect to the RK model-based adaptive MIMO PID controller presented by Çetin and İplikçi.[2] These two controllers are tested on the same nonlinear three-tank liquid-level control problem, with the same plant parameters, initial conditions, reference trajectories, sampling times, input constraints, and operating conditions. The plant and the prediction models are propagated numerically via the RK4 scheme.

All simulations were implemented in MATLAB R2019b and executed on a 64-bit computer equipped with a 13th-generation Intel Core i7-1370P processor operating at 1.90 GHz and 16 GB of RAM under the Windows 11 Enterprise operating system. The computation times reported in Section 4.5 were obtained using this computational environment.

Three simulation scenarios are considered. The first scenario represents nominal operation without measurement noise and parameter uncertainty. In the second scenario, measurement noise is added to the measured liquid levels. In the third scenario, the inter-tank flow coefficient az_{13} is varied over time to introduce plant–model mismatch. The three scenarios are used to evaluate tracking accuracy, noise sensitivity, online adaptation, parameter-estimation capability, and computational performance. For a fair comparison, the same measurement-noise realizations and the same time-varying parameter profile are applied to both controllers in the corresponding scenarios. The comparative evaluation considers the tracking responses, control inputs, adaptive controller parameters, additive correction contributions, composite performance indices, and computation times.

4.1. Three-tank system model

The three-tank system is a widely adopted benchmark for assessing the control performance of MIMO controller architectures developed for nonlinear MIMO systems in the literature.[1–3] Its schematic representation is shown in **Figure 4**. The setup consists of three identical cylindrical tanks connected in series, two pumps responsible for circulating water from the reservoir at the base, interconnecting valves that regulate the flow between adjacent tanks, and a common water reservoir located beneath the tanks.

The state, input, and controlled-output vectors are defined as

$$\begin{aligned} \mathbf{x}(t) &= \begin{bmatrix} h_1(t) & h_2(t) & h_3(t) \end{bmatrix}^{\mathrm{T}}, \\ \mathbf{u}(t) &= \begin{bmatrix} q_1(t) & q_2(t) \end{bmatrix}^{\mathrm{T}}, \\ \mathbf{y}(t) &= \begin{bmatrix} h_1(t) & h_2(t) \end{bmatrix}^{\mathrm{T}}. \end{aligned} \tag{95}$$

The nonlinear model is derived from the mass-balance equations of the three tanks.[29] Assuming that the tanks have the same cross-sectional area A, the liquid-level dynamics are described by[1–3,30]

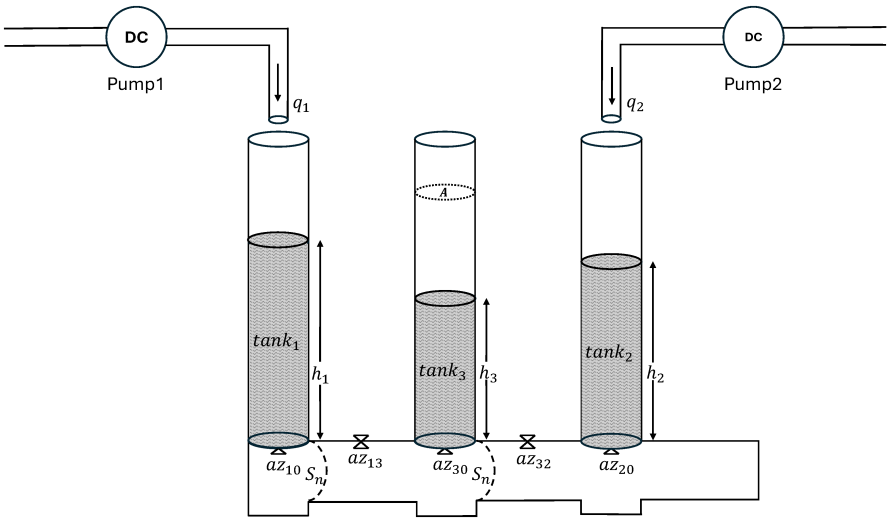


Figure 4. Schematic representation of the three-tank liquid-level system[1–3, 29, 30]

$$\begin{aligned}
\dot{h}_1(t) &= \frac{1}{A}\Big[q_1(t) - Q_{13}(t) - Q_{10}(t)\Big],\\
\dot{h}_2(t) &= \frac{1}{A}\Big[q_2(t) + Q_{32}(t) - Q_{20}(t)\Big],\\
\dot{h}_3(t) &= \frac{1}{A}\Big[Q_{13}(t) - Q_{32}(t) - Q_{30}(t)\Big].
\end{aligned} \tag{96}$$

Here, $Q_{13}(t)$ and $Q_{32}(t)$ denote the inter-tank flow rates, whereas $Q_{10}(t)$, $Q_{20}(t)$, and $Q_{30}(t)$ denote the outlet flow rates.[1–3, 30] According to Bernoulli's law, the inter-tank flows are expressed as[1–3, 30]

$$\begin{aligned}
Q_{13}(t) &= az_{13} S_n \operatorname{sgn}\big(h_1(t) - h_3(t)\big)\\
&\quad \times \sqrt{2g\,|h_1(t) - h_3(t)|},\\
Q_{32}(t) &= az_{32} S_n \operatorname{sgn}\big(h_3(t) - h_2(t)\big)\\
&\quad \times \sqrt{2g\,|h_3(t) - h_2(t)|}.
\end{aligned} \tag{97}$$

The outlet flows are given by[1–3, 30]

$$\begin{aligned}
Q_{10}(t) &= az_{10} S_n \sqrt{2gh_1(t)},\\
Q_{20}(t) &= az_{20} S_n \sqrt{2gh_2(t)},\\
Q_{30}(t) &= az_{30} S_n \sqrt{2gh_3(t)}.
\end{aligned} \tag{98}$$

In these equations, S_n is the outlet cross-sectional area, g is the gravitational acceleration, and az_{ij} denotes the corresponding flow coefficient.[1–3, 29, 30] The sign functions in (97) allow the inter-tank flow directions to vary according to the relative liquid levels.[1–3, 29, 30] The nonlinear model in (96) is discretized using the RK4 method presented in Section 2.1.

The nominal plant parameters, initial conditions, and controller settings used in the simulations are summarized in **Table 1**. In the nominal and measurement-noise cases, the plant and RK model use the same nominal parameters. In the parameter-uncertainty case, the actual value of az_{13} is varied while $\mathrm{RK}_{\mathrm{model}}$ is initialized with its nominal value.

Several prediction horizon lengths were investigated during the controller design stage. While extending the prediction horizon enables the controller to anticipate system behavior further into the future, it also increases the computational burden associated with solving the online optimization problem. Based on extensive simulation studies, a prediction horizon of $\kappa = 5$ was found to provide a favorable balance between predictive capability and computational efficiency, and was therefore adopted in the final controller configuration.

The damping parameter of the LM algorithm, denoted by μ, was determined through a series of preliminary numerical experiments involving different candidate values. Within the LM framework, the damping factor improves the numerical properties of the approximated Hessian matrix by reducing ill-conditioning and facilitating a stable matrix inversion process. The simulation results indicated that $\mu = 1$ ensured reliable convergence characteristics and robust matrix inversion without introducing excessive damping effects. Consequently, this value was selected for the proposed controller.

The weighting coefficient associated with the control increment term, λ, was selected by considering the relative numerical magnitudes of the terms appearing in the cost function. Since the manipulated variables of the system are typically on the order of 10^{-4}, the contribution of the control effort term remains significantly smaller than that of the tracking error term unless an appropriate scaling factor is introduced. For this reason, $\lambda = 10^4$ was employed to increase the influence of the control component within the optimization

criterion. Without this weighting adjustment, the cost function would be dominated by the tracking error term, potentially encouraging the optimizer to achieve improved tracking performance at the expense of large variations in the manipulated variables, which may result in undesirable chattering behavior. The projection algorithm parameters deployed in the adaptive fuzzy MIMO PID are tabulated in **Table 2**.

Table 1. System parameters, initial conditions, common predictive-adaptation settings, and controller-specific initialization values used in the comparative simulations[1–3,30]

Parameter	Description	Value
az_{13}	Flow coefficient between Tanks 1 and 3	0.5
az_{32}	Flow coefficient between Tanks 3 and 2	0.55
az_{10}	Outlet flow coefficient of Tank 1	0.26
az_{20}	Outlet flow coefficient of Tank 2	0.28
az_{30}	Outlet flow coefficient of Tank 3	0.45
A	Tank cross-sectional area	$0.0154\ \mathrm{m}^2$
S_n	Outlet cross-sectional area	$5 \times 10^{-5}\ \mathrm{m}^2$
g	Gravitational acceleration	$9.81\ \mathrm{m\,s}^{-2}$
T_s	Sampling period	1 s
N_{sim}	Number of simulation samples	1800
$h_1(0)$	Initial liquid level of Tank 1	0.02 m
$h_2(0)$	Initial liquid level of Tank 2	0.02 m
$h_3(0)$	Initial liquid level of Tank 3	0.02 m
$q_1(0)$	Initial inlet flow rate of Tank 1	$1.0 \times 10^{-4}\ \mathrm{m}^3\,\mathrm{s}^{-1}$
$q_2(0)$	Initial inlet flow rate of Tank 2	$1.0 \times 10^{-4}\ \mathrm{m}^3\,\mathrm{s}^{-1}$
$q_{\min}$	Minimum control input	$0\ \mathrm{m}^3\,\mathrm{s}^{-1}$
$q_{\max}$	Maximum control input	$1.0 \times 10^{-4}\ \mathrm{m}^3\,\mathrm{s}^{-1}$
Adaptive fuzzy PID controller settings		
$K_{e,rq}(0)$	Initial error scaling parameter	1.0×10^{-25}
$K_{de,rq}(0)$	Initial error-difference scaling parameter	1.0×10^{-25}
$\alpha_{rq}(0)$	Initial output scaling parameter α	1.0×10^{-25}
$\beta_{rq}(0)$	Initial output scaling parameter β	1.0×10^{-25}
κ	Prediction horizon	5
λ_1	Control-increment penalty for q_1	1.0×10^4
λ_2	Control-increment penalty for q_2	1.0×10^4
μ	LM damping parameter	1
η	Additive correction coefficient	0.9
Adaptive PID controller settings		
$K_{p,rq}(0)$	Proportional gain parameter	1.0×10^{-25}
$K_{i,rq}(0)$	Integral gain parameter	1.0×10^{-25}
$K_{d,rq}(0)$	Derivative gain parameter	1.0×10^{-25}
κ	Prediction horizon	5
λ_1	Control-increment penalty for q_1	1.0×10^4
λ_2	Control-increment penalty for q_2	1.0×10^4
μ	LM damping parameter	1
η	Additive correction coefficient	0.9
EKF Parameters (Measurement-Noise Scenario)		
$\mathbf{Q}$	Process-noise covariance	$10^{-7}\mathbf{I}_3$
$\mathbf{R}$	Measurement-noise covariance	$5 \times 10^{-5}\mathbf{I}_2$
$\mathbf{P}_0$	Initial state-estimation error covariance	$10^{-3}\mathbf{I}_3$

Table 2. Projection algorithm parameters used in the adaptive fuzzy MIMO PID controller

Parameter	$[\boldsymbol{\theta_{min}}\ \boldsymbol{\theta_{max}}]$
$K_{e_{11}}$	$[-0.04 \quad 0.02]$
$K_{de_{11}}$	$[-5\times10^{-6} \quad 0.02]$
α_{11}	$[-4\times10^{-5} \quad 0.015]$
β_{11}	$[-0.05 \quad 0.05]$
$K_{e_{12}}$	$[-0.03 \quad 0.035]$
$K_{de_{12}}$	$[-0.0005 \quad 0.025]$
α_{12}	$[-0.01 \quad 0.0035]$
β_{12}	$[5\times10^{-21} \quad 0.1]$
$K_{e_{21}}$	$[5\times10^{-21} \quad 5\times10^{-10}]$
$K_{de_{21}}$	$[5\times10^{-21} \quad 5\times10^{-11}]$
α_{21}	$[5\times10^{-21} \quad 25\times10^{-11}]$
β_{21}	$[5\times10^{-21} \quad 35\times10^{-10}]$
$K_{e_{22}}$	$[5\times10^{-21} \quad 0.035]$
$K_{de_{22}}$	$[5\times10^{-21} \quad 0.003]$
α_{22}	$[5\times10^{-21} \quad 0.03]$
β_{22}	$[-0.04 \quad 0.04]$
ϵ	10^{-4}

Similarly, the projection algorithm parameters utilized in the adaptive MIMO PID are given in **Table 3**.

Table 3. Projection algorithm parameters used in the adaptive MIMO PID controller

Parameter	$[\boldsymbol{\theta_{min}}\ \boldsymbol{\theta_{max}}]$
$K_{p_{11}},\ K_{p_{12}},\ K_{p_{22}}$	$[-0.002 \quad 0.002]$
$K_{i_{11}},\ K_{i_{12}},\ K_{i_{22}}$	$[-0.002 \quad 0.002]$
$K_{d_{11}},\ K_{d_{12}},\ K_{d_{22}}$	$[-0.002 \quad 0.002]$
$K_{p_{21}},\ K_{i_{21}},\ K_{d_{21}}$	$[-0.0002 \quad 0.0002]$
ϵ	10^{-4}

4.2. Nominal case

In the nominal case, the proposed RK model-based adaptive MIMO fuzzy PID controller and the benchmark RK model-based adaptive MIMO PID controller are evaluated without measurement noise or parametric uncertainty. For both controllers, the plant and the corresponding RK prediction model use the nominal system parameters. The comparative results include the tracking responses, control inputs, adaptive controller parameters, and additive correction contributions.

Figure 5 shows that $h_1(t)$ and $h_2(t)$ closely follow their staircase reference trajectories. Small transient deviations occur when the references change because of the dynamic coupling induced by the inter-tank flows. Nevertheless, the tracking errors rapidly decrease, and the control inputs $q_1(t)$ and $q_2(t)$ remain within the specified limits throughout the simulation.

Figure 6 shows that both liquid levels closely follow their staircase reference trajectories under the adaptive MIMO PID controller. Small transient deviations occur during reference changes because of the dynamic coupling between the tanks. Nevertheless, the tracking errors rapidly decrease, and both control inputs remain within the specified limits throughout the simulation.

To evaluate the effectiveness of the correction term and its influence on the overall control action, the percentage contributions of the main controller output and the additive correction term to the total control signal were analyzed for both controllers. **Figures 7** and **8** show these contributions for the adaptive fuzzy PID and adaptive PID controllers, respectively.

For both controllers, the correction terms provide supplementary control action mainly during the initial transient and reference changes. During steady-state operation, their contributions decrease, and the adapted main controllers generate most of the required control action.

Figures 9–12 present the online evolution of the fuzzy PID scaling parameters. FPID_{11} and FPID_{22} represent the direct subcontrollers, whereas FPID_{12} and FPID_{21} compensate for the cross-coupling effects. The parameters are mainly adjusted during the initial response and the reference transitions. All parameter trajectories remain bounded, and no continuous parameter drift is observed under nominal conditions.

Figures 13–16 present the online evolution of the adaptive PID gains. PID_{11} and PID_{22} represent the direct subcontrollers, whereas PID_{12} and PID_{21} compensate for the cross-coupling effects. The proportional, integral, and derivative gains are mainly adjusted during the initial response and reference transitions. All gain trajectories remain bounded, and no continuous parameter drift is observed under nominal conditions.

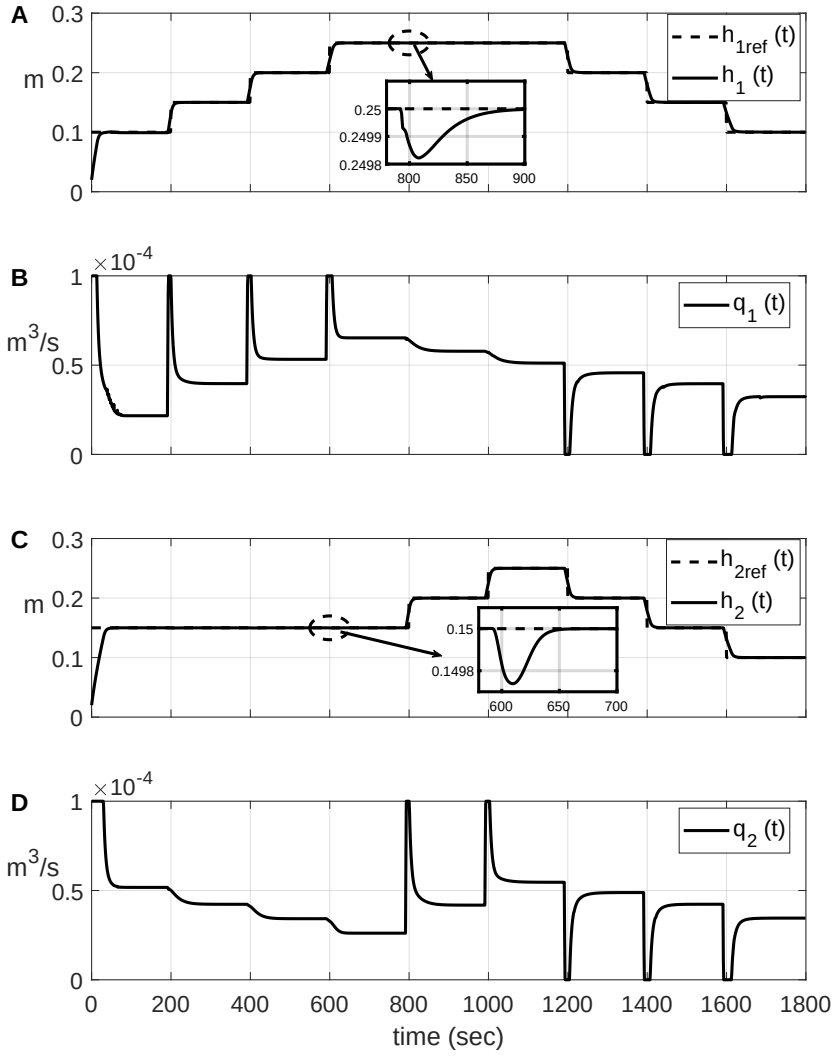


Figure 5. Comparison of the reference and actual liquid levels h_1 (**A**) and h_2 (**C**), and the corresponding control inputs q_1 (**B**) and q_2 (**D**) in the nominal case for fuzzy PID

Figure 17 illustrates the Lyapunov function $V[n]$ and its forward difference $\Delta V[n]$ for the proposed controller over the 0–180 s simulation interval in the nominal case. The Lyapunov candidate $V[n]$ decays rapidly after reference transitions and approaches zero during steady-state tracking, while $\Delta V[n] \leq 0$ is consistently maintained, corroborating the asymptotic convergence established in Section 3.3 for the nominal-case scenario.

Overall, the nominal-case results demonstrate that both the proposed RK model-based adaptive MIMO fuzzy PID controller and the benchmark RK model-based adaptive MIMO PID controller achieve accurate reference tracking with bounded control inputs and adaptive parameters. The direct and cross-coupling subcontrollers of both control structures effectively compensate for the nonlinear and coupled dynamics of the three-tank system.

4.3. Measurement noise case

In this case, measurement noise is added to the feedback signals, while the plant parameters are kept at their nominal values. The reference trajectories are identical to those used in the nominal case. The noise is modeled as zero-mean additive white Gaussian noise corrupting the measured liquid levels,

$$\begin{aligned} h_{m,i}[n] &= h_i[n] + v_i[n], \\ v_i[n] &\sim \mathcal{N}\big(0,\ \sigma_{v,i}^2\big), \qquad i = 1, 2. \end{aligned} \tag{99}$$

where $h_i[n]$ and $h_{m,i}[n]$ denote the noise-free and the measured levels of the i-th tank, respectively. The noise variances are selected such that each measurement attains a signal-to-noise ratio of 30 dB,[32,38]

$$\mathrm{SNR}_i = 10 \log_{10}\left(\frac{\sigma_{y,i}^2}{\sigma_{v,i}^2}\right) \ \mathrm{dB}, \tag{100}$$

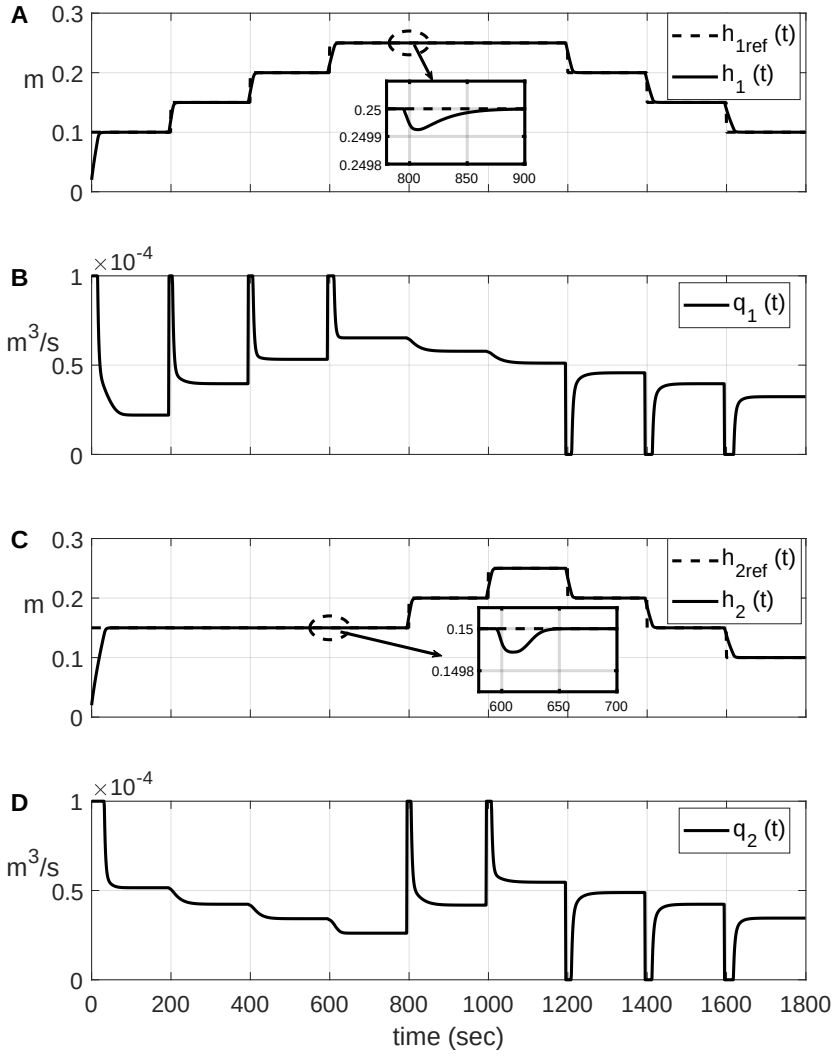


Figure 6. Comparison of the reference and actual liquid levels h_1 (**A**) and h_2 (**C**), and the corresponding control inputs q_1 (**B**) and q_2 (**D**) in the nominal case for PID

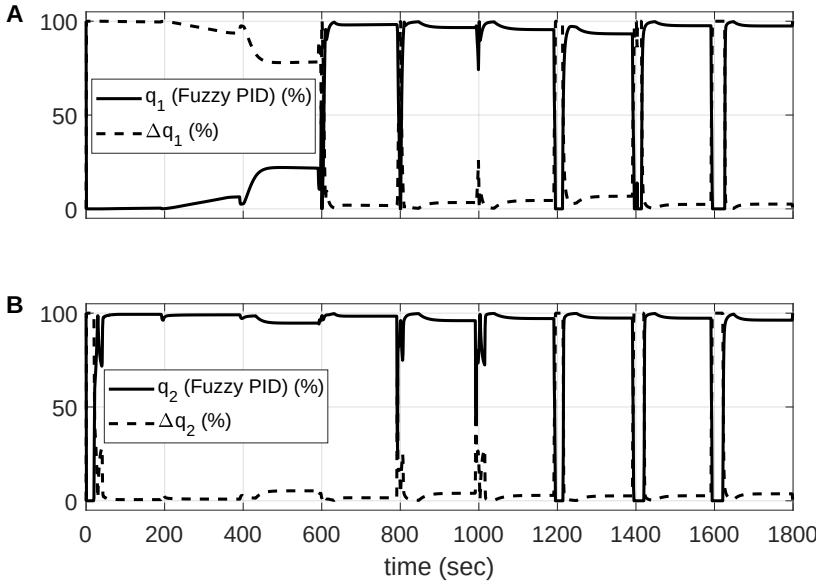


Figure 7. Comparison of the fuzzy PID and additive correction contributions to the control inputs q_1 (**A**) and q_2 (**B**) in the nominal case

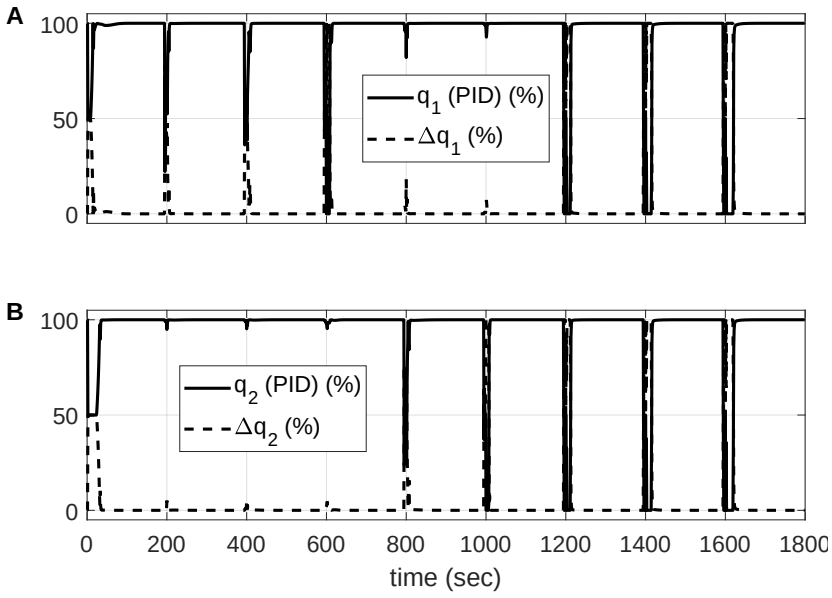


Figure 8. Comparison of the PID and additive correction contributions to the control inputs q_1 (**A**) and q_2 (**B**) in the nominal case

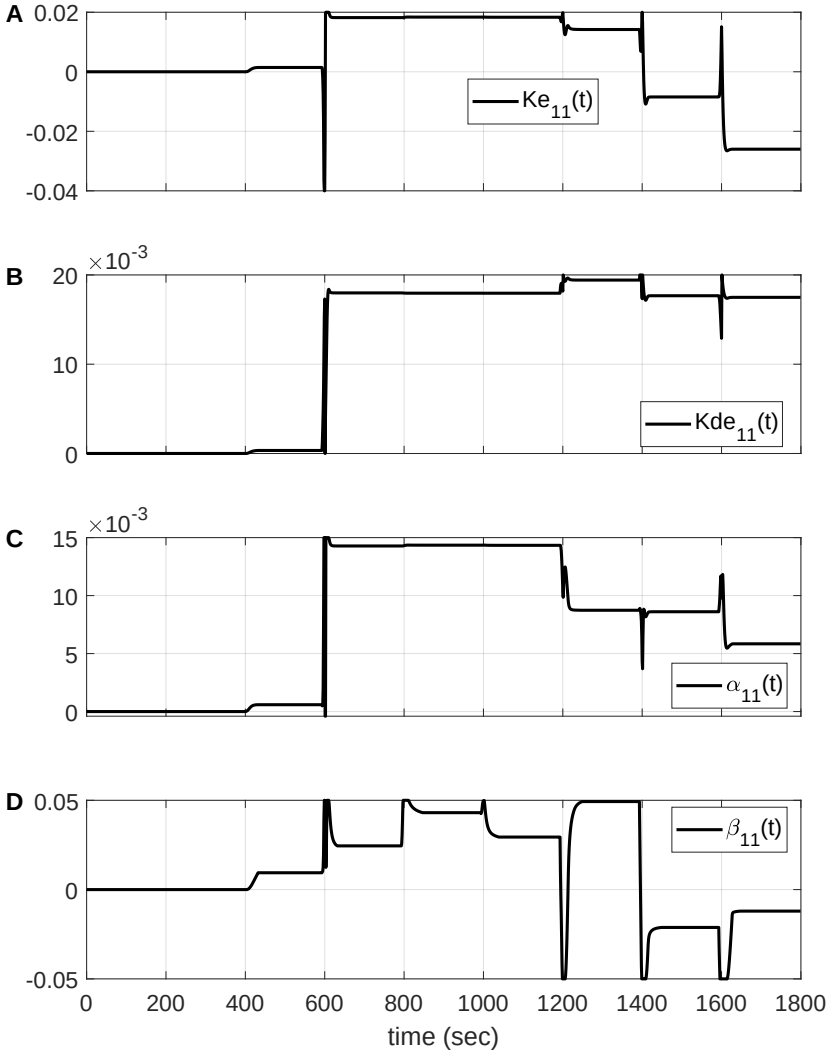


Figure 9. Evolution of the adaptive scaling parameters $K_{e,11}$ (**A**), $K_{de,11}$ (**B**), α_{11} (**C**) and β_{11} (**D**) of the direct FPID_{11} subcontroller in the nominal case for fuzzy PID

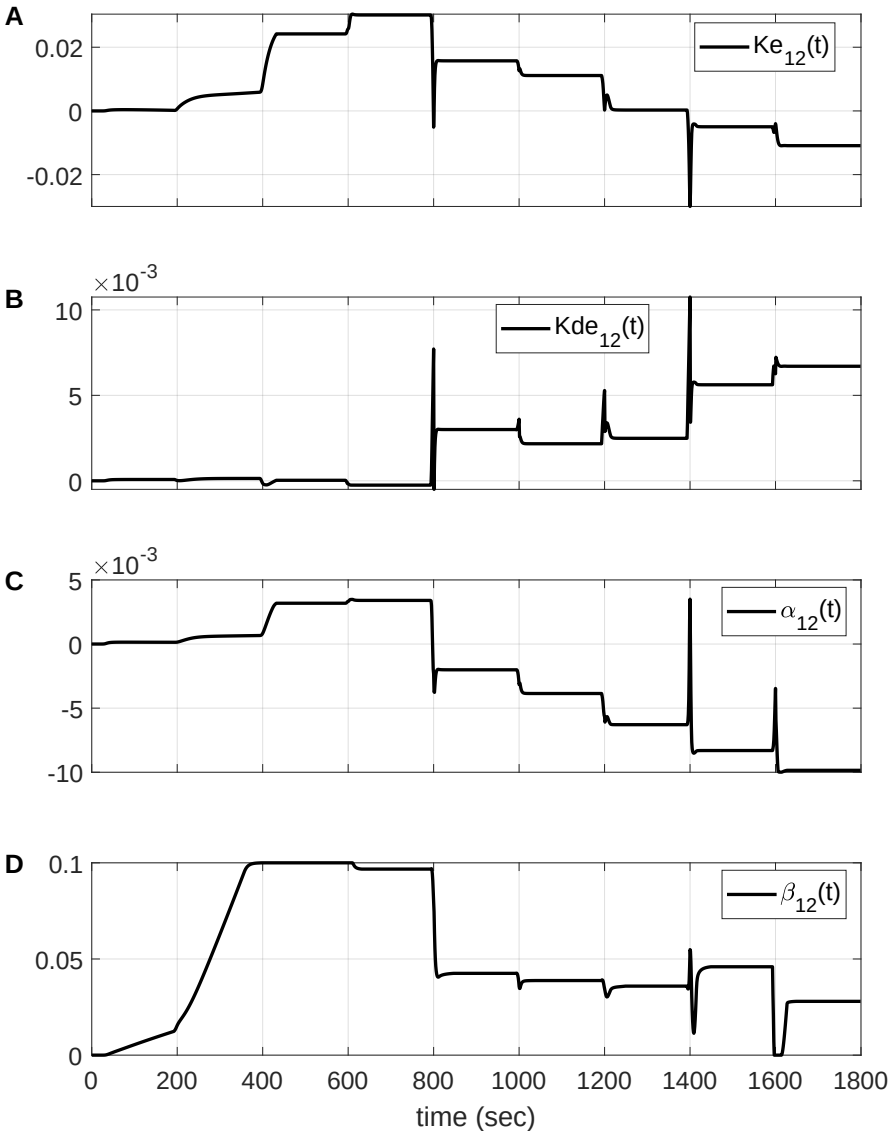


Figure 10. Evolution of the adaptive scaling parameters $K_{e,12}$ (**A**), $K_{de,12}$ (**B**), α_{12} (**C**) and β_{12} (**D**) of the cross-coupling FPID_{12} subcontroller in the nominal case for fuzzy PID

where $\sigma_{y,i}^2$ and $\sigma_{v,i}^2$ denote the variances of the noise-free output and of the injected noise, respectively, evaluated over the entire simulation horizon. The two noise sequences are mutually independent. The corrupted measurements are supplied to both control structures and therefore affect the tracking errors, the RK_{EKF} state estimates, the RK-model predictions, and the online adaptation of the controller parameters. This scenario is used to comparatively evaluate the tracking, adaptation, and control-input behavior of both controllers under identical noisy-feedback conditions.

Figures 18 and **19** present the tracking responses and control inputs obtained using the adaptive MIMO fuzzy PID and adaptive MIMO PID controllers, respectively, at an SNR of 30 dB. For both controllers, the liquid levels $h_1(t)$ and $h_2(t)$ closely follow their staircase reference trajectories. Small fluctuations are observed around the reference levels, particularly during steady-state intervals, because of the measurement noise. Nevertheless, both controllers maintain stable tracking, and the control inputs $q_1(t)$ and $q_2(t)$ remain within the specified limits without significant noise-induced chattering.

Figures 20 and **21** show the percentage contributions of the main controller outputs and the additive correction terms for the adaptive fuzzy PID and adaptive PID controllers, respectively.

For both controllers, the correction mechanism provides supplementary control action mainly during the initial transient and reference transitions. Although measurement noise introduces small variations in these contributions, the adapted main controllers continue to generate the dominant part of the control action.

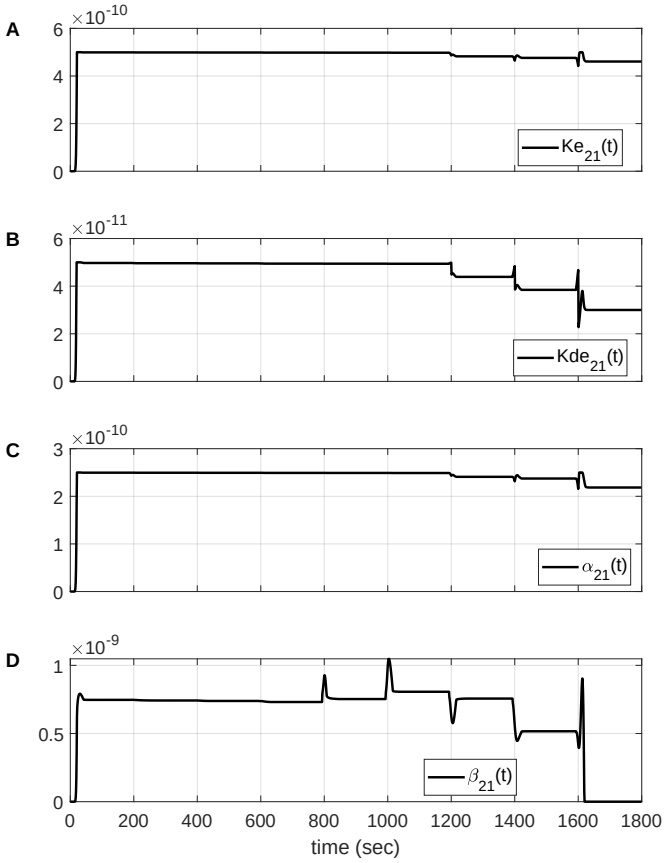


Figure 11. Evolution of the adaptive scaling parameters $K_{e,21}$ (**A**), $K_{de,21}$ (**B**), α_{21} (**C**) and β_{21} (**D**) of the cross-coupling FPID_{21} subcontroller in the nominal case for fuzzy PID

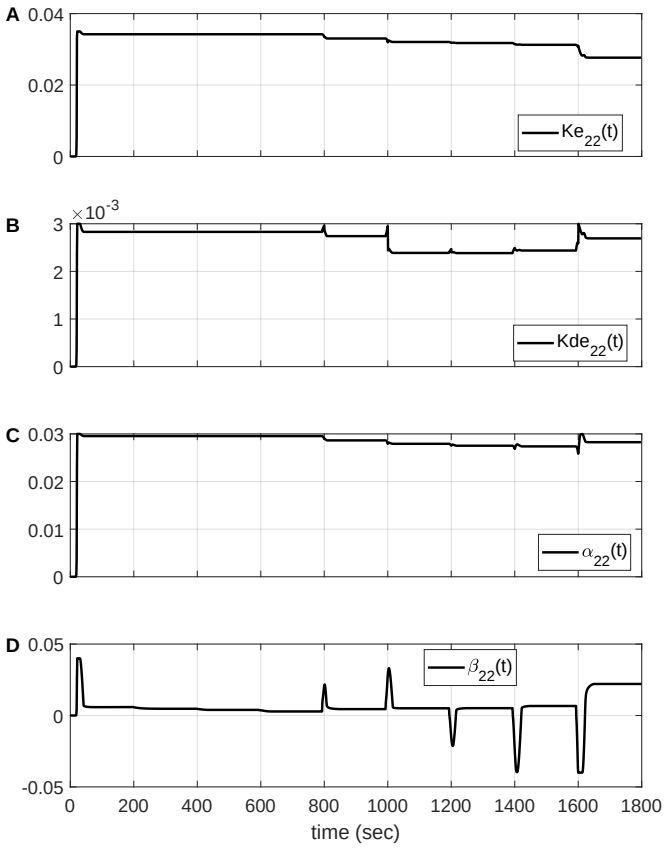


Figure 12. Evolution of the adaptive scaling parameters $K_{e,22}$ (**A**), $K_{de,22}$ (**B**), α_{22} (**C**) and β_{22} (**D**) of the direct FPID_{22} subcontroller in the nominal case for fuzzy PID

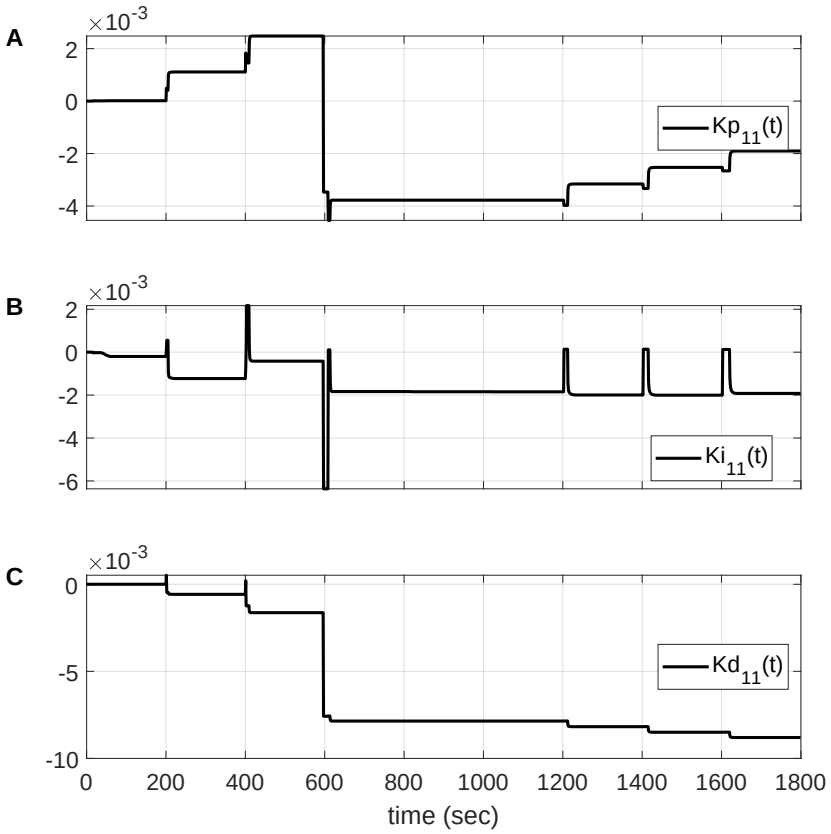


Figure 13. Evolution of the adaptive PID gains $K_{p,11}$ (**A**), $K_{i,11}$ (**B**) and $K_{d,11}$ (**C**) of the direct PID_{11} subcontroller in the nominal case for PID

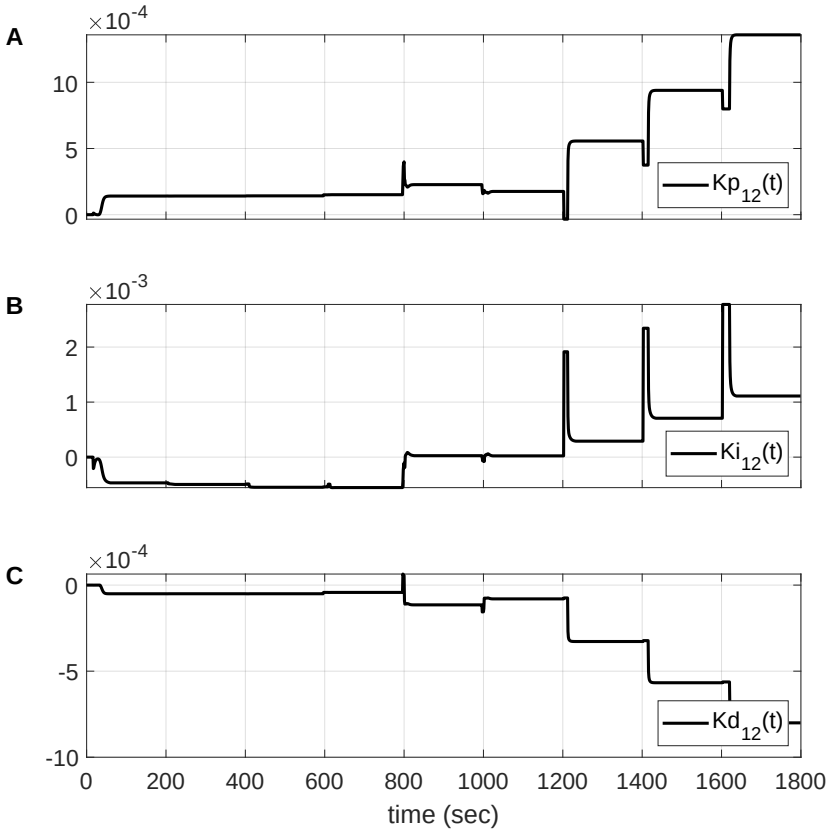


Figure 14. Evolution of the adaptive PID gains $K_{p,12}$ (**A**), $K_{i,12}$ (**B**) and $K_{d,12}$ (**C**) of the cross-coupling PID_{12} subcontroller in the nominal case for PID

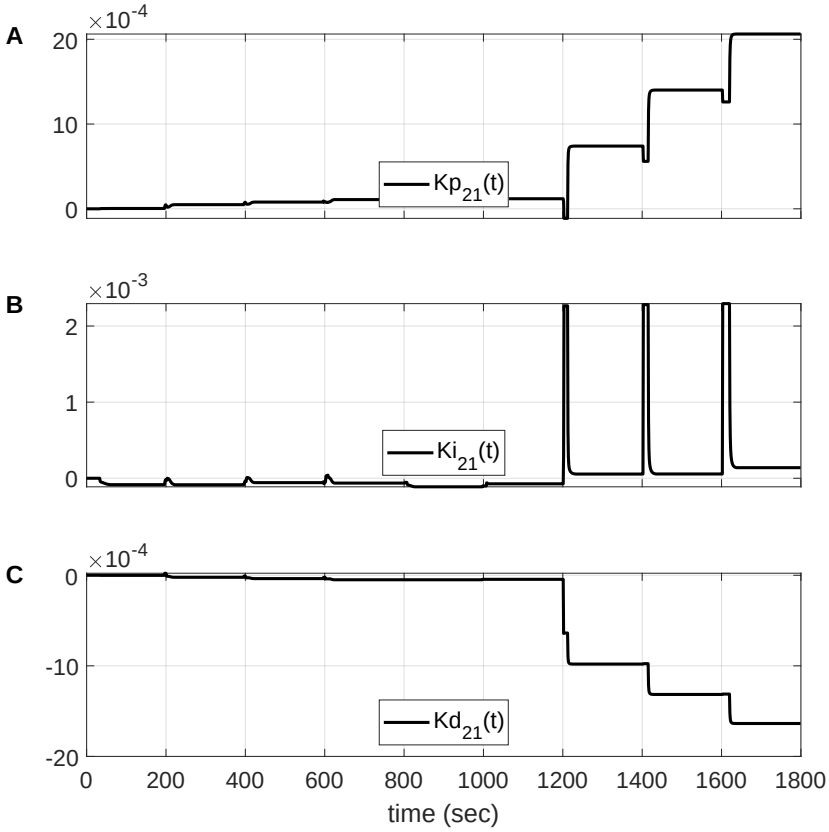


Figure 15. Evolution of the adaptive PID gains $K_{p,21}$ (**A**), $K_{i,21}$ (**B**) and $K_{d,21}$ (**C**) of the cross-coupling PID_{21} subcontroller in the nominal case for PID

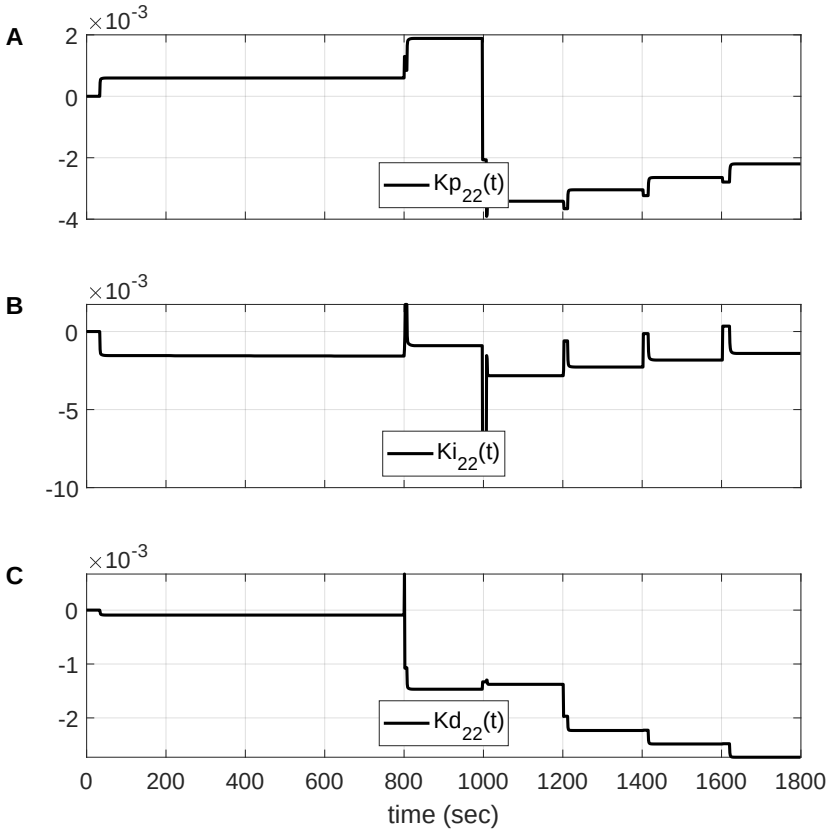


Figure 16. Evolution of the adaptive PID gains $K_{p,22}$ (**A**), $K_{i,22}$ (**B**) and $K_{d,22}$ (**C**) of the direct PID_{22} subcontroller in the nominal case for PID

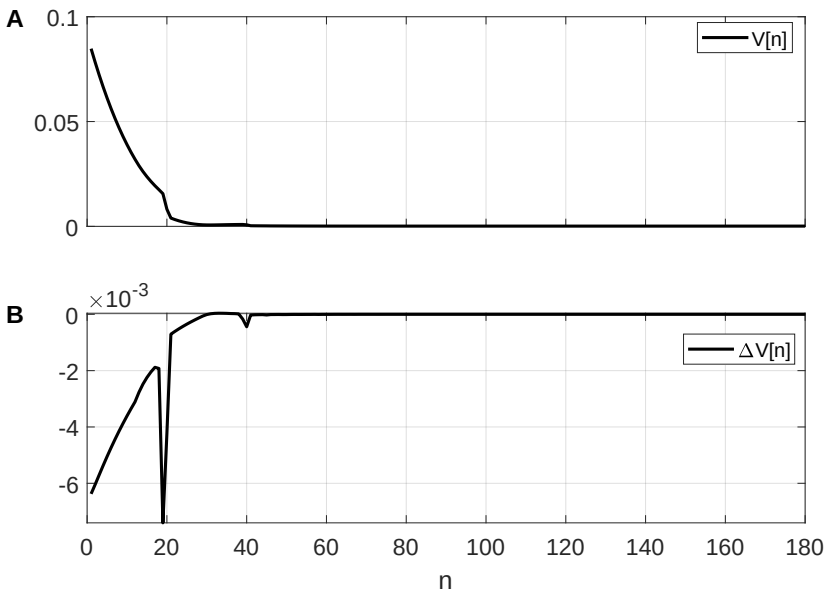


Figure 17. Lyapunov-based stability indicators under nominal operating conditions during the 0–180 s simulation period. (**A**) Evolution of the Lyapunov function $V[n]$. (**B**) Incremental Lyapunov difference $\Delta V[n]$, demonstrating the monotonic decrease of the Lyapunov function and the stability of the proposed adaptive fuzzy PID controller.

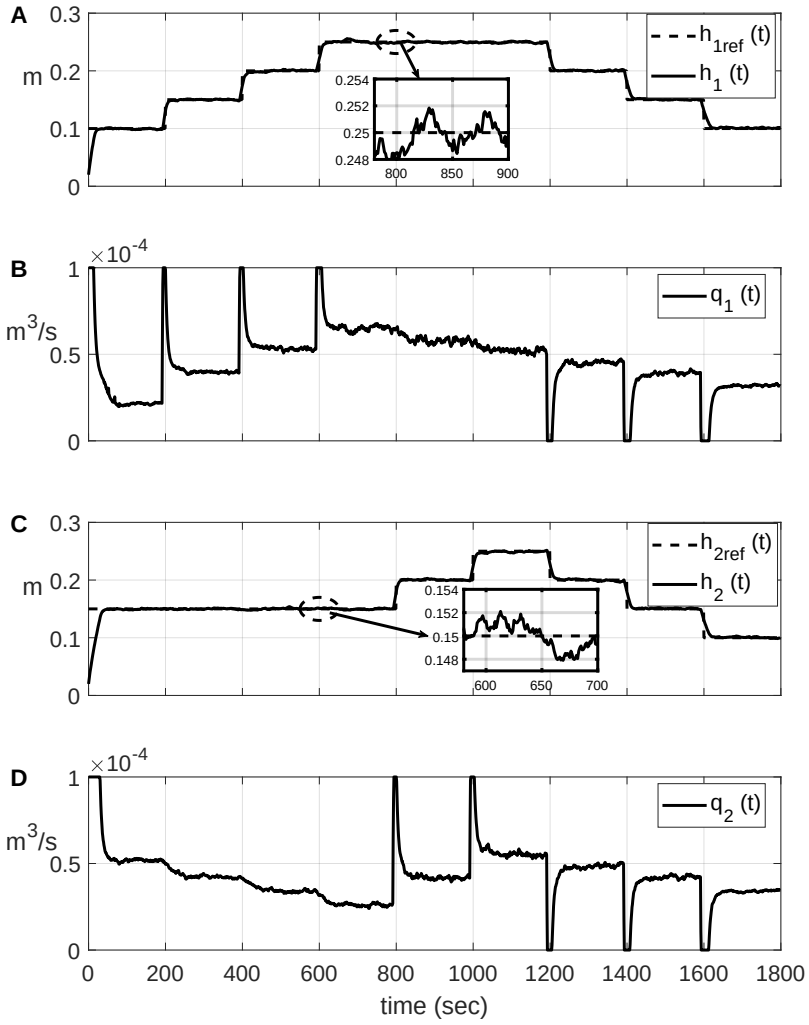


Figure 18. Comparison of the reference and measured liquid levels h_1 (**A**) and h_2 (**C**), and the corresponding control inputs q_1 (**B**) and q_2 (**D**) in the measurement noise case for fuzzy PID

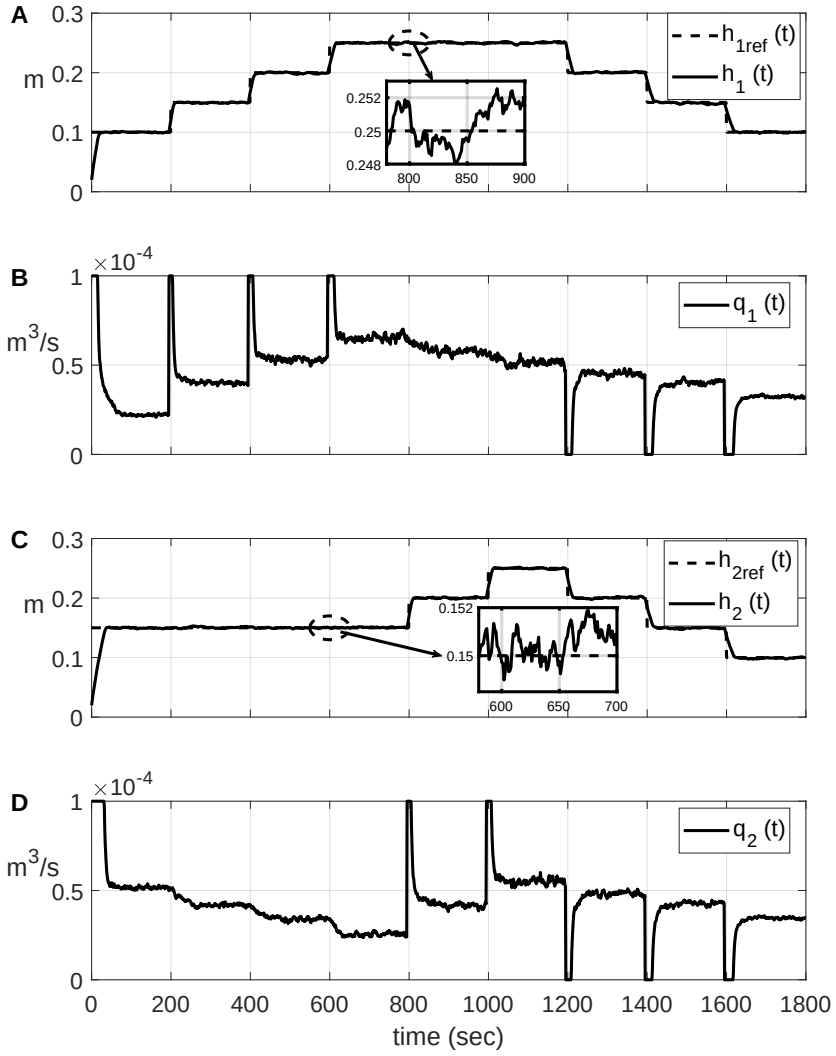


Figure 19. Comparison of the reference and measured liquid levels h_1 (**A**) and h_2 (**C**), and the corresponding control inputs q_1 (**B**) and q_2 (**D**) in the measurement noise case for PID

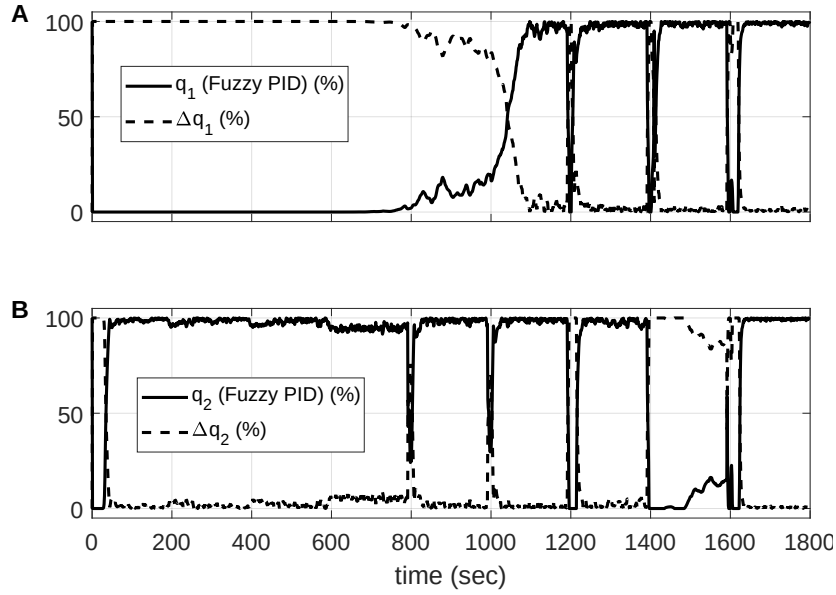


Figure 20. Comparison of the fuzzy PID and additive correction contributions to the control inputs q_1 (**A**) and q_2 (**B**) in the measurement noise case for fuzzy PID

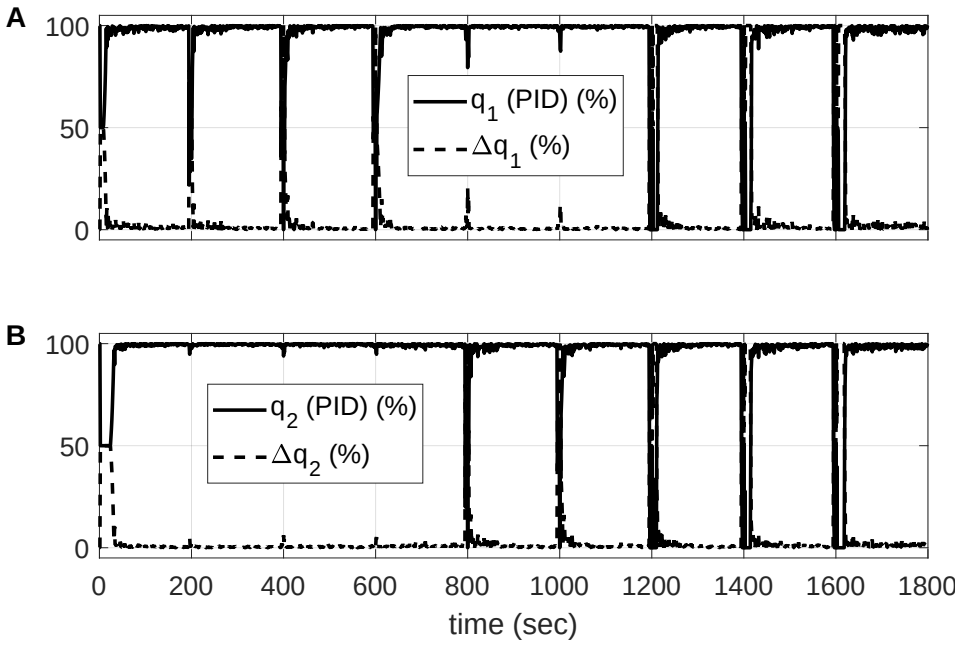


Figure 21. Comparison of the PID and additive correction contributions to the control inputs q_1 (**A**) and q_2 (**B**) in the measurement noise case for PID

Figures 22–25 present the adaptive scaling parameters of the four FPID subcontrollers. FPID_{11} and FPID_{22} represent the direct subcontrollers, whereas FPID_{12} and FPID_{21} compensate for the cross-coupling effects. The parameters respond mainly to the initial transient and reference changes, with additional small variations caused by the noisy feedback signals. All scaling parameters remain bounded throughout the simulation.

Figures 26–29 present the online evolution of the adaptive PID gains of the four subcontrollers. PID_{11} and PID_{22} represent the direct subcontrollers, whereas PID_{12} and PID_{21} compensate for the cross-coupling effects. The proportional, integral, and derivative gains are continuously adjusted in response to the reference transitions and exhibit additional fluctuations due to the noisy feedback signals. More pronounced gain variations occur at several reference changes; nevertheless, all gain trajectories remain bounded throughout the simulation.

Figure 30 presents the corresponding Lyapunov stability indicators under noisy feedback over the 0–180 s simulation period. Although measurement noise induces small residual oscillations around zero in $\Delta V[n]$, the Lyapunov function $V[n]$ remains bounded within a compact region throughout the simulation, consistent with the uniform ultimate boundedness (UUB) property established in Section 3.3.

Overall, the results demonstrate that both the RK model-based adaptive MIMO fuzzy PID controller and the benchmark adaptive MIMO PID controller preserve stable reference tracking under noisy feedback conditions. The bounded control inputs, adaptive parameters, and correction terms indicate that the direct and cross-coupling subcontrollers of both controller structures maintain regular adaptation behavior in the presence of measurement noise.

4.4. Parametric uncertainty case

In this case, measurement noise is not included. To introduce time-varying parametric uncertainty, the inter-tank flow coefficient az_{13} is varied sinusoidally[30] around its nominal value as

$$az_{13}[n] = 0.5 + 0.3\sin\left(\frac{2\pi n T_s}{500}\right). \qquad (101)$$

where T_s is the sampling period. The variation has a period of 500 s and drives the parameter over the interval $[0.2,\ 0.8]$, corresponding to a $\pm 60\%$ deviation from the nominal value of 0.5 on which the controllers are designed.

The variation profile in (101) is not available to either controller. For both control structures, the time-varying parameter is estimated online using the same RK-based identification mechanism, and the corresponding prediction model is updated at each sampling instant. This scenario

comparatively evaluates the tracking, parameter-estimation, controller-adaptation, and control-input behavior of both controllers under the same time-varying plant–model mismatch.

Figures 31 and **32** present the tracking responses, control inputs, and online parameter-estimation results obtained using the adaptive MIMO fuzzy PID and adaptive MIMO PID controllers, respectively, under time-varying parametric uncertainty. For both controllers, the liquid levels $h_1(t)$ and $h_2(t)$ closely follow their reference trajectories despite the continuous variation in the inter-tank flow coefficient az_{13}. Small tracking deviations occur during the initial transient and during variations in az_{13}. Nevertheless, the tracking responses remain stable and bounded, while the control inputs $q_1(t)$ and $q_2(t)$ remain within their specified limits throughout the simulation. Moreover, the estimated parameter closely follows its actual time-varying value, demonstrating the effectiveness of the online parameter-estimation mechanism for both control structures.

The parameter-estimation results presented in panel E of both **Figures 31** and **32** show that the online estimate $\widehat{az}_{13}[n]$ closely tracks the sinusoidal variation of the actual parameter $az_{13}[n]$ over the entire [0.2, 0.8] range, with a slight lag near the extrema of the sinusoidal trajectory. Therefore, the RK-based identification mechanism updates the prediction model according to the changing dynamics and reduces the resulting plant–model mismatch.

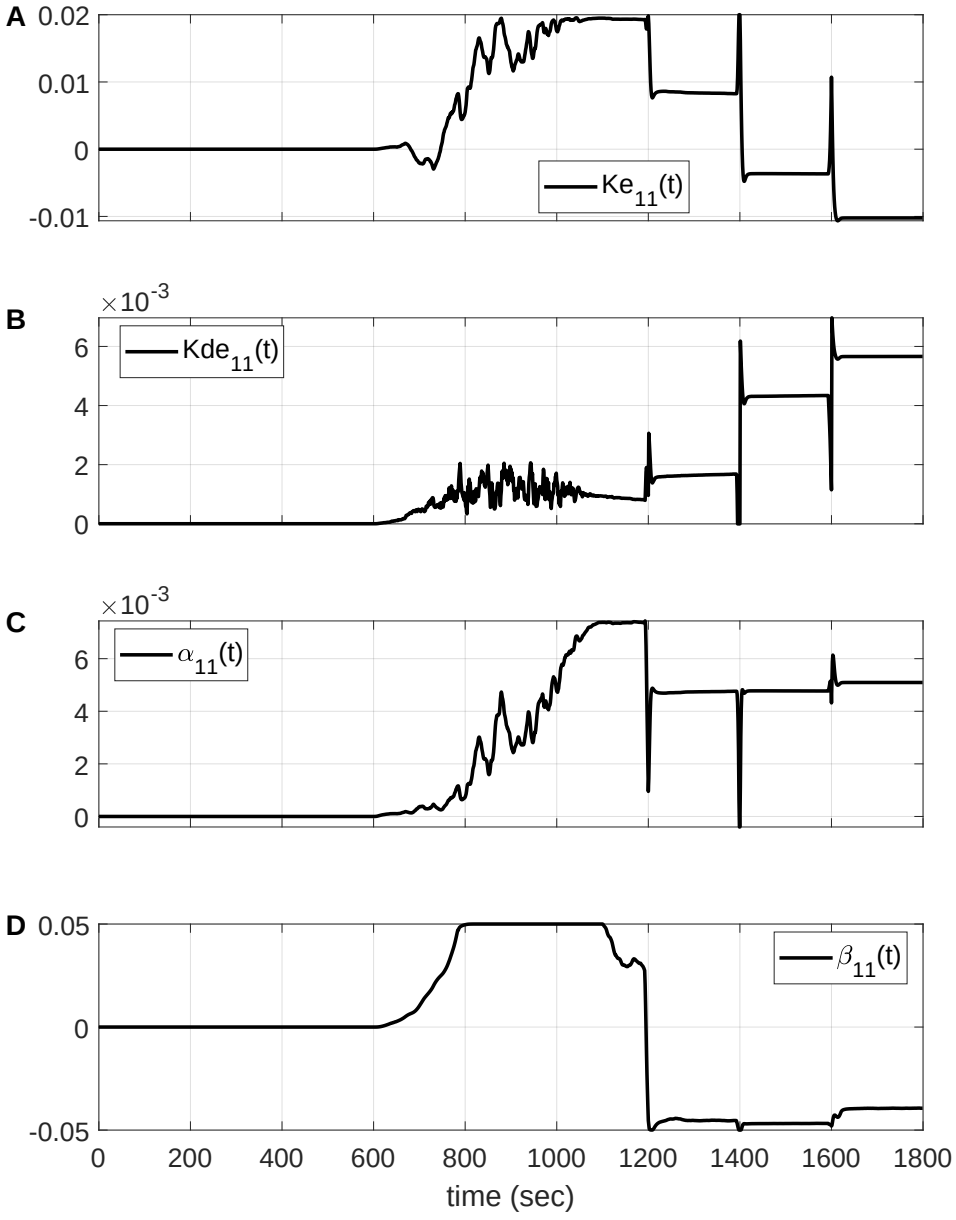


Figure 22. Evolution of the adaptive scaling parameters $K_{e,11}$ (**A**), $K_{de,11}$ (**B**), α_{11} (**C**) and β_{11} (**D**) of the direct FPID_{11} subcontroller in the measurement noise case

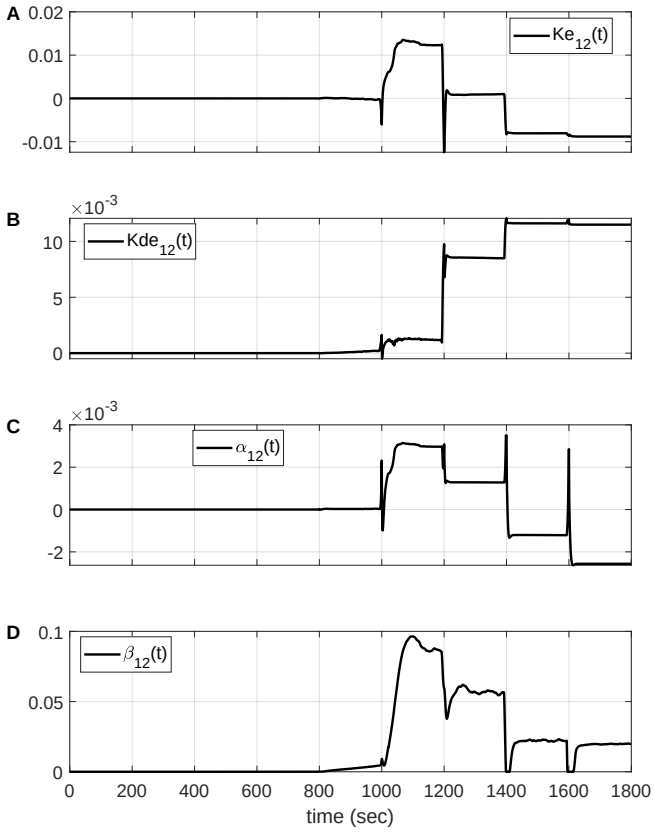


Figure 23. Evolution of the adaptive scaling parameters $K_{e,12}$ (**A**), $K_{de,12}$ (**B**), α_{12} (**C**) and β_{12} (**D**) of the cross-coupling FPID_{12} subcontroller in the measurement noise case

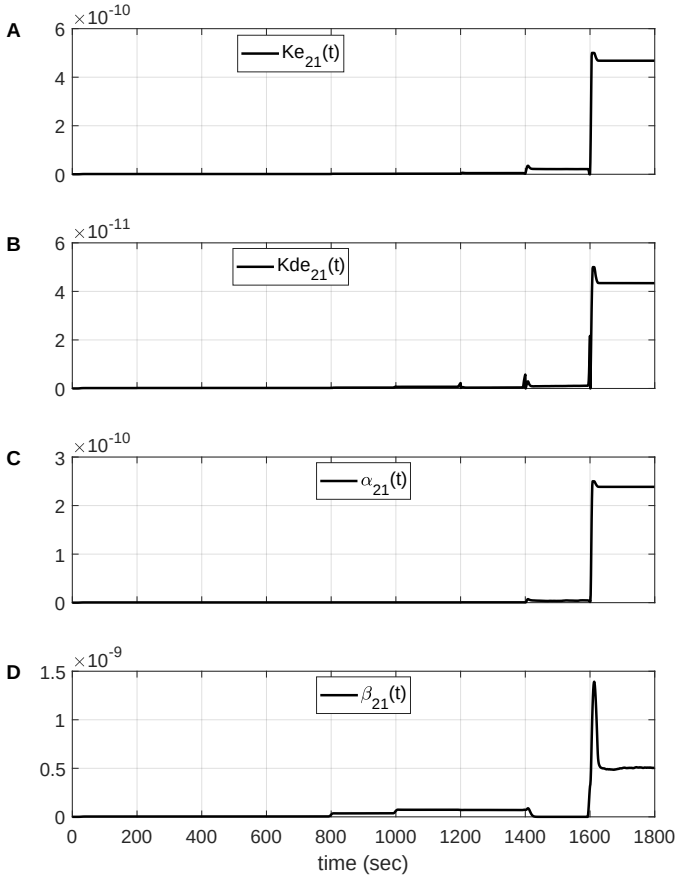


Figure 24. Evolution of the adaptive scaling parameters $K_{e,21}$ (**A**), $K_{de,21}$ (**B**), α_{21} (**C**) and β_{21} (**D**) of the cross-coupling FPID_{21} subcontroller in the measurement noise case

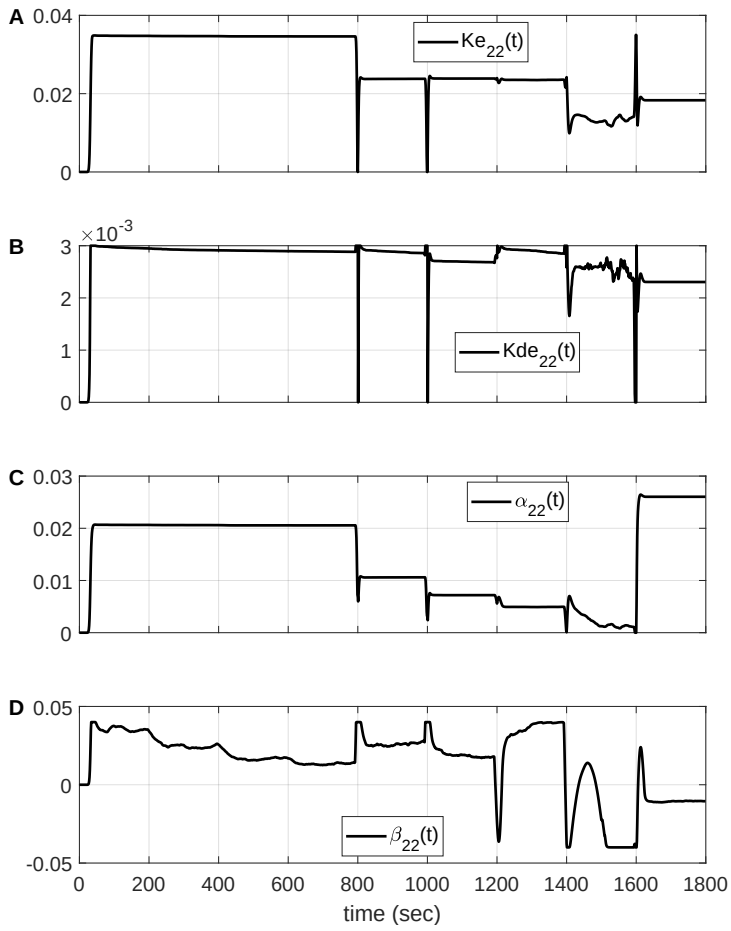


Figure 25. Evolution of the adaptive scaling parameters $K_{e,22}$ (**A**), $K_{de,22}$ (**B**), α_{22} (**C**) and β_{22} (**D**) of the direct FPID_{22} subcontroller in the measurement noise case

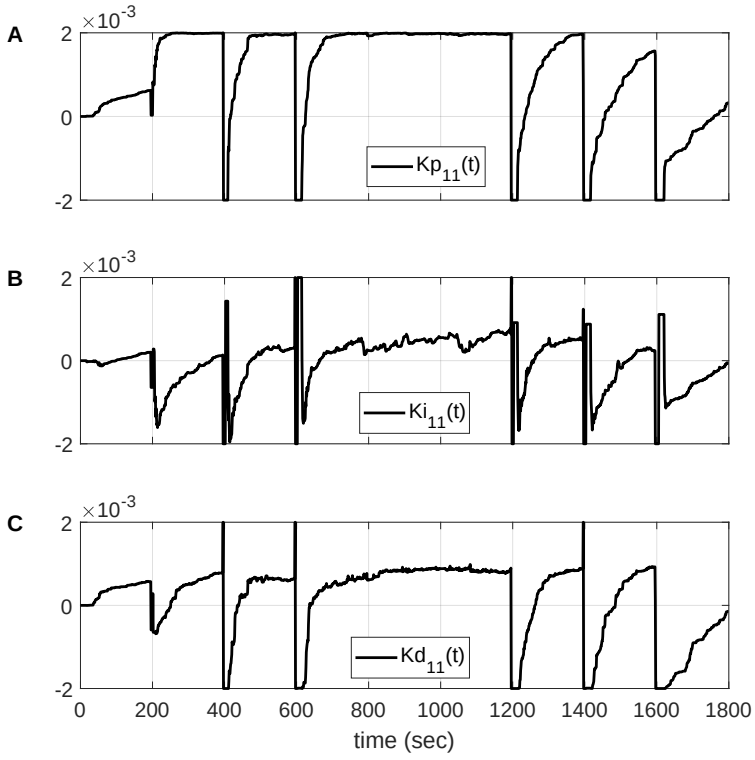


Figure 26. Evolution of the adaptive PID gains $K_{p,11}$ (**A**), $K_{i,11}$ (**B**) and $K_{d,11}$ (**C**) of the direct PID_{11} subcontroller in the measurement noise case

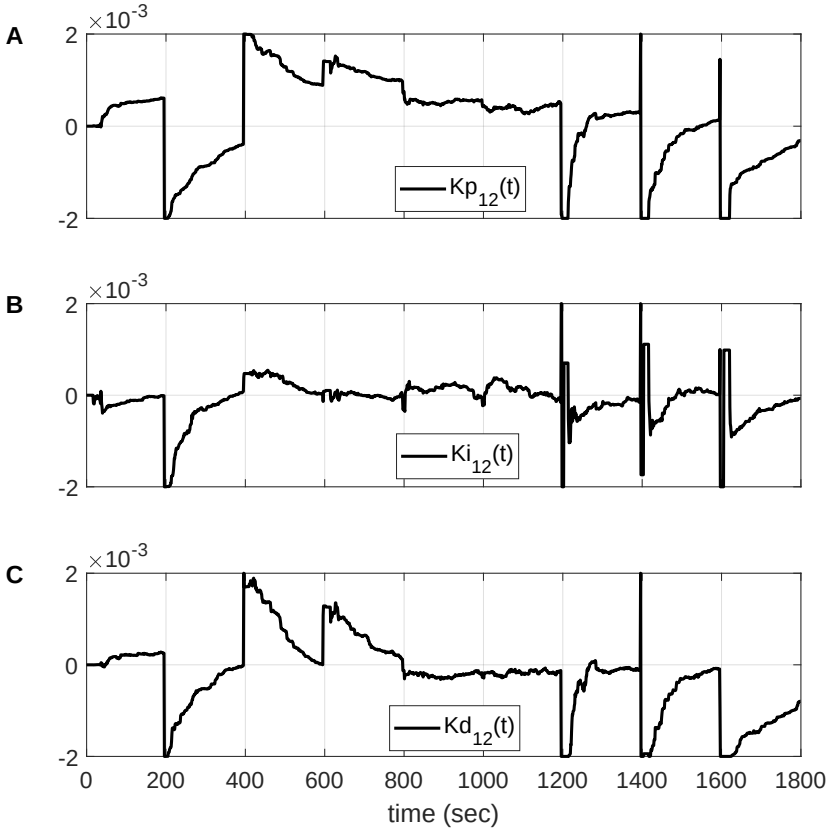


Figure 27. Evolution of the adaptive PID gains $K_{p,12}$ (**A**), $K_{i,12}$ (**B**) and $K_{d,12}$ (**C**) of the cross-coupling PID_{12} subcontroller in the measurement noise case

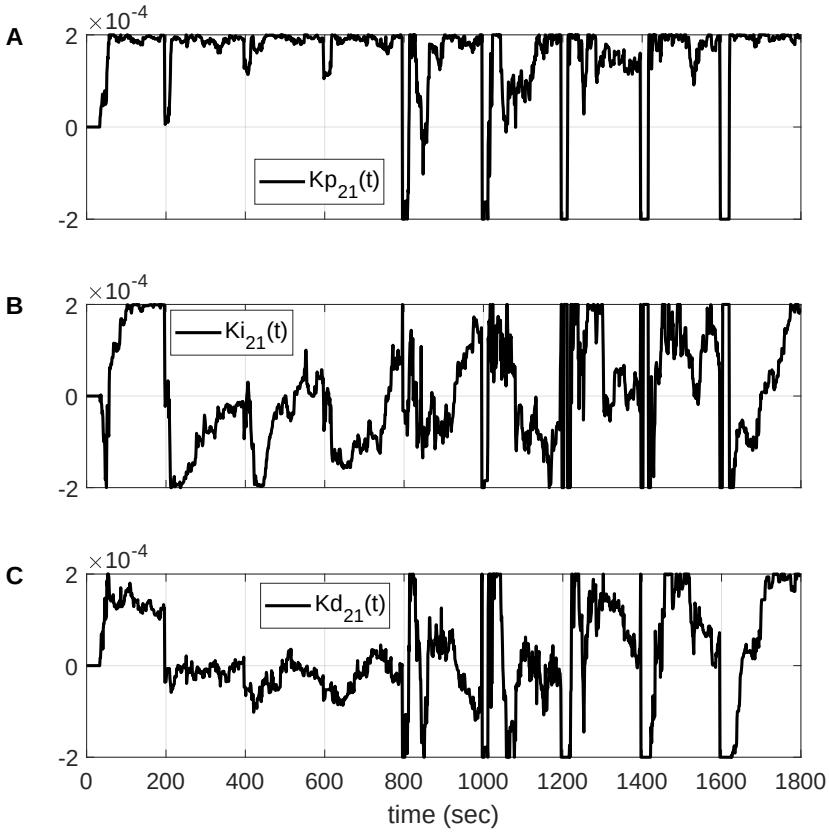


Figure 28. Evolution of the adaptive PID gains $K_{p,21}$ (**A**), $K_{i,21}$ (**B**) and $K_{d,21}$ (**C**) of the cross-coupling PID_{21} subcontroller in the measurement noise case

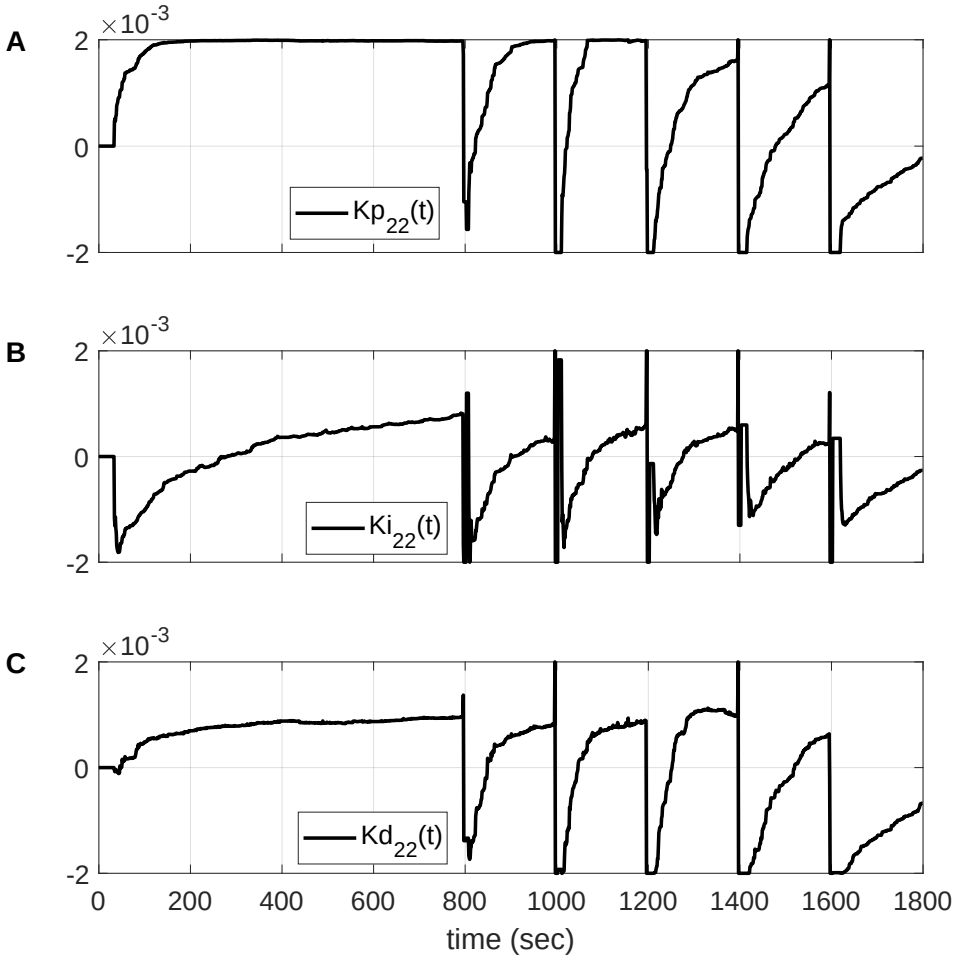


Figure 29. Evolution of the adaptive PID gains $K_{p,22}$ (**A**), $K_{i,22}$ (**B**) and $K_{d,22}$ (**C**) of the direct PID_{22} subcontroller in the measurement noise case

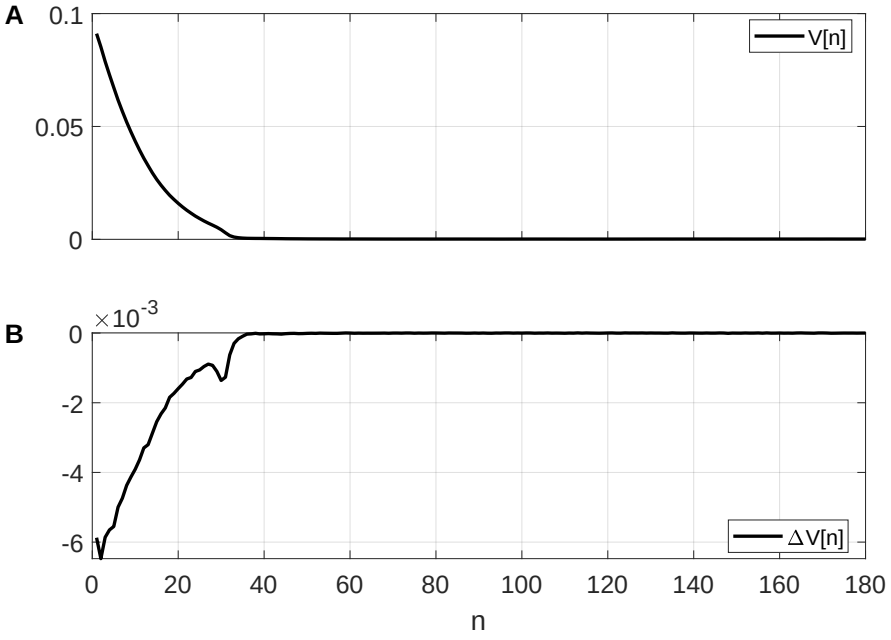


Figure 30. Lyapunov-based stability indicators under measurement noise operating conditions over the 0–180 s simulation interval. (**A**) Evolution of the Lyapunov function $V[n]$. (**B**) Incremental Lyapunov difference $\Delta V[n]$, illustrating the decreasing Lyapunov trend and stability of the proposed adaptive fuzzy PID controller.

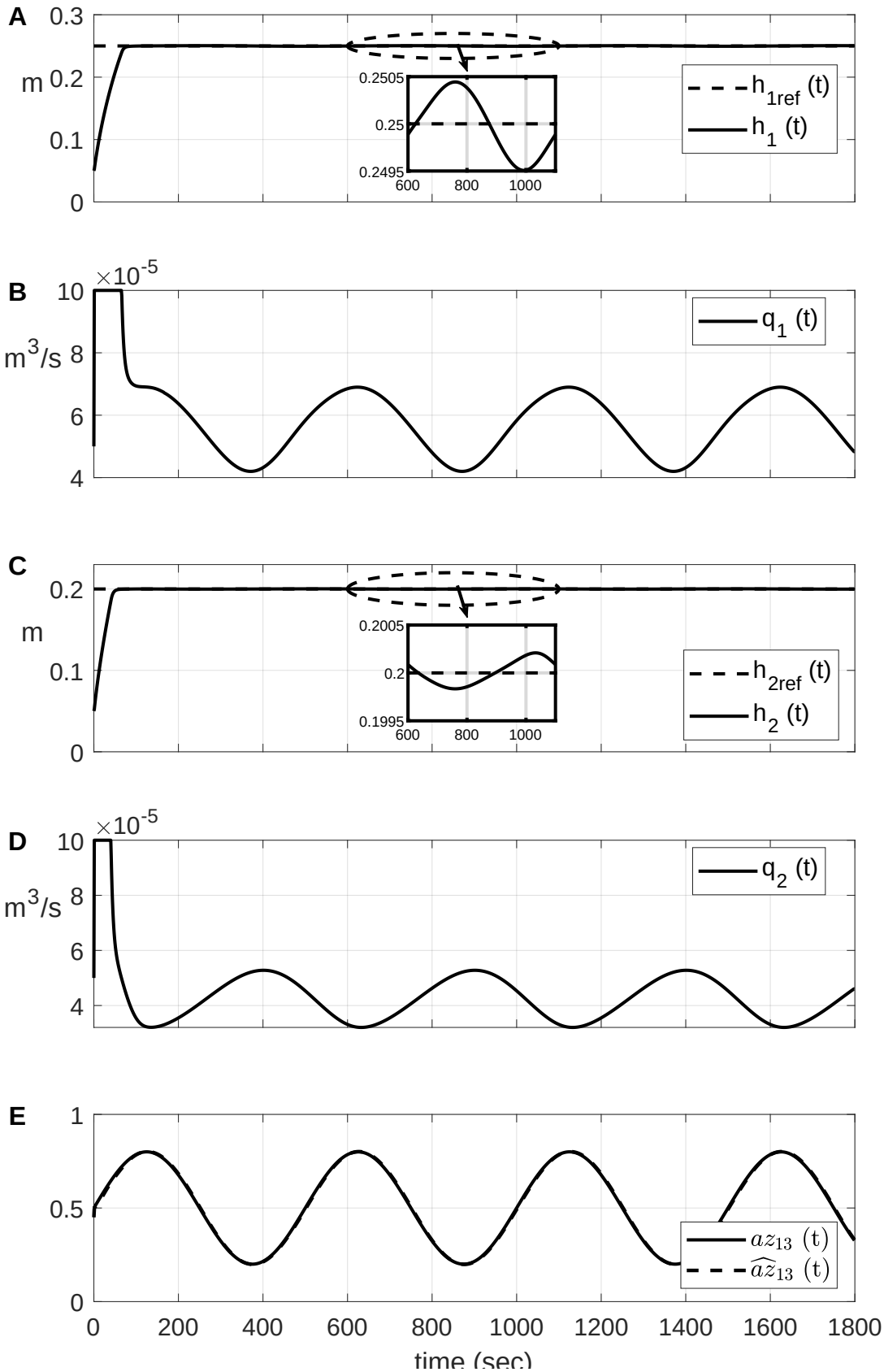


Figure 31. Comparison of the reference and actual liquid levels h_1 (**A**) and h_2 (**C**), the corresponding control inputs q_1 (**B**) and q_2 (**D**), and the actual time-varying flow coefficient az_{13} and its online estimate $\widehat{az}_{13}$ (**E**) in the parametric uncertainty case for fuzzy PID

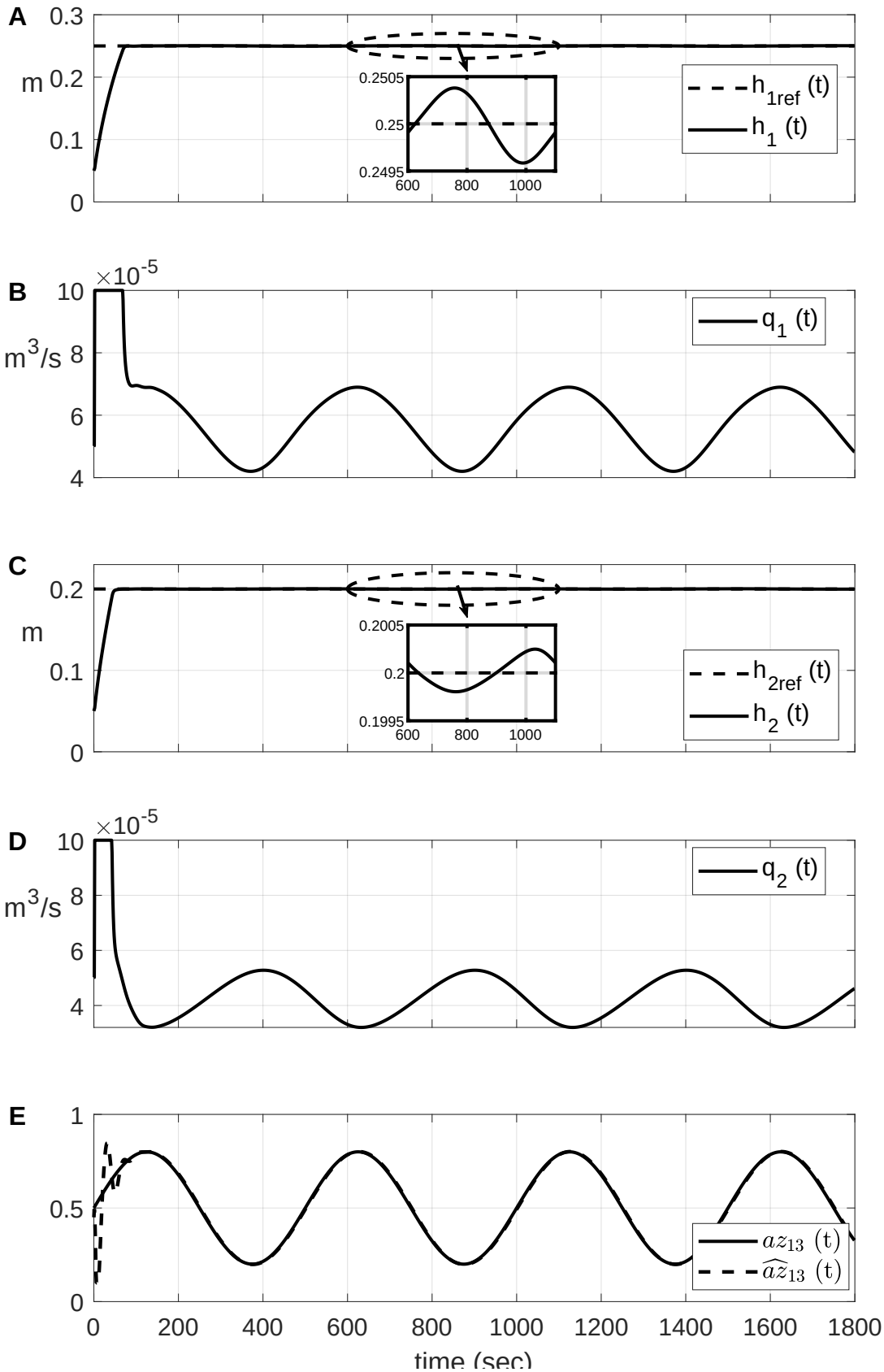


Figure 32. Comparison of the reference and actual liquid levels h_1 (**A**) and h_2 (**C**), the corresponding control inputs q_1 (**B**) and q_2 (**D**), and the actual time-varying flow coefficient az_{13} and its online estimate $\widehat{az}_{13}$ (**E**) in the parametric uncertainty case for PID

Figures 33 and **34** present the percentage contributions of the main controller outputs and the additive correction terms for the adaptive fuzzy PID and adaptive PID controllers, respectively. For both control structures, the correction mechanism provides supplementary control action, particularly during the initial transient and in response to variations in az_{13}. Although the relative contributions vary with the time-varying parametric uncertainty, the adapted fuzzy PID and PID controllers generate the dominant part of the total control action over most of the simulation period.

Figures 35–38 show the adaptive scaling parameters of the four FPID subcontrollers. FPID_{11} and FPID_{22} represent the direct subcontrollers, whereas FPID_{12} and FPID_{21} compensate for the cross-coupling effects. The scaling parameters respond to the reference changes and variations in az_{13} but remain bounded throughout the simulation.

Figures 39–42 present the online evolution of the adaptive proportional, integral, and derivative gains of the four PID subcontrollers under time-varying parametric uncertainty. PID_{11} and PID_{22} represent the direct subcontrollers, whereas PID_{12} and PID_{21} compensate for the cross-coupling effects. The PID gains undergo their most pronounced adjustments during the initial transient and subsequently evolve smoothly in response to the reference changes and variations in az_{13}. Most gain trajectories approach constant or slowly varying bounded levels. No persistent gain oscillation or continuous parameter drift is observed throughout the simulation.

Figure 43 presents the evolution of $V[n]$ and $\Delta V[n]$ over the 0–180 s simulation period under time-varying parametric uncertainty. Despite the severe $\pm 60\%$ continuous variations in az_{13}, the Lyapunov function $V[n]$ dissipates rapidly after reference transitions and remains uniformly bounded within the predicted UUB region throughout the entire simulation horizon, further supporting the theoretical stability guarantees established in Section 3.3.

Overall, the results indicate that both the proposed RK model-based adaptive MIMO fuzzy PID controller and the benchmark RK model-based adaptive MIMO PID controller maintain stable and bounded tracking behavior under time-varying parametric uncertainty. For both control structures, the online estimation of az_{13} reduces the plant–model mismatch, while the bounded adaptation of the direct and cross-coupling subcontrollers compensates for the resulting variations in the coupled nonlinear flow dynamics.

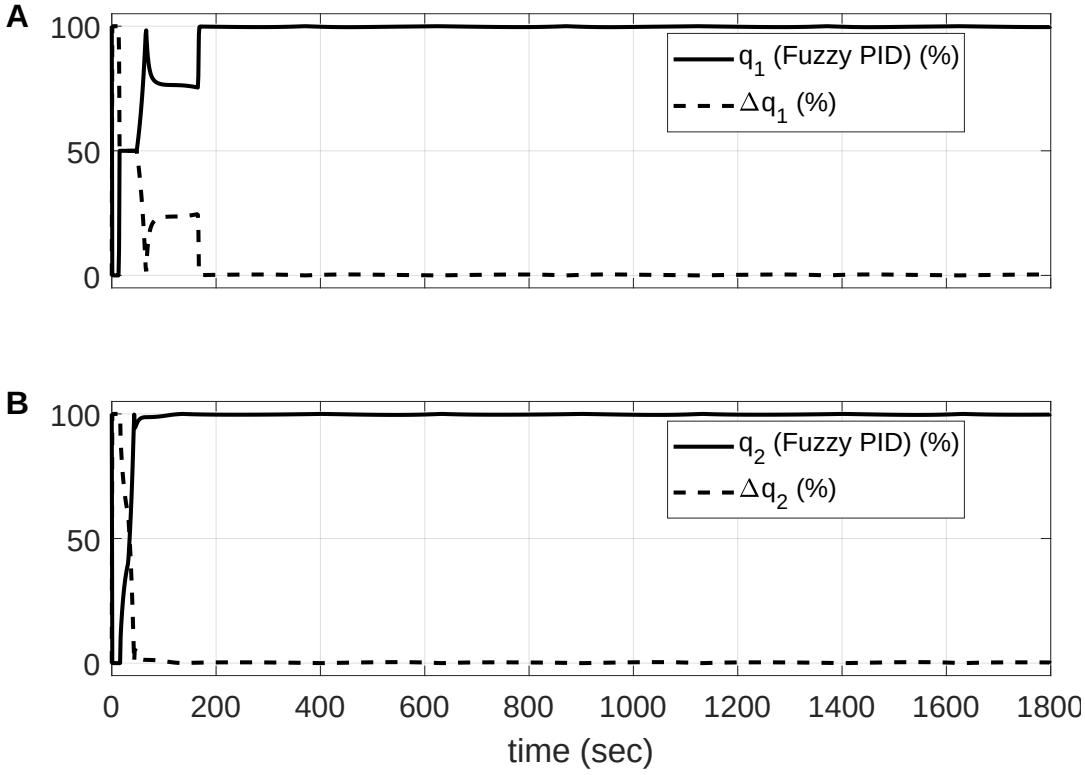


Figure 33. Comparison of the fuzzy PID and additive correction contributions to the control inputs q_1 (**A**) and q_2 (**B**) in the parametric uncertainty case for fuzzy PID

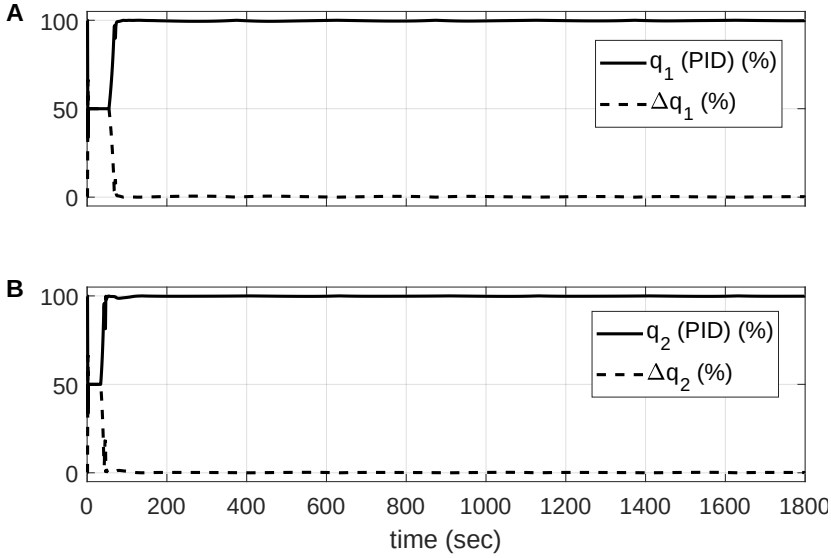


Figure 34. Comparison of the PID and additive correction contributions to the control inputs q_1 (**A**) and q_2 (**B**) in the parametric uncertainty case for PID

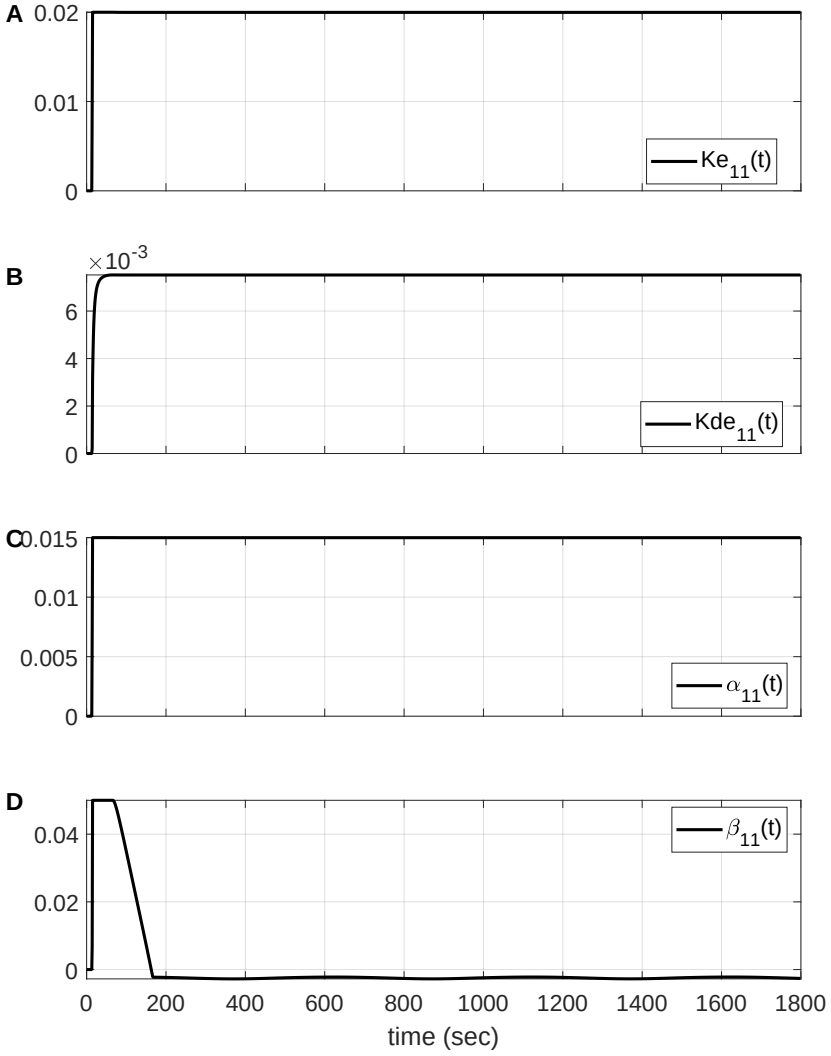


Figure 35. Evolution of the adaptive scaling parameters $K_{e,11}$ (**A**), $K_{de,11}$ (**B**), α_{11} (**C**) and β_{11} (**D**) of the direct FPID_{11} subcontroller in the parametric uncertainty case

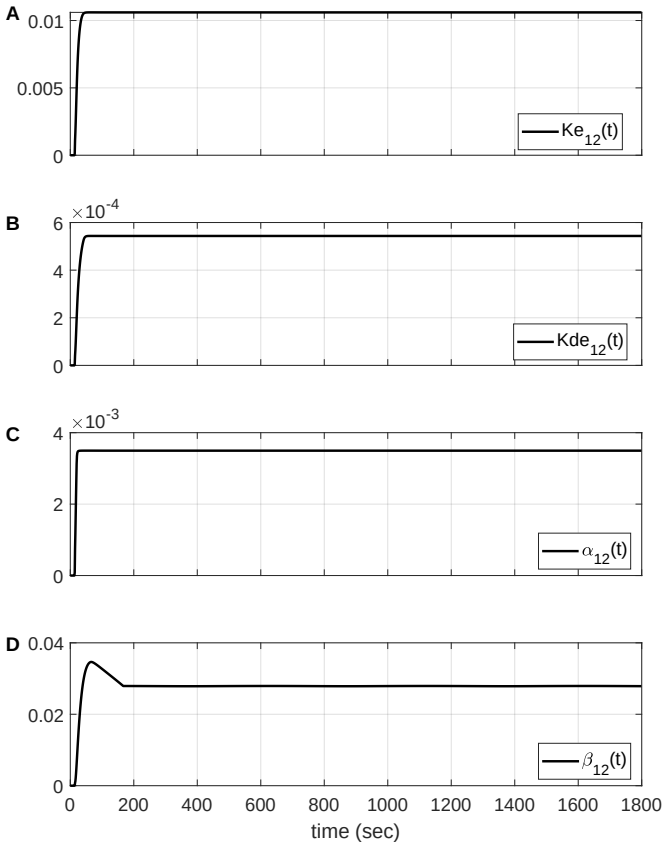


Figure 36. Evolution of the adaptive scaling parameters $K_{e,12}$ (**A**), $K_{de,12}$ (**B**), α_{12} (**C**) and β_{12} (**D**) of the cross-coupling FPID_{12} subcontroller in the parametric uncertainty case

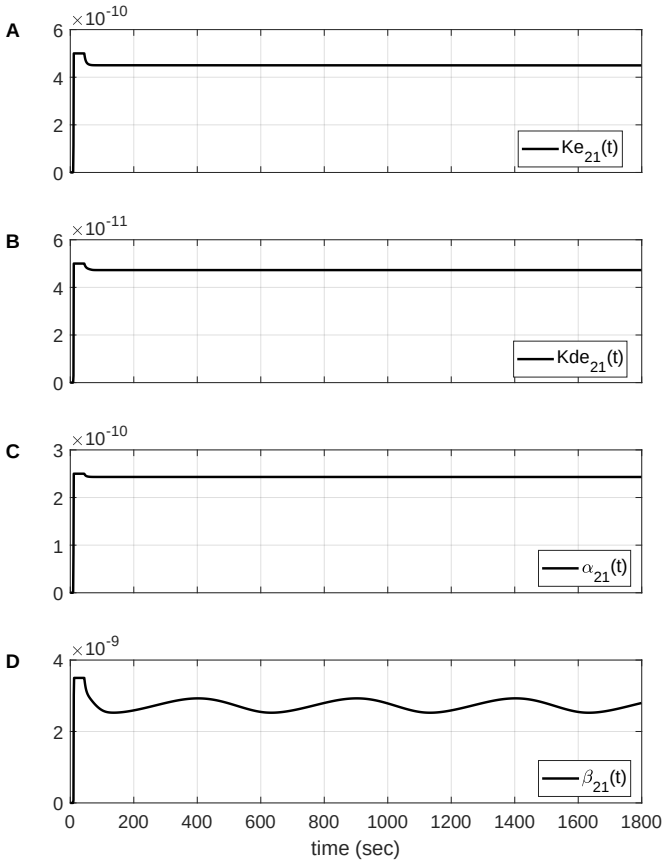


Figure 37. Evolution of the adaptive scaling parameters $K_{e,21}$ (**A**), $K_{de,21}$ (**B**), α_{21} (**C**) and β_{21} (**D**) of the cross-coupling FPID_{21} subcontroller in the parametric uncertainty case

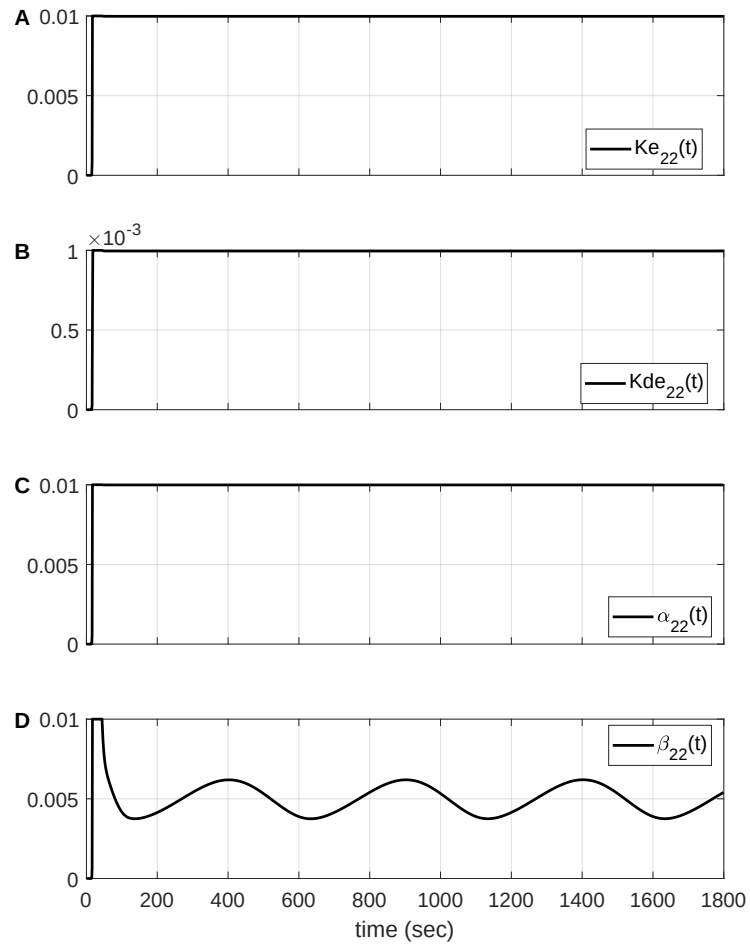


Figure 38. Evolution of the adaptive scaling parameters $K_{e,22}$ (**A**), $K_{de,22}$ (**B**), α_{22} (**C**) and β_{22} (**D**) of the direct FPID_{22} subcontroller in the parametric uncertainty case

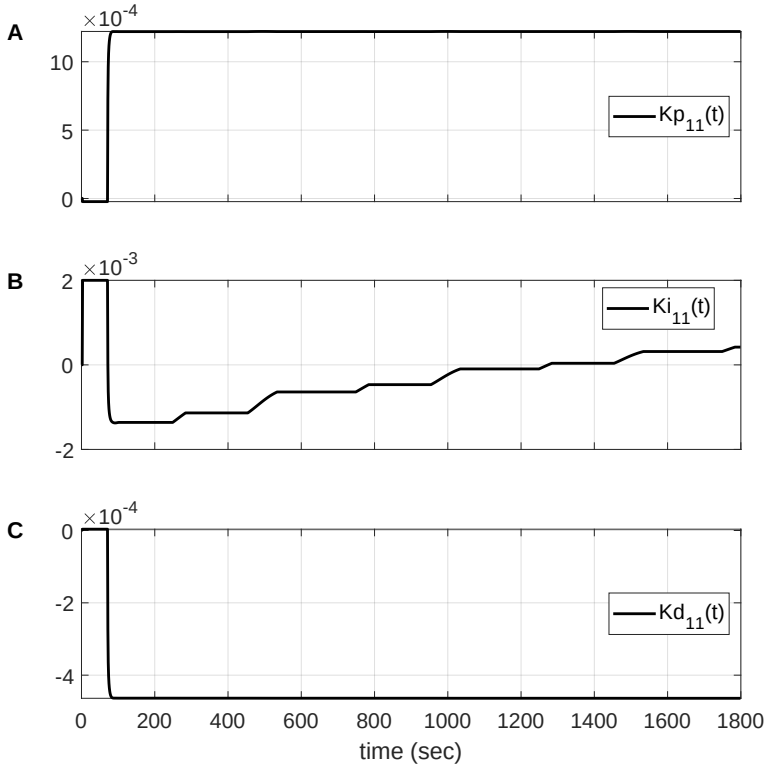


Figure 39. Evolution of the adaptive PID gains $K_{p,11}$ (**A**), $K_{i,11}$ (**B**) and $K_{d,11}$ (**C**) of the direct PID_{11} subcontroller in the parametric uncertainty case for PID

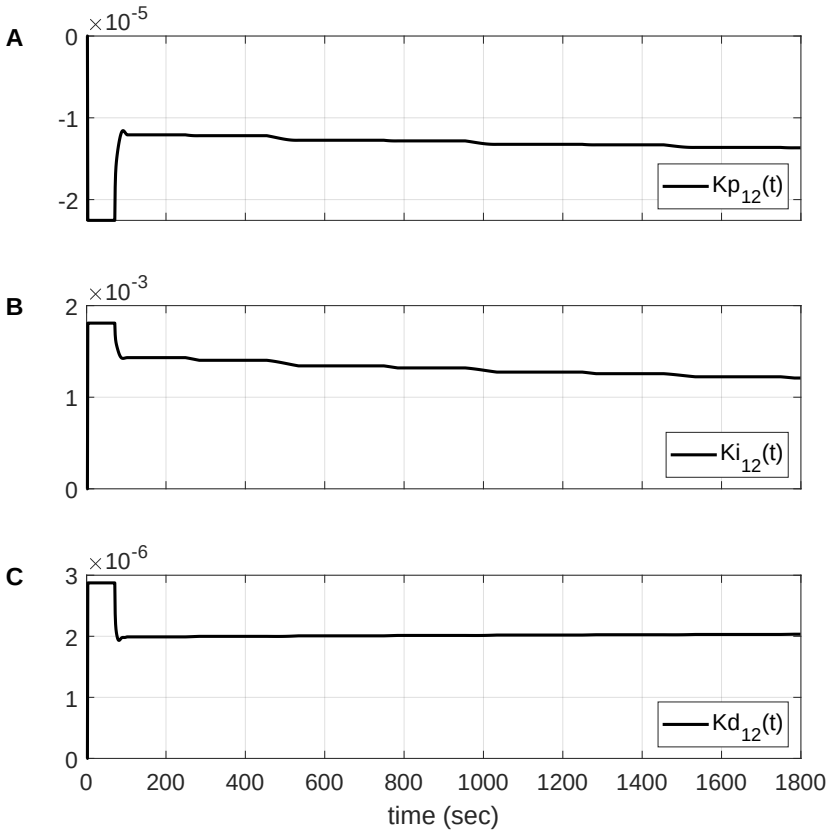


Figure 40. Evolution of the adaptive PID gains $K_{p,12}$ (**A**), $K_{i,12}$ (**B**) and $K_{d,12}$ (**C**) of the cross-coupling PID_{12} subcontroller in the parametric uncertainty case for PID

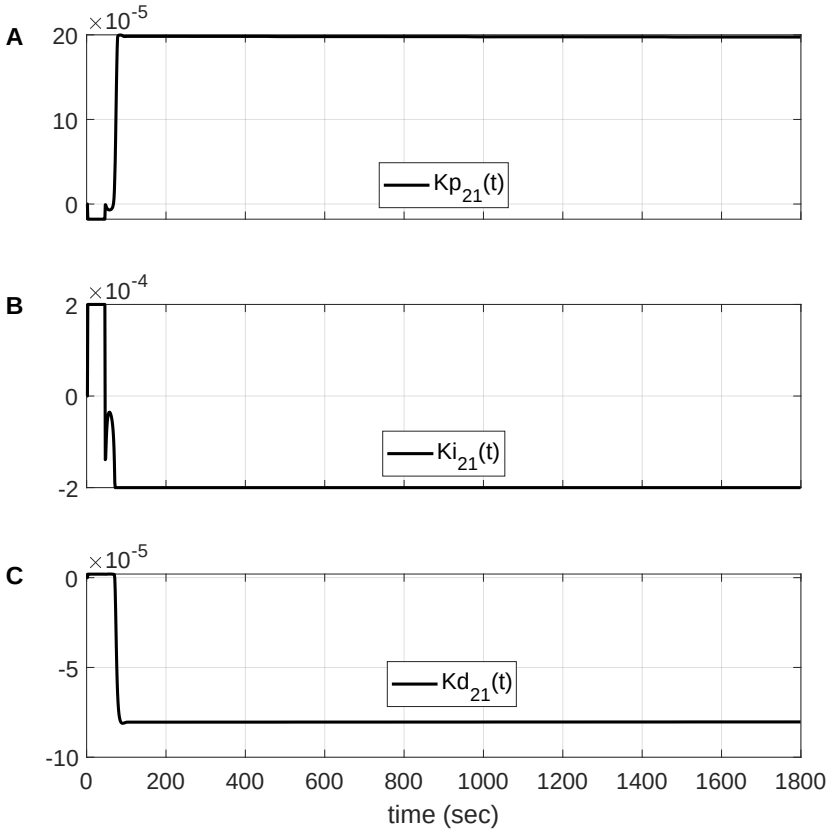


Figure 41. Evolution of the adaptive PID gains $K_{p,21}$ (**A**), $K_{i,21}$ (**B**) and $K_{d,21}$ (**C**) of the cross-coupling PID_{21} subcontroller in the parametric uncertainty case for PID

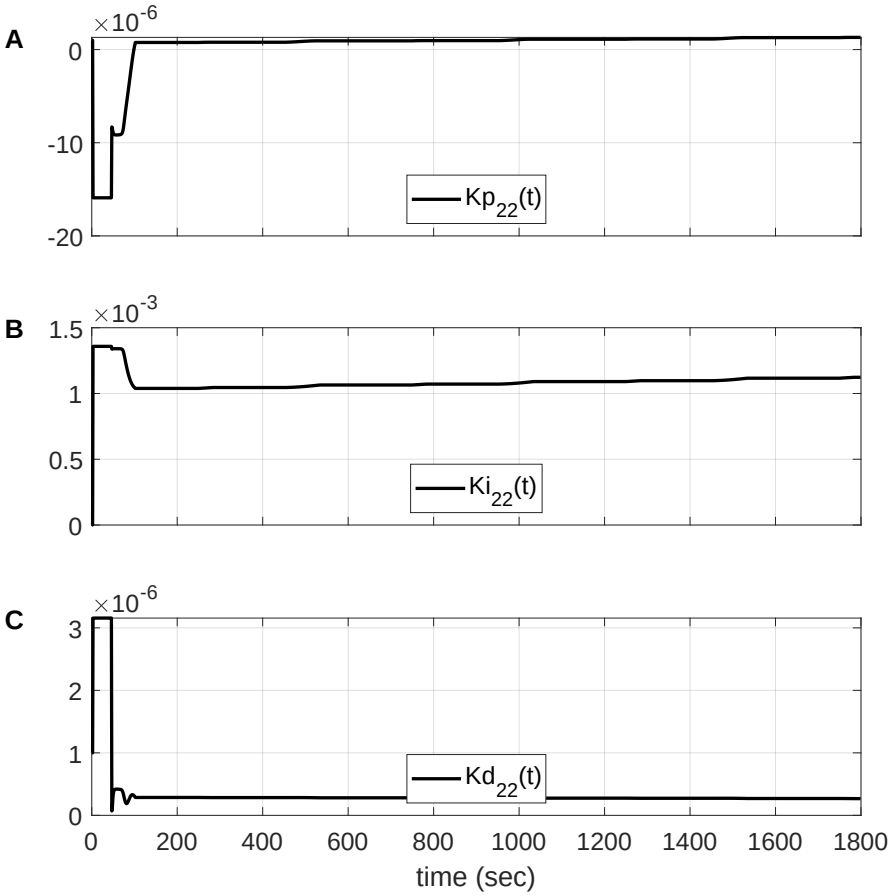


Figure 42. Evolution of the adaptive PID gains $K_{p,22}$ (**A**), $K_{i,22}$ (**B**) and $K_{d,22}$ (**C**) of the direct PID_{22} subcontroller in the parametric uncertainty case for PID

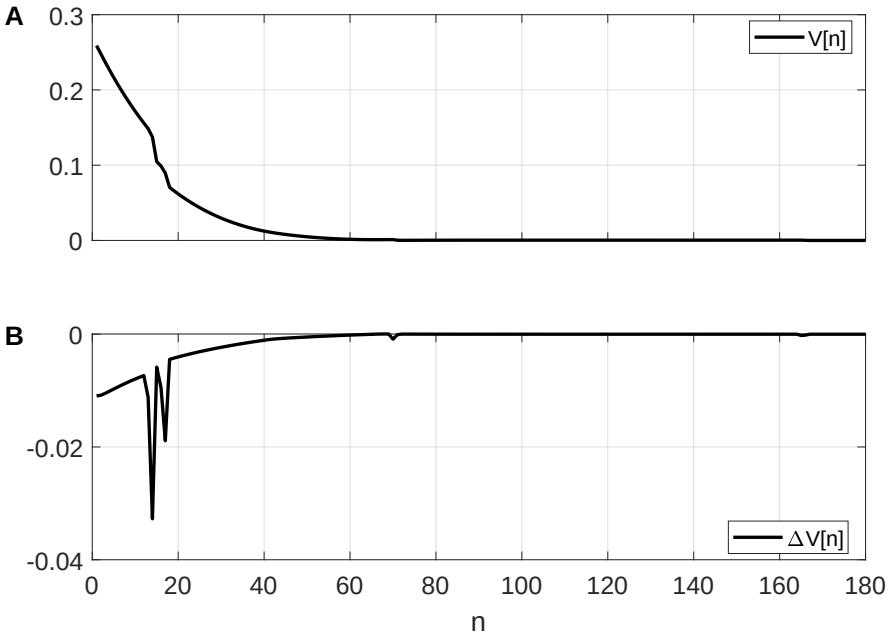


Figure 43. Lyapunov-based stability indicators under parametric uncertainty operating conditions over the 0–180 s simulation interval. (**A**) Evolution of the Lyapunov function $V[n]$. (**B**) Incremental Lyapunov difference $\Delta V[n]$, illustrating the decreasing Lyapunov trend and stability of the proposed adaptive fuzzy PID controller.

4.5. Overall performance evaluation

The overall performance of the proposed RK model-based adaptive MIMO fuzzy PID controller and the benchmark RK model-based adaptive MIMO PID controller are evaluated using the same composite performance index, J_{comp}. The index considers both the liquid-level tracking errors and the variations in the control inputs.[3,32] The same simulation horizon, reference trajectories, weighting coefficients, and performance definition are used for both controllers.

$$J_{\text{comp}} = \sum_{n=1}^{N_{\text{sim}}} \Big[\big(h_1[n] - h_{1,\text{ref}}[n]\big)^2 + \big(h_2[n] - h_{2,\text{ref}}[n]\big)^2 \Big] + \sum_{n=2}^{N_{\text{sim}}} \Big[\lambda_1 \Delta q_1^2[n] + \lambda_2 \Delta q_2^2[n] \Big]. \tag{102}$$

The first term of (102) represents the tracking cost, whereas the second term represents the control-variation cost. Therefore, a lower value of J_{comp} indicates a better balance between tracking accuracy and control smoothness.

Tables 4 and **5** compare the overall performance of the RK model-based adaptive MIMO fuzzy PID controller and the adaptive MIMO PID controller, respectively. In all simulation cases, the adaptive MIMO fuzzy PID controller achieves a lower J_{comp} value than the adaptive MIMO PID controller. Specifically, the fuzzy PID controller reduces the performance index from 0.2833 to 0.2660 in the nominal case, corresponding to a 6.11% improvement. In the measurement-noise case, J_{comp} decreases from 0.2877 to 0.2701, yielding a 6.12% improvement. Under time-varying parametric uncertainty, the performance index is reduced from 1.1139 to 1.0739, corresponding to a 3.59% improvement.

Both controllers attain their lowest performance indices under nominal conditions and exhibit only a limited performance degradation in the presence of measurement noise. The higher J_{comp} values obtained under parametric uncertainty result from variations in the inter-tank flow coefficient az_{13}, which alter the plant dynamics and require larger tracking and control adjustments. Nevertheless, the consistently lower performance indices demonstrate that the adaptive MIMO fuzzy PID controller provides better combined tracking and control-variation performance than the adaptive MIMO PID controller in all three simulation cases.

Table 4. Overall performance of the RK model-based adaptive MIMO fuzzy PID controller

Simulation case	$\boldsymbol{J_{\text{comp}}}$
Nominal case	0.2660
Measurement noise case	0.2701
Parametric uncertainty case	1.0739

Table 5. Overall performance of the RK model-based adaptive MIMO PID controller

Simulation case	$\boldsymbol{J_{\text{comp}}}$
Nominal case	0.2833
Measurement noise case	0.2877
Parametric uncertainty case	1.1139

Tables 6 and **7** compare the average computation time per control cycle of the RK model-based adaptive MIMO fuzzy PID controller and the adaptive MIMO PID controller, respectively. In all simulation cases, the adaptive MIMO PID controller requires less computation time than the adaptive MIMO fuzzy PID controller. In the nominal case, the average cycle time decreases from 0.1297 ms to 0.0770 ms, corresponding to a 40.63% reduction. In the measurement-noise case, it decreases from 0.1485 ms to 0.0998 ms, yielding a 32.79% reduction. Under time-varying parametric uncertainty, the average cycle time is reduced from 0.1774 ms to 0.1269 ms, corresponding to a 28.47% reduction.

Table 6. Average computation time per control cycle of the RK model-based adaptive MIMO fuzzy PID controller

Simulation case	**Average cycle time**
Nominal case	0.1297 ms
Measurement-noise case	0.1485 ms
Parametric-uncertainty case	0.1774 ms

Table 7. Average computation time per control cycle of the RK model-based adaptive MIMO PID controller

Simulation case	**Average cycle time**
Nominal case	0.0770 ms
Measurement-noise case	0.0998 ms
Parametric-uncertainty case	0.1269 ms

For both controllers, the nominal case has the lowest computational cost, whereas the

parametric-uncertainty case has the highest average cycle time because it additionally involves online estimation of the time-varying inter-tank flow coefficient az_{13}. The higher computation times of the adaptive MIMO fuzzy PID controller are mainly associated with the additional fuzzy inference operations. Nevertheless, the average cycle times of both controllers remain substantially below the sampling period of $T_s = 1$ s, demonstrating their suitability for real-time implementation.

Tables 8 and **9** compare the maximum computation times per control cycle of the RK model-based adaptive MIMO fuzzy PID controller and the adaptive MIMO PID controller, respectively. In all simulation cases, the adaptive MIMO PID controller exhibits a lower maximum computation time than the adaptive MIMO fuzzy PID controller. In the nominal case, the maximum cycle time decreases from 26.119 ms to 10.914 ms, corresponding to a 58.21% reduction. In the measurement-noise case, it decreases from 28.551 ms to 14.638 ms, yielding a 48.73% reduction. Under time-varying parametric uncertainty, the maximum cycle time is reduced from 32.619 ms to 17.586 ms, corresponding to a 46.09% reduction.

Table 8. Maximum computation time of the RK model-based adaptive MIMO fuzzy PID controller

Simulation case	**Maximum cycle time**
Nominal case	26.119 ms
Measurement-noise case	28.551 ms
Parametric-uncertainty case	32.619 ms

Table 9. Maximum computation time of the RK model-based adaptive MIMO PID controller

Simulation case	**Maximum cycle time**
Nominal case	10.914 ms
Measurement-noise case	14.638 ms
Parametric-uncertainty case	17.586 ms

For both controllers, the lowest maximum computation time is obtained in the nominal case, whereas the parametric-uncertainty case produces the highest value because of the additional online estimation of the time-varying inter-tank flow coefficient az_{13}. The higher maximum computation times of the adaptive MIMO fuzzy PID controller reflect the additional computational burden associated with the fuzzy inference operations. Nevertheless, even the largest recorded computation time, 32.619 ms, remains substantially below the sampling period of $T_s = 1$ s, confirming that both controllers satisfy the computational requirements for real-time implementation.

Overall, both the RK model-based adaptive MIMO fuzzy PID and adaptive MIMO PID controllers maintain bounded tracking responses, control inputs, adaptive parameters, and additive correction terms under nominal, noisy, and time-varying uncertain operating conditions. The performance-index comparison shows that the adaptive MIMO fuzzy PID controller provides better combined tracking and control-variation performance in all three cases, whereas the adaptive MIMO PID controller achieves lower average and maximum computation times. Nevertheless, the maximum computation times of both controllers remain substantially below the sampling period of $T_s = 1$ s. These results confirm the tracking effectiveness, robustness, regular adaptation behavior, and real-time computational feasibility of both control structures.

5. Conclusions

This study proposed an RK model-based predictive adaptive MIMO fuzzy PID controller for nonlinear and coupled systems. The RK4 prediction framework was used to propagate the known nonlinear plant dynamics and to provide the model sensitivities required for online adaptation. The input and output scaling parameters of the direct and cross-coupling fuzzy PID subcontrollers were adjusted using the LM method. An additive correction mechanism was also incorporated to provide supplementary control action during transient operation and plant–model mismatch. For comparative evaluation, the existing RK model-based adaptive MIMO PID controller reported in[2] was employed solely as a benchmark.

The proposed controller and the benchmark adaptive MIMO PID controller were evaluated on the same nonlinear three-tank liquid-level system under nominal operation, measurement noise, and time-varying parametric uncertainty. Both controllers maintained bounded tracking responses, control inputs, and adaptive parameters in all three scenarios, confirming the uniform ultimate boundedness established by the Lyapunov stability analysis. For both control structures, the additive correction mechanism contributed mainly during the initial transient, reference changes, and parameter variations, whereas the adapted main controller generated the dominant part of the control action during most of the simulation period.

The performance-index values obtained with the RK model-based adaptive MIMO fuzzy PID

controller were 0.2660, 0.2701, and 1.0739 for the nominal, measurement-noise, and parametric-uncertainty cases, respectively. The corresponding values obtained with the adaptive MIMO PID controller were 0.2833, 0.2877, and 1.1139. Thus, the adaptive MIMO fuzzy PID controller reduced the performance index by 6.11%, 6.12%, and 3.59%, respectively, compared with the adaptive MIMO PID controller. The limited increase observed under measurement noise indicated that both controllers maintained regular tracking and adaptation behavior under noisy feedback. In the parametric-uncertainty case, the online estimate closely followed the variations in the actual inter-tank flow coefficient az_{13}, thereby reducing the resulting plant–model mismatch. Although this case produced higher performance-index values for both controllers, all tracking responses, control inputs, and adaptive parameters remained bounded.

The average computation times per control cycle of the adaptive MIMO fuzzy PID controller were 0.1297 ms, 0.1485 ms, and 0.1774 ms for the nominal, measurement-noise, and parametric-uncertainty cases, respectively. The corresponding average cycle times of the adaptive MIMO PID controller were 0.0770 ms, 0.0998 ms, and 0.1269 ms, representing reductions of 40.63%, 32.79%, and 28.47%, respectively. Similarly, the maximum computation times were 26.119 ms, 28.551 ms, and 32.619 ms for the adaptive MIMO fuzzy PID controller and 10.914 ms, 14.638 ms, and 17.586 ms for the adaptive MIMO PID controller. Although the fuzzy PID structure required additional computation due to its fuzzy inference operations, the average and maximum cycle times of both controllers remained substantially below the sampling period of $T_s = 1$ s. These findings demonstrate that the adaptive MIMO fuzzy PID controller provides improved tracking and control-variation performance, whereas the adaptive MIMO PID controller offers lower computational cost, with both approaches remaining computationally feasible for the selected sampling period in the tested simulation environment.

In addition, a discrete-time Lyapunov-based stability analysis was established to evaluate closed-loop stability in the presence of aggregate disturbances, including plant–model mismatch, RK truncation errors, measurement noise, and estimation errors. The theoretical derivation guarantees the uniform ultimate boundedness (UUB) of the tracking error and ensures the boundedness of the adaptive scaling parameters via the projection algorithm, with asymptotic tracking convergence achieved in the disturbance-free nominal case. The Lyapunov stability indicators evaluated across the nominal, noisy, and uncertain simulation scenarios fully validate these theoretical findings. Future work will address experimental validation on physical laboratory testbeds and extension of the theoretical analysis to the complete interconnected closed loop.

Acknowledgments

This study has been conducted as part of the M.Sc. thesis undertaken by İsmail Koray GÜNDOĞAN in the Mechatronics Engineering Master of Science Program at Istanbul Technical University Graduate School, under the supervision of Assoc. Prof. Dr. Kemal UÇAK. The authors thank TÜBİTAK for its support under Grant Number 125E336. The stability analysis presented in this study was further developed and refined based on theoretical insights gained during the research activities carried out within the scope of the TÜBİTAK Project No. 125E336.

Funding

This study was supported by Scientific and Technological Research Council of Türkiye(TUBITAK) under the Grant Number 125E336. The author thanks to TUBITAK for their supports.

Conflict of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Author contributions

Conceptualization: Kemal Uçak
Formal analysis: All authors
Investigation: All authors
Methodology: All authors
Supervision: Kemal Uçak
Software: İsmail Koray Gündoğan
Validation: All authors
Visualization: İsmail Koray Gündoğan
Writing–original draft: All authors
Writing–review & editing: All authors

Availability of data

Not applicable. The findings of this study are based on numerical simulations. As no empirical datasets were used or generated, there are no data available for sharing.

AI tools statement

During the preparation of this work, the authors used Claude and ChatGPT only in order to improve readability and language. After using these tools/services, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

References

1. İplikçi S. Runge-Kutta model-based adaptive predictive control mechanism for non-linear processes. *Trans Inst Meas Control*. 2013;35(2):166-180. https://doi.org/10.1177/0142331212438910
2. Çetin M, İplikçi S. A novel auto-tuning PID control mechanism for nonlinear systems. *ISA Trans*. 2015;58:292-308. https://doi.org/10.1016/j.isatra.2015.05.017
3. Uçak K. A novel model predictive Runge-Kutta neural network controller for nonlinear MIMO systems. *Neural Process Lett*. 2020;51(2):1789-1833. https://doi.org/10.1007/s11063-019-10167-w
4. Slotine JJE, Li W. *Applied Nonlinear Control*. Englewood Cliffs, NJ: Prentice Hall; 1991. Accessed October 19, 2026. https://lewisgroup.uta.edu/ee5323/notes/Slotine%20and%20Li%20applied%20nonlinear%20control-%20bad%20quality.pdf
5. Ang KH, Chong G, Li Y. PID control system analysis, design, and technology. *IEEE Trans Control Syst Technol*. 2005;13(4):559-576. https://doi.org/10.1109/TCST.2005.847331
6. Sastry SS, Bodson M. *Adaptive Control: Stability, Convergence, and Robustness*. Englewood Cliffs, NJ: Prentice Hall; 1989. Accessed October 19, 2026. https://my.eng.utah.edu/~bodson/acscr/acscr.pdf
7. Zhao C, Guo L. Towards a theoretical foundation of PID control for uncertain nonlinear systems. *Automatica*. 2022;142:110360. https://doi.org/10.1016/j.automatica.2022.110360
8. Zhao C. Semiglobal stability of PID for uncertain nonaffine systems. *Automatica*. 2024;160:111429. https://doi.org/10.1016/j.automatica.2023.111429
9. Zhao C, Wang D, Xue W. Beyond linear limits: design of robust nonlinear PID control. *Automatica*. 2025;173:112075. https://doi.org/10.1016/j.automatica.2024.112075
10. Çelik D, Khosravi N, Khan MA, Waseem M, Ahmed H. Advancements in nonlinear PID controllers: a comprehensive review. *Comput Electr Eng*. 2026;129:110775. https://doi.org/10.1016/j.compeleceng.2025.110775
11. Qiao WZ, Mizumoto M. PID type fuzzy controller and parameters adaptive method. *Fuzzy Sets Syst*. 1996;78(1):23-35. https://doi.org/10.1016/0165-0114(95)00115-8
12. Mann GKI, Hu BG, Gosine RG. Analysis of direct action fuzzy PID controller structures. *IEEE Trans Syst Man Cybern B Cybern*. 1999;29(3):371-388. https://doi.org/10.1109/3477.764871
13. Carvajal J, Chen G, Ogmen H. Fuzzy PID controller: design, performance evaluation, and stability analysis. *Inf Sci*. 2000;123(3-4):249-270. https://doi.org/10.1016/S0020-0255(99)00127-9
14. Woo ZW, Chung HY, Lin JJ. A PID type fuzzy controller with self-tuning scaling factors. *Fuzzy Sets Syst*. 2000;115(2):321-326. https://doi.org/10.1016/S0165-0114(98)00159-6
15. Uçak K, Öke G. Adaptive fuzzy PID controller based on online LSSVR. In: *Proceedings of the International Symposium on Innovations in Intelligent Systems and Applications*. Piscataway, NJ: IEEE; 2012:1-7. https://doi.org/10.1109/INISTA.2012.6247020
16. Uçak K, Arslantürk BN. Adaptive MIMO fuzzy PID controller based on peak observer. *Int J Optim Control Theor Appl*. 2023;13(2):139-150. https://doi.org/10.11121/ijocta.2023.1247
17. Zhou Y, Wang X. Adaptive fuzzy command filtering control for nonlinear MIMO systems with full state constraints and unknown control direction. *Neurocomputing*. 2022;493:474-485. https://doi.org/10.1016/j.neucom.2021.12.091
18. Puviyarasi B, Murukesh C. Design and implementation of adaptive mixed fuzzy controller for MIMO nonlinear systems. *Math Comput Simul*. 2023;203:71-91. https://doi.org/10.1016/j.matcom.2022.06.019
19. Zhang X, Yang X, Huang C, Cao J, Liu H. Adaptive fuzzy finite-time PID backstepping control for chaotic systems with full states constraints and unmodeled dynamics. *Inf Sci*. 2024;661:120148. https://doi.org/10.1016/j.ins.2024.120148
20. Saleem O, Ahmad KR, Iqbal J. Fuzzy-augmented model reference adaptive PID control law design for robust voltage regulation in DC-DC buck converters. *Mathematics*. 2024;12(12):1893. https://doi.org/10.3390/math12121893
21. Rawlings JB, Mayne DQ, Diehl M. *Model Predictive Control: Theory, Computation, and Design*. 2nd ed. Madison, WI: Nob Hill Publishing; 2017. Accessed October 19, 2026. https://sites.engineering.ucsb.edu/~jbraw/mpc/MPC-book-2nd-edition-1st-printing.pdf
22. Chen Z, Chen X, Liu J, Cen L, Gui W. Learning model predictive control of nonlinear systems with time-varying parameters using Koopman operator. *Appl Math Comput*. 2024;470:128577. https://doi.org/10.1016/j.amc.2024.128577

23. Asvadi-Kermani O, Sorouri H, Oshnoei A. Adaptive regulation of a non-inverting buck-boost converter with Levenberg-Marquardt-based sliding-mode predictive control scheme. *Sci Rep.* 2025;15:28783. https://doi.org/10.1038/s41598-025-14691-9
24. Uçak K. A novel adaptive Runge-Kutta controller for nonlinear dynamical systems. *Soft Comput.* 2021;25(16):10915-10933. https://doi.org/10.1007/s00500-021-05792-4
25. Haring M, Grøtli EI, Riemer-Sørensen S, Seel K, Hanssen KG. A Levenberg-Marquardt algorithm for sparse identification of dynamical systems. *IEEE Trans Neural Netw Learn Syst.* 2023;34(11):9323-9336. https://doi.org/10.1109/TNNLS.2022.3157963
26. Waseem W, Ali S, Ali A, Thaljaoui A, Meetei MZ. Dynamics of infectious disease mathematical model through unsupervised stochastic neural network paradigm. *Comput Biol Chem.* 2025;115:108291. https://doi.org/10.1016/j.compbiolchem.2024.108291
27. Khan Z, Alfwzan WF, Ali A, *et al.* Intelligent computing for electromagnetohydrodynamic bioconvection flow of micropolar nanofluid with thermal radiation and stratification: Levenberg-Marquardt backpropagation algorithm. *AIP Adv.* 2024;14(3):035224. https://doi.org/10.1063/5.0187124
28. Hagan MT, Menhaj MB. Training feedforward networks with the Marquardt algorithm. *IEEE Trans Neural Netw.* 1994;5(6):989-993. https://doi.org/10.1109/72.329697
29. Amira GmbH. *DTS200-Laboratory Setup Three-Tank System.* Duisburg, Germany: Amira GmbH; 2000. Accessed October 19, 2026. https://www.yumpu.com/en/document/view/55596767/dts200-laboratory-setup-three-tank-system
30. Uçak K. A Runge-Kutta neural network-based control method for nonlinear MIMO systems. *Soft Comput.* 2019;23(17):7769-7803. https://doi.org/10.1007/s00500-018-3405-5
31. Zare H, Hajarian M. An efficient Gauss-Newton algorithm for solving regularized total least squares problems. *Numer Algorithms.* 2022;89:1049-1073. https://doi.org/10.1007/s11075-021-01145-2
32. İplikçi S. A comparative study on a novel model-based PID tuning and control mechanism for nonlinear systems. *Int J Robust Nonlinear Control.* 2010;20(13):1483-1501. https://doi.org/10.1002/rnc.1524
33. Zadeh LA. Fuzzy sets. *Inf Control.* 1965;8(3):338-353. https://doi.org/10.1016/S0019-9958(65)90241-X
34. Mamdani EH, Assilian S. An experiment in linguistic synthesis with a fuzzy logic controller. *Int J Man Mach Stud.* 1975;7(1):1-13. https://doi.org/10.1016/S0020-7373(75)80002-2
35. Camacho EF, Bordons C. *Model Predictive Control.* 2nd ed. London, UK: Springer; 2004. Accessed October 19, 2026. https://d-nb.info/971690634/04
36. Dogan KM, Gruenwald BC, Yucelen T, Muse JA. Relaxing the stability limit of adaptive control systems in the presence of unmodelled dynamics. *Int J Control.* 2017;90(12):2565-2576. https://doi.org/10.1080/00207179.2017.1330557
37. Uçak K. A novel data-driven model predictive controller for nonlinear dynamical systems based on Runge-Kutta sparse identification. *Neural Comput Appl.* 2026;38:591. https://doi.org/10.1007/s00521-026-12208-7
38. Uçak K, Öke Günel G. An adaptive support vector regressor controller for nonlinear systems. *Soft Comput.* 2016;20(7):2531-2556. https://doi.org/10.1007/s00500-015-1654-0

İsmail Koray Gündoğan *is a graduate student in Mechatronics Engineering at Istanbul Technical University. His research interests are adaptive control, adaptive fuzzy control architectures, and robust control.*
https://orcid.org/0009-0003-1530-7611

Kemal Uçak *received his M.Sc. and Ph.D. degrees in Control and Automation Engineering from Istanbul Technical University in 2012 and 2016, respectively. He is currently an Associate Professor in the Department of Control and Automation Engineering at Istanbul Technical University. He conducted postdoctoral research at the Massachusetts Institute of Technology (MIT) in 2020. His research interests pertain to adaptive control, artificial intelligence, data-driven control, machine learning, robust adaptive control, and support vector machines. Dr. Uçak is a member of IEEE and IFAC.*
https://orcid.org/0000-0001-7005-7940

An International Journal of Optimization and Control: Theories & Applications (https://accscience.com/journal/ijocta)



This work is licensed under a Creative Commons Attribution 4.0 International License. The authors retain ownership of the copyright for their article, but they allow anyone to download, reuse, reprint, modify, distribute, and/or copy articles in IJOCTA, so long as the original authors and source are credited. To see the complete license contents, please visit http://creativecommons.org/licenses/by/4.0/.