Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

A Stability-Aware Gap Metric Oriented Framework for Smooth and Stable Blending in Multiple Model Control

Mahdi Ahmadi

mahdiahmadi.ee@gmail.com

Esfarayen University of Technology

Ali Zribi

National Engineering School of Gabes, Universty of Gabes

Research Article

Keywords: Multiple model control, gap metric, stability-aware weighting, nonlinear systems, auxiliary local models

Posted Date: November 17th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11476118/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

A Stability-Aware Gap Metric Oriented Framework for Smooth and Stable Blending in Multiple Model Control

Mahdi Ahmadi[1*], Ali Zribi[2]

[1*]Faculty of Electrical and Computer Engineering, Esfarayen University of Technology, Esfarayen, Iran, E-mail: mahdiahmadi.ee@gmail.com

[2]Research Laboratory: Numerical Control of Industrial Processes, National Engineering School of Gabes, Universty of Gabes,Tunisia, E-mail: ali_zribi@yahoo.fr

Abstract:

This paper addresses a critical limitation of gap metric-based blending approaches for multiple model control of nonlinear systems. While existing methods compute weighting functions solely based on the open-loop distance between local models, they neglect explicit closed-loop stability criteria during the blending stage. Consequently, even when the local controllers are individually stabilizing, the resulting global controller may compromise overall closed-loop stability. To overcome this shortcoming, a novel stability-aware weighting framework is proposed that explicitly integrates stability conditions into the gap metric-based weights calculation. The proposed method ensures that only stabilizing local controllers contribute to the global multiple model control signal. Furthermore, to prevent hard switching and oscillations that may arise from directly enforcing stability constraints, an auxiliary local model strategy is introduced that enables smooth transitions across operating regions. The effectiveness of the framework is demonstrated through comprehensive simulations on a highly nonlinear continuous stirred tank reactor (CSTR) benchmark over a wide operating range. Results confirm that the proposed approach not only guarantees closed-loop stability but also achieves superior tracking performance compared to conventional gap metric-based methods, while eliminating undesirable oscillations during region transitions through the use of auxiliary local models.

Keywords: Multiple model control, gap metric, stability-aware weighting, nonlinear systems, auxiliary local models.

1. Introduction

Multiple model control is a well-established and powerful methodology for controlling complex nonlinear systems [1, 2]. Its fundamental principle is to decompose the complex global behavior of a system by employing a finite set of simpler local models, each accurately capturing the

system dynamics within a specific region of the operating space [3]. By transforming a complex nonlinear control problem into a series of more tractable linear sub-problems, multiple model control leverages the analytical power of linear control theory to achieve the adaptability and robustness needed to handle nonlinear dynamic behavior, uncertainties, and changing operating points.

The architecture of a standard multiple model control framework typically comprises three interconnected stages [3]. First, a bank of nominal local models is designed to span the system's operational range. Second, a dedicated local linear controller is synthesized for each nominal local model to guarantee stability and performance in its respective region. Third, a supervisory unit is implemented to determine the overall control action in real-time by strategically switching among or smoothly blending the outputs of the local controllers. The efficacy of the multiple model architecture depends on the correct design of the first two stages, with the supervisory unit playing a pivotal role in providing proper transitions between operating regions while preserving stability and meeting performance specifications.

The supervisory layer typically operates using one of two mechanisms: switching [4, 5], which activates the single most appropriate local controller at each time step, or blending [6-8], which synthesizes a global control signal from a weighted combination of local controllers. This work focuses on blending techniques due to their inherent capability for smooth interpolation, thereby avoiding the chattering and abrupt transitions associated with switching [3]. Among the various methods for calculating the blending weights, including Bayesian inference [9], Gaussian and trapezoidal functions [10, 11], optimization [12], and adaptation algorithms [7], the approach based on the gap metric [13-15] offers distinct advantages. These include tuning simplicity via a single parameter and, critically, its foundation in a closed-loop measure of distance, a property that distinguishes it from conventional alternatives. Furthermore, gap metric-based weights can often be calculated off-line, significantly reducing the online computational burden compared to methods requiring continuous adaptation [7] or optimization [12].

The development of gap metric-based weighting for multiple model control has been driven by a key trade-off between theoretical rigor and computational feasibility. The foundational work by Arslan et al. [16] established the principle, employing a closed-loop gap metric that incorporates local controller dynamics to compute weights. While theoretically rigorous, this method required heavy online computation as weights depended on controller parameters, limiting real-time application. To overcome this, Du and Johansen [13] proposed a significant simplification by calculating weights directly using the open-loop gap metric between plant

models. This yielded static functions like $\frac{1}{\delta}$ and $1-\delta$, where a smaller distance δ grants a model a larger weight. This development enabled off-line precomputation and drastically reduced online computational load, a practical advantage later confirmed in an experimental study by Prasad and Rao [17]. Subsequent research has focused on refining these efficient, open-loop foundations. Du et al. [14] introduced a new weighting function, a variant of the $1-\delta$ form, aimed at improving closed-loop performance metrics such as transient response. Most recently, Elkhalil and Zribi [15] developed a generalized structure by fusing the gap metric with trapezoidal functions, creating a more flexible framework for smooth interpolation.

However, a critical limitation persists across these methods: none explicitly incorporates closed-loop stability criteria into the weights calculation. The blending logic is based solely on the "distance" between models, without guaranteeing that the resulting blended controller preserves stability. Consequently, even when the decomposition and local controller design stages are performed with stability criteria, as demonstrated in works like [11, 18], the application of standard gap metric-based weights can still compromise the stability of the overall closed-loop system during the blending stage. This limitation constitutes the primary motivation for the present study.

In response to this shortcoming, this article introduces a novel blending framework for multiple model control that guarantees closed-loop stability. The fundamental contributions are threefold:

- We propose a new weighting function that explicitly integrates stability criteria with the gap metric, ensuring the global controller maintains stability during transitions across operating regions, and directly addressing the main limitation of prior methods.
- To prevent the chattering and oscillations that a direct stability constraint could induce, we introduce a strategy employing auxiliary nominal local models. This ensures smooth interpolation and soft transitions by maintaining multiple active controllers during region crossing.
- The framework's effectiveness and robustness are demonstrated through comprehensive simulation of a complex nonlinear system with a wide operating range, confirming enhanced performance in reference tracking while maintaining guaranteed stability.

The remainder of this article is organized as follows. Section 2 reviews the necessary preliminaries on the gap metric and stability. The proposed stability-aware blending framework

and the auxiliary model strategy are presented in Section 3. Simulation results and discussion are given in Section 4, followed by conclusions in Section 5.

2. Theoretical Background

The gap metric is a fundamental tool in systems and control theory for measuring the distance between linear dynamical systems. Its significance stems from its intrinsic connection to closed-loop stability, making it a critical measure for robust control design, model approximation, and; most pertinently, for the synthesis and analysis of multiple model control systems.

2.1. Gap Metric: A Measure of System Similarity

The gap metric entered the control literature through the work of Zames and El-Sakkary in 1985 [19], who recognized its profound suitability for analyzing feedback systems and model uncertainty. While traditional norms measured open-loop output differences, the gap metric was shown to quantify the difference in closed-loop behavior, making it a natural metric for questions of robust stabilization. This was later refined by the foundational work of Georgiou in 1988 [20], who provided a computable H_∞ formula for the metric, cementing its practical utility in control engineering.

The gap metric, denoted $\delta(G_1, G_2)$, provides a rigorous measure of the "distance" or similarity between two linear time-invariant (LTI) systems, G_1 and G_2. Unlike classic norms (e.g., H_∞ or H_2), which measure open-loop output differences, the gap metric quantifies the difference in closed-loop behavior. It is defined within the framework of normalized coprime factorizations. Let a system G have a normalized left coprime factorization $G = M^{-1}N$, where $N, M \in \mathcal{RH}_\infty$ and satisfy $MM^* + NN^* = 1$. The graph of G, a fundamental object representing all its input-output pairs (u, y), is given by:

$$\mathcal{G}_G = \left\{ \begin{bmatrix} N \\ M \end{bmatrix} u : u \in \mathcal{H}_\infty \right\} \tag{1}$$

The gap between two systems G_1 and G_2 is then defined as the directed gap $\vec{\delta}(G_1, G_2)$:

$$\vec{\delta}(G_1, G_2) = \inf_{Q \in H_\infty} \left\| \begin{bmatrix} N_1 \\ M_1 \end{bmatrix} - \begin{bmatrix} N_2 \\ M_2 \end{bmatrix} Q \right\|_\infty \tag{2}$$

with the (undirected) gap metric given by [21]:

$$\delta(G_1, G_2) = \max\left\{ \vec{\delta}(G_1, G_2), \vec{\delta}(G_2, G_1) \right\} \tag{3}$$

The gap metric satisfies $0 \leq \delta(G_1, G_2) \leq 1$. A small gap ($\delta(G_1, G_2) \approx 0$) indicates the systems have nearly identical closed-loop behaviors and can be stabilized by the same controller. A large gap ($\delta(G_1, G_2) \approx 1$) signifies fundamental dissimilarity in their closed-loop dynamics.

2.2. Gap Metric and Robust Stability Connection

The primary significance of the gap metric in control theory lies in its direct connection to robust stability. The core relationship between the gap metric and stability is captured by the following theorem.

Theorem 1 [3]: Let K be a controller that stabilizes a nominal plant G. Then K also stabilizes every plant G_p in the set:

$$\mathcal{G}_\epsilon = \{G_p : \delta(G, G_p) \leq \epsilon\} \tag{4}$$

if and only if the following condition holds:

$$\epsilon \leq b_{G,K} = \left\| \begin{bmatrix} I \\ K \end{bmatrix} (I - GK)^{-1} [I \quad G] \right\|_\infty^{-1}. \tag{5}$$

Here, $b_{G,K}$ is the stability margin (or robustness margin) for the pair (G, K).

This theorem establishes a direct and quantitative link as follows:

- $b_{G,K}$ is the largest gap metric ball around the nominal plant G within which stability is guaranteed under the fixed controller K.
- Stability is guaranteed if $\delta(G, G_p) \leq b_{G,K}(G, K)$. If the distance to a new plant G_p is less than the stability margin, the controller K will also stabilize G_p.
- Instability risk occurs if $\delta(G, G_p) > b_{G,K}(G, K)$. If a plant model is too "far" in the gap metric sense, the original controller may fail to stabilize it.

This property establishes the gap metric as the natural and rigorous tool for quantifying the admissible "size" of model uncertainty in a feedback loop. Owing to these vital features, its framework is now directly applied in multiple model control to enable the systematic design of stable model banks and / or switching schemes.

3. Traditional Gap Metric-Based Weighting Functions

A nonlinear system can be described by the following general equations:

$$\begin{aligned} \dot{x} &= f(x, u), \\ y &= g(x, u), \end{aligned} \tag{6}$$

with the state vector $x \in \mathbb{R}^n$, the control input $u \in \mathbb{R}^m$, and the measured output $y \in \mathbb{R}^m$. The vector-valued functions $f(\cdot): \mathbb{R}^{n \times m} \to \mathbb{R}^n$ and $g(\cdot): \mathbb{R}^{n \times m} \to \mathbb{R}^m$ are known, nonlinear, and differentiable.

The dynamic behavior of the nonlinear system (6) is parameterized by introducing a slowly varying scheduling vector, denoted as $\theta(t) \in \mathbb{R}^p$. This vector serves as a low-dimensional proxy for the system's operational regime, capturing the essential nonlinearities by depending

on key physical variables. While θ can, in principle, be formulated as a function of states, inputs, outputs, or external disturbances, the most frequent and practical choice is to select one or more measured input and output variables, as these are directly accessible for real-time scheduling. The set of all admissible values for θ defines the system's operating space or scheduling space, denoted by:

$\Phi \subseteq \mathbb{R}^p$(i.e., $\Phi = \{\theta | \theta_{min} \leq \theta \leq \theta_{max}\}$) (7)

To construct a multiple model control, the operating space Φ is first divided. This is achieved by partitioning Φ into N_m convex, non-overlapping regions Φ_i using a standard decomposition technique such as grid-based dividing [22], or dichotomy [23]. Each region Φ_i is represented by a specific operating point, which also defines the equilibrium point for a nominal local linear model. The i-th operating point, $(x_{e_i}, u_{e_i}, y_{e_i})$, must correspond to a valid steady-state solution of the original nonlinear system (6). Consequently, it satisfies the equilibrium conditions as follows:

$f(x_{e_i}, u_{e_i}) = 0$ and $y_{e_i} = g(x_{e_i}, u_{e_i})$. (8)

A local linear time invariant model, G_i, is subsequently derived for each region by applying first order Taylor series expansion (linearization) of the nonlinear function (6) around the associated equilibrium point (x_{e_i}, u_{e_i}). This yields a state-space model expressed in terms of deviation variables as follows:

$$\begin{aligned} \delta\dot{x}_i &= A_i \delta x_i + B_i \delta u_i \\ \delta y_i &= C_i \delta x_i + D_i \delta u_i \end{aligned}, \quad (9)$$

where the state-space matrices are the Jacobians of the nonlinear functions evaluated at the equilibrium point as follows:

$A_i = \frac{\partial f}{\partial x}|_{(x_{ei}, u_{ei})}$, $B_i = \frac{\partial f}{\partial u}|_{(x_{ei}, u_{ei})}$, $C_i = \frac{\partial g}{\partial x}|_{(x_{ei}, u_{ei})}$, $D_i = \frac{\partial g}{\partial u}|_{(x_{ei}, u_{ei})}$. (10)

Also, the deviation variables, representing perturbations from the nominal operating condition, are defined as follows:

$\delta x_i = x - x_{ei}$, $\delta u_i = u - u_{ei}$, and $\delta y_i = y - y_{ei}$. (11)

For each of these N_m linear models, a local controller K_i is synthesized using an appropriate linear design methodology, such as PID, LQR, LQG, or H_∞ control, to achieve desired stability and performance specifications within its respective region. Finally, the global control law is then generated as a time-varying convex blending of the individual local control signals as follows:

$u(t) = \sum_{i=1}^{N_m} \omega_i(\theta(t)) u_i(t)$, (12)

where $u_i(t)$ is the control input generated by the i-th local controller. The scheduling dependent weighting functions $\omega_i\big(\theta(t)\big)$ play a critical role in ensuring a smooth transition between local controllers and must satisfy the convexity conditions as follows:

$$\begin{matrix}\sum_{i=1}^{N_m}\omega_i\big(\theta(t)\big)=1\\ 0\le\omega_i\big(\theta(t)\big)\le 1,\forall\, i=1,\cdots,N_m\end{matrix}. \tag{13}$$

The design of the weighting functions $\omega_i\big(\theta(t)\big)$ is paramount, as it directly influences system performance, transient response, and steady-state accuracy. Among various alternatives (Gaussian functions, Bayesian methods, etc.), the gap metric-based blending method is selected for this study. Its principal advantage lies in its simplicity and minimal tuning requirements unlike other approaches whose parameter count scales with N_m.

For the available nonlinear model, described in general by Equation (6), its instantaneous representation at time t is denoted nG_t. This model is characterized by a concurrently evolving scheduling vector $\theta(t)$, which corresponds to a specific steady-state operating condition $\big(x_{e_t},u_{e_t},y_{e_t}\big)$. A local linear approximation of the system is obtained through Jacobian linearization as shown in (9), yielding the linear time-invariant model G_t. Then the affinity between this instantaneous linear model G_t and each of the N_m pre-defined nominal local models G_i is computed by gap metric as follows:

$$\mu_i\big(\theta(t)\big)=\delta(G_i,G_t),i=1,2,\cdots,N_m. \tag{14}$$

For the i-th model at time t, the weighting function, which governs the contribution of each local controller in the global control law in (12); is constructed in the following forms [13-15]:

$$\omega_i\big(\theta(t)\big)=\frac{\big(1-\mu_i(\theta(t))\big)^{\beta}}{\sum_{i=1}^{N_m}\big(1-\mu_i(\theta(t))\big)^{\beta}},i=1,2,\cdots,N_m \tag{15}$$

$$\omega_i\big(\theta(t)\big)=\frac{\left(\frac{1}{\mu_i(\theta(t))}\right)^{\beta}}{\sum_{i=1}^{N_m}\left(\frac{1}{\mu_i(\theta(t))}\right)^{\beta}},i=1,2,\cdots,N_m \tag{16}$$

with $\beta\ge 1$, a designer-specified tuning parameter that controls the blending sharpness. The functional form in (15) intrinsically satisfies the convexity constraints specified in (13). The functional behavior of this gap metric-based weighting scheme is illustrated in Fig. 1.

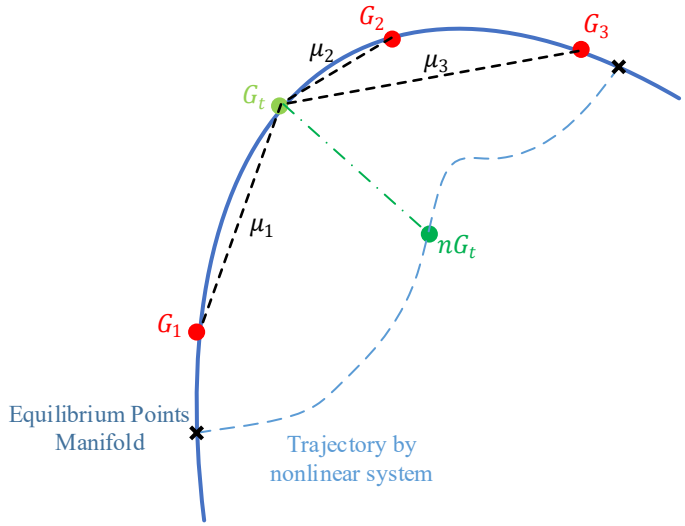


Fig.1. Gap metric-based blending weights scheme.

4. Stability-Aware Weighting Functions

4.1. Stable Transition Between Regions

From Fig. 1 and Equations (15) and (16), we can see that the contribution of each local controller within the global controller is determined by the distance between the time-dependent linear model G_t and the nominal local models G_1, G_2, and G_3, which quantified by the parameters μ_1, μ_2, and μ_3, respectively. While this observation holds true in the absence of stability considerations, it needs to be reevaluated when stability is taken into account. For instance, if the distance between G_t and G_3 is less than one, we would expect local controller K_3 to have a significant impact on the global controller. However, if the stability margin of G_3 is smaller than this distance (i.e., less than μ_3), then according to Theorem 1, G_3 cannot stabilize G_t. Therefore, including K_3 in the global controller would not only be inappropriate but could potentially destabilize the overall nonlinear system. This limitation is addressed by Proposition 1, which proposes a revised formulation to tackle this issue.

Proposition 1: Consider a multiple model control framework comprising N_m local linear models G_i and corresponding stabilizing controllers K_i, $i = 1,2,\cdots,N_m$. Let $\delta_i = \delta(G_i, G_t)$ denote the gap metric between the i-th nominal model and the instantaneous linearized model G_t at time t, and let $\varepsilon_i > 0$ be the stability margin associated with the pair (G_i, K_i). Define a revised function as follows:

$$\bar{\mu}_i\big(\theta(t)\big) = \frac{1}{2}\big((1+\delta_i) + (1-\delta_i)\,\mathrm{sgn}(\delta_i - \varepsilon_i)\big), i = 1,2,\cdots,N_m. \tag{17}$$

where the sign function follows the convention $\mathrm{sgn}(x) = 1$ for $x > 0$ and $\mathrm{sgn}(x) = -1$ for

$x \leq 0$. Let the blending weights be given by:

$$\omega_i\big(\theta(t)\big) = \frac{\big(1-\bar{\mu}_i(\theta(t))\big)^{\beta}}{\sum_{i=1}^{Nm}\big(1-\bar{\mu}_i(\theta(t))\big)^{\beta}}, i = 1,2,\cdots,N_m \tag{18}$$

$$\omega_i\big(\theta(t)\big) = \frac{\left(\frac{1}{\bar{\mu}_i(\theta(t))}\right)^{\beta}}{\sum_{i=1}^{Nm}\left(\frac{1}{\bar{\mu}_i(\theta(t))}\right)^{\beta}}, i = 1,2,\cdots,N_m \tag{19}$$

with $\beta \geq 1$, a tuning parameter. Then: (I)- the weights satisfy the convexity conditions given in (13); (II)- the closed-loop system under the global control law (12) remains stable at every time instant t.

Proof: (I)- Convexity follows directly from the normalization in (18) and (19). (II)- Stability is ensured because (17) implements a hard switch at $\delta_i = \varepsilon_i$:

- For $\delta_i(t) \leq \varepsilon_i$, we have $\bar{\mu}_i\big(\theta(t)\big) = \delta_i(t)$;
- For $\delta_i(t) > \varepsilon_i$, we obtain $\bar{\mu}_i\big(\theta(t)\big) = 1$.

Consequently, only models with $\delta_i(t) \leq \varepsilon_i$ (i.e., within the stability margin of their respective controllers) receive non-zero weight, while all others are excluded. The blended controller thus interpolates exclusively among locally stabilizing controllers, each operating within its certified stability region, thereby preserving closed-loop stability. ■

Remark 1: The parameter β controls the selectivity of the weighting: larger values of β sharpen the transition, giving dominant weight to the model with the smallest gap metric. The function (17) guarantees that weights are automatically assigned only to models whose gap metric does not exceed the stability margin ε_i.

4.2. Smooth Transition Between Regions

Consider a scenario in which the operating space is decomposed into non-overlapping sub-regions, and the weighting functions are defined based on stability criteria according to Equation (18) or (19). The weighting functions (18) or (19) can exhibit hard switching behavior during boundary transitions, as shown for the Φ_1 to Φ_2 transition in Fig. 2a. In such instances, the fundamental advantage for employing gap metric theory to compute the weighting functions is lost, potentially leading to oscillations and excessive chattering in the closed-loop response. To address this issue, the introduction of auxiliary local models is proposed, as shown in Fig. 2b. These auxiliary local models are situated at the boundaries between adjacent sub-regions, with corresponding local controllers designed for each of them. In Fig. 2b, the primary local models are denoted as G_1, G_2, and G_3, while the auxiliary local models are labeled G_{A_1} and G_{A_2}.

Consequently, when the system operates near a sub-region boundary, the auxiliary local controller facilitates a smooth transition between sub-regions by blending the control actions of at least two local controllers, thereby mitigating oscillations and suppressing chattering.

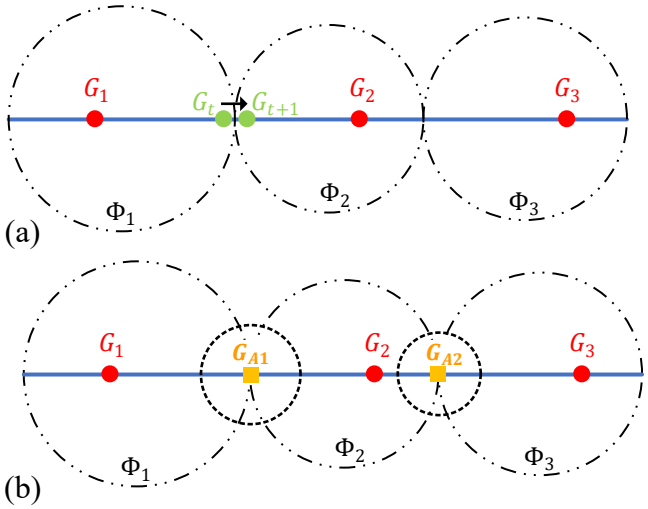


Fig.2. Transition between regions scheme: (a) Hard switching, using primary nominal local models (●); (b) Soft switching using primary (●) and auxiliary (■) nominal local models.

Remark 2: The stability margins of the auxiliary local controllers must be carefully bounded. A large stability margin would cause these controllers to dominate the weighting functions (18) or (19) and compromise the action of the primary controllers. While various methodologies can be employed for designing these auxiliary controllers, the approach presented in reference [3] can be utilized, with the stability margin explicitly considered as follows:

$$b_{G,K}\left(G_{A_j}, K_{A_j}\right) = \min\left\{b_{opt}\left(G_{A_j}\right), \max\left\{\delta\left(G_{A_j}, G_i\right), \delta\left(G_{A_j}, G_{i+1}\right)\right\}, 0.2\right\}, i = 1,2,\cdots,N_m,$$
$$j = 1,2,\cdots,N_m - 1, \quad (20)$$

where, K_{A_j} is the local controller for the j-th auxiliary local model G_{A_j}, and $b_{opt}\left(G_{A_j}\right)$ is the maximum stability margin, defined as follows:

$$b_{\text{opt}}\left(G_{A_j}\right) = \left\{\inf_{K \text{ stabilizing}} \left\|\begin{bmatrix} I \\ K \end{bmatrix}\left(I + G_{A_j}K\right)^{-1}\begin{bmatrix} I & G_{A_j} \end{bmatrix}\right\|_\infty\right\}^{-1} = \sqrt{1 - \left\|\begin{bmatrix} N_{A_j} & M_{A_j} \end{bmatrix}\right\|_{\text{H}}^2} < 1 \quad (21)$$

with N_{A_j} and M_{A_j} being the normalized left coprime factorization of G_{A_j} (i.e. $G_{A_j}(s) = M_{A_j}^{-1}N_{A_j}$), and $\|\cdot\|_{\text{H}}$ Hankel norm. This formulation results in an auxiliary local controller with a stability margin as defined by Equation (20). Therefore, the auxiliary local controller is only active within a small, and bounded region at the boundary between two consecutive sub-regions and is utilized solely during short transition periods. This ensures that the performance of the primary controllers is maintained without any compromise.

5. Simulation Results and Discussion

The dynamic behavior of a nonlinear continuous stirred tank reactor (CSTR) system is described by the following first order differential equation [18, 24]:

$$\begin{cases} \frac{dh(t)}{dt} = Q_1(t) + Q_2(t) - \alpha\sqrt{h(t)} \\ \frac{dC_b(t)}{dt} = \left(C_{b1} - C_b(t)\right)\frac{Q_1(t)}{h(t)} + \left(C_{b2} - C_b(t)\right)\frac{Q_2(t)}{h(t)} - \frac{k_1 C_b(t)}{\left(1+k_2 C_b(t)\right)^2} \end{cases}. \quad (22)$$

Here, $h(t)$ and $C_b(t)$ represent the liquid level and product concentration, respectively, which constitute the states of the system. The process is supplied by two feed streams: the first has a concentration $C_{b1} = 24.9\ kmol/m^3$ with a manipulable flow rate $Q_1(t)$, while the second has a concentration $C_{b2} = 5\ kmol/m^3$ and a fixed flow rate $Q_2(t) = 0.7\ l/min$. The reaction kinetics are characterized by rate constants $k_1 = k_2 = 1$, and the parameter α is taken as 0.2. The control objective is to regulate the flow rate $Q_1(t)$ such that the product concentration $C_b(t)$ achieves and maintains a desired set-point value.

To construct the multiple model controller, the output concentration, C_b, is selected as the scheduling variable, as it effectively characterizes the nonlinear dynamic behavior and defines the operating condition of the CSTR process. The scheduling variable varies over the range $C_b \in [0.8, 16]$. The static input-output map, obtained by applying the equilibrium conditions given in (8), is illustrated in Fig. 3a.

The dichotomy gridding algorithm with a threshold of 0.1 [23] was applied to partition the operating range into 30 operating points, as shown in Fig. 3a. A bank of local linear models was then generated by linearizing the nonlinear model (22) at each of these operating points using (9) - (11).

The pairwise distance between local models was quantified using the gap metric, computed via (3) and the "gapmetric" command in MATLAB. The resulting gap metric matrix for the set of 30 linear models is displayed as a color map in Fig. 3b. Each cell (i, j) corresponds to the gap metric $\delta(G_i, G_j)$, with the matrix symmetry reflected in the symmetric coloring. In this representation, darker colors indicate larger gaps (less similarity between local models), while lighter colors denote smaller gaps (greater similarity). The diagonal entries appear light since $\delta(G_i, G_i) = 0$. Off-diagonal cells are generally darker, reflecting increasing distance between distinct local models. The maximum gap of 0.95 occurs between local models 1 and 30 (i.e., $\delta(G_1, G_{30}) = 0.95$).

This significant variation across the operating range confirms that a single linear controller cannot ensure stability and performance for the nonlinear CSTR process (22). Indeed, as

demonstrated previously by Ahmadi and Haeri [18], at least three local controllers are required to stabilize the system using a multiple-model approach. Consequently, the CSTR process represents a highly nonlinear dynamic system and serves as a suitable case study for evaluating the proposed blending approach in this work.

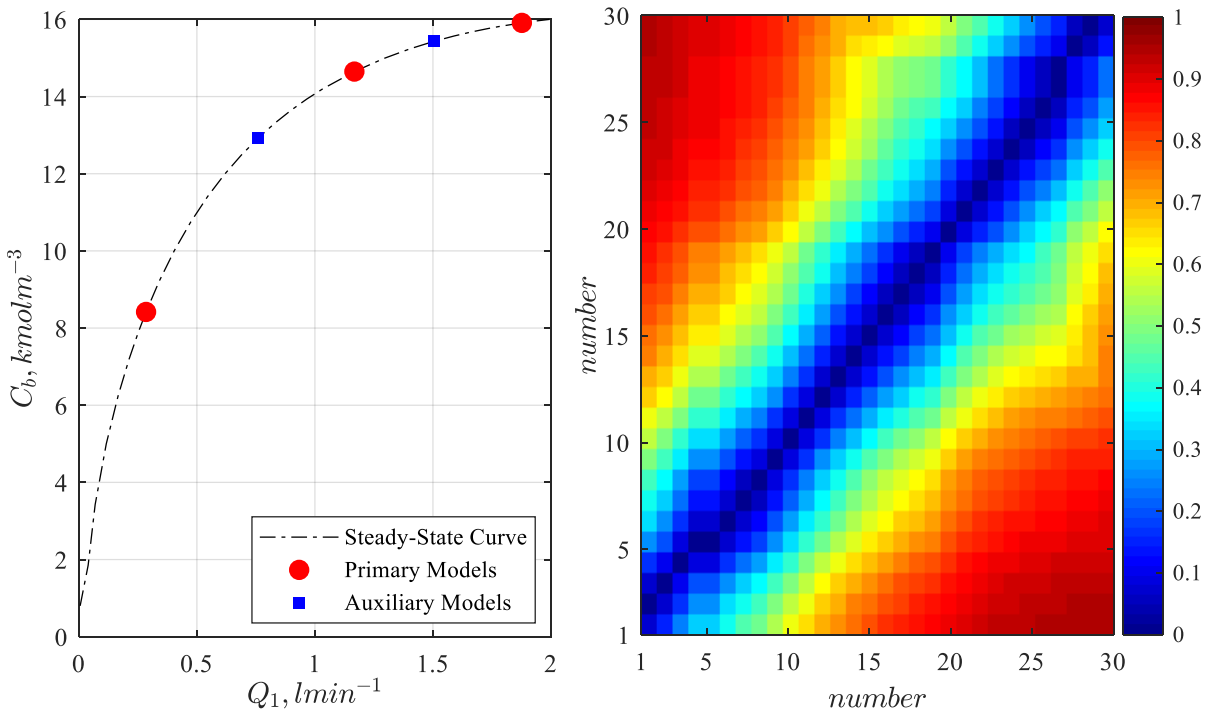


Fig. 3: (a) Steady-state curve with primary and auxiliary local models for the nonlinear system (22) (Left); (b) Visualization of the pairwise gap metric between 30 local models (Right).

As previously mentioned, following the method proposed by Ahmadi and Haeri [18], the operational range was divided into three sub-regions. Each sub-region is defined by a specific set of local linear models, a chosen nominal model, and a corresponding locally stabilizing controller. The details are as follows:

Sub-region 1 encompasses local models 1 through 15, corresponding to a scheduling variable range of $C_b \in [0.8, 12.92]$. The nominal local model is defined at $C_b = 8.42$ (local model 8) and is described by the transfer function $G_1(s) = \frac{0.68s+0.01}{s^2+0.05s+0.0006}$. A corresponding PI controller, $K_1(s) = 1.62 + \frac{0.20}{s}$, was synthesized using an integrated stability-performance method [18], achieving a generalized stability margin of $b_{G,K}(G_1, K_1) = 0.50$.

Sub-region 2 is defined by local models 15 to 23, with $C_b \in [12.92, 15.42]$. Its nominal operating point is at $C_b = 14.64$ (local model 20). The nominal model is $G_2(s) =$

$\frac{0.12s+0.001}{s^2+0.03s+0.0002}$, stabilized by the controller $K_2(s) = 3.18 + \frac{0.20}{s}$. This pairing yields a stability margin of $b_{G,K}(G_2, K_2) = 0.30$.

Sub-region 3 covers local models 23 to 30, operates within $C_b \in [15.42, 16]$. The nominal local model at $C_b = 15.9$ (local model 28) is given by $G_3(s) = \frac{0.05s+8.5e-5}{s^2+0.02s+9.6e-5}$. Its associated controller, $K_3(s) = 2.00 + \frac{0.14}{s}$, provides a stability margin of $b_{G,K}(G_3, K_3) = 0.41$.

As established in previous analyses, traditional gap metric-based weighting functions pose a significant limitation: they do not explicitly consider closed-loop stability. To investigate this issue, we computed the gap between the first nominal model, G_1, and all other local models, as shown in Fig. 4a. The stability margin achieved by the corresponding local controller, K_1, is also displayed (green dash-dotted line). This analysis was then repeated for the remaining nominal models, G_2 and G_3, with their results presented in Figs. 4b and 4c, respectively.

The calculated gap metric-based weighting functions, determined using (15) with $\beta = 1$, are illustrated in Figs. 4d-4f. Local models that do not meet the stability condition in (5) are marked by red vertical lines in Figs. 4a-4c. Importantly, Figs. 4d-4f show that the weighting functions for these local models are less than one. This assigns them non-zero blending weights in the global multiple-model controller, thereby allowing them into the feedback loop. Consequently, their instability can propagate and compromise the stability of the overall closed-loop system. This highlights the importance of incorporating stability conditions directly into the weighting function logic for nominal local controllers. This is crucial to prevent the destabilizing impact of inappropriate local models in the blended global control law.

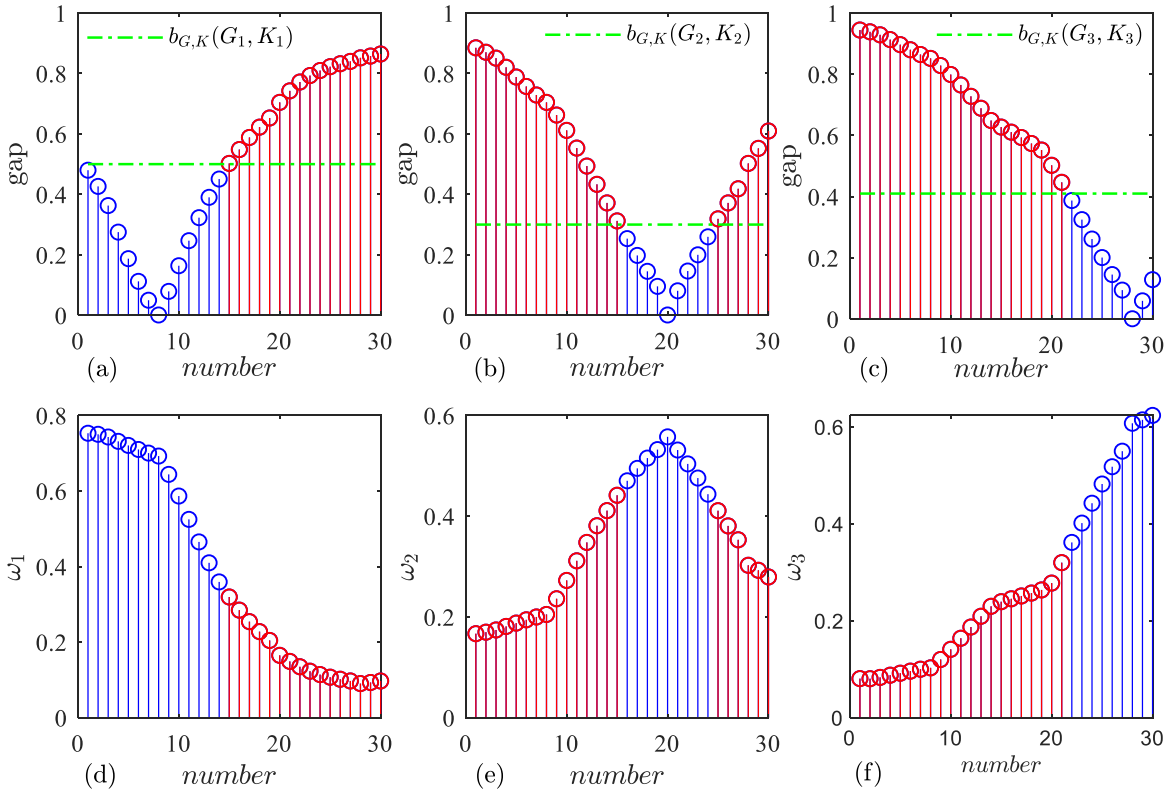


Fig. 4: (a)-(c) Gap between each nominal local model (G_1, G_2, G_3) and all 30 local models (needle bars) , with the stability margin of each corresponding local controller shown as a green dash-dotted line. (d)-(f) Corresponding gap metric-based weighting functions calculated via (15) with $\beta = 1$. Red vertical lines highlight local models that violate the stability condition (5).

To evaluate the set-point tracking performance of the nonlinear CSTR process (22), three control scenarios are investigated and compared as follows:

- **Scenario 1** employs the traditional gap metric-based weighting function defined by (15) with $\beta = 1$. The resulting output and manipulated input variables are denoted as $C_{b-\mathrm{M}_1}$ and $Q_{1-\mathrm{M}_1}$, respectively.
- **Scenario 2** utilizes the proposed stability-aware weighting functions from (18), with $\beta = 1$. The corresponding signals for this case are labeled $C_{b-\mathrm{M}_2}$ and $Q_{1-\mathrm{M}_2}$.
- **Scenario 3** enhances the stability-aware approach from Scenario 2 by introducing auxiliary local controllers for the boundary models 15 and 23. These auxiliary controllers, designed as $K_{A1} = 4.68 + \frac{2.62}{s}$ and $K_{A2} = 4.89 + \frac{0.82}{s}$, each possess a stability margin of 0.2. The output and input for this final configuration are denoted as $C_{b-\mathrm{M}_3}$ and $Q_{1-\mathrm{M}_3}$, respectively.

The set-point tracking performance for the three investigated scenarios is presented in Fig. 5. Visually, all three methods achieve comparable reference tracking performance. A closer inspection reveals that the traditional method (Scenario 1) yields a marginally superior response. However, as established previously, this approach lacks theoretical guarantees for closed-loop stability across the entire operating range.

The responses of the proposed stability-aware method (Scenario 2) and its enhanced version with auxiliary controllers (Scenario 3) are nearly identical in terms of output tracking. The critical distinction emerges in the control effort, Q_1. As seen in Fig. 5, Scenario 2 induces noticeable oscillations in the manipulated input, particularly during transitions between operating sub-regions at approximately 70 and 100 minutes. In contrast, Scenario 3 successfully mitigates these transients, producing a significantly smoother control signal. This demonstrates that the introduced auxiliary local models effectively fulfill their intended role by facilitating a stable and smooth transition across the operating space, thereby eliminating undesirable control actuation chattering.

To quantitatively compare the performance of the three scenarios, the MIAE values were computed using the following expression:

$$\text{MIAE} = \frac{1}{n}\sum_{i=1}^{n}|\text{ref}(t_i) - C_b(t_i)| \tag{23}$$

Where $\text{ref}(t_i)$, $C_b(t_i)$ are the reference input and output signals at time t_i, respectively, and n is the total number of data points. The MIAE values for the three scenarios are 0.19, 0.23, and 0.21, respectively. Although Scenario 1 yields the lowest MIAE, it cannot theoretically guarantee closed-loop stability. In contrast, Scenario 2 exhibits the highest MIAE due to the oscillations induced by hard switching. However, unlike Scenario 1, Scenario 2 explicitly enforces closed-loop stability condition during transitions between different operating regions. Scenario 3, which incorporates auxiliary models, achieves a lower MIAE than Scenario 2 while still preserving closed-loop stability condition during region switches. Therefore, by considering all factors, Scenario 3 demonstrates the best overall reference tracking performance.

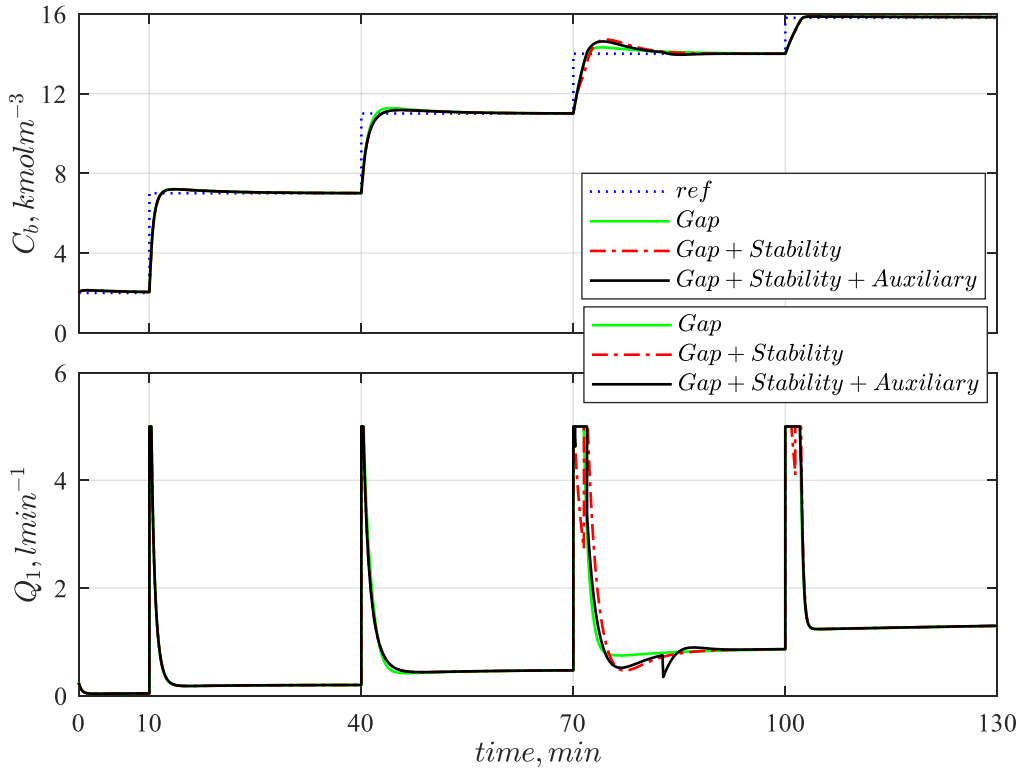


Fig. 5: Closed-loop output response and manipulated input under the three investigated weighting strategies.

The weighting functions corresponding to Scenarios 1, 2, and 3 are presented in Figs. 6, 7, and 8, respectively. Fig. 6 shows the final weighting functions for each of the three nominal local models, denoted ω_1, ω_2, and ω_3, together with the instantaneous gap metrics μ_1, μ_2, and μ_3 between the linearized model at time t and each nominal local model. A detailed examination of Fig. 6 leads to the following observations:

- **Between 40 and 70 minutes,** only nominal local model 1 satisfies the stability condition ($\mu_1 \leq b_{G,K}(G_1, K_1)$). Nevertheless, the other two nominal local models still receive non-zero weights in the control signal computation (see the red and green lines in ω).
- **Between 70 and 100 minutes,** only nominal local model 2 satisfies the stability condition ($\mu_2 \leq b_{G,K}(G_2, K_2)$), yet the remaining two nominal local models contribute non-zero (albeit small) weights to the final control signal (blue and green lines in ω).
- **Between 100 and 130 minutes,** only nominal local model 3 satisfies the stability condition ($\mu_3 \leq b_{G,K}(G_3, K_3)$), but local model 2 continues to influence the control signal with a non-zero weight (red line in ω).

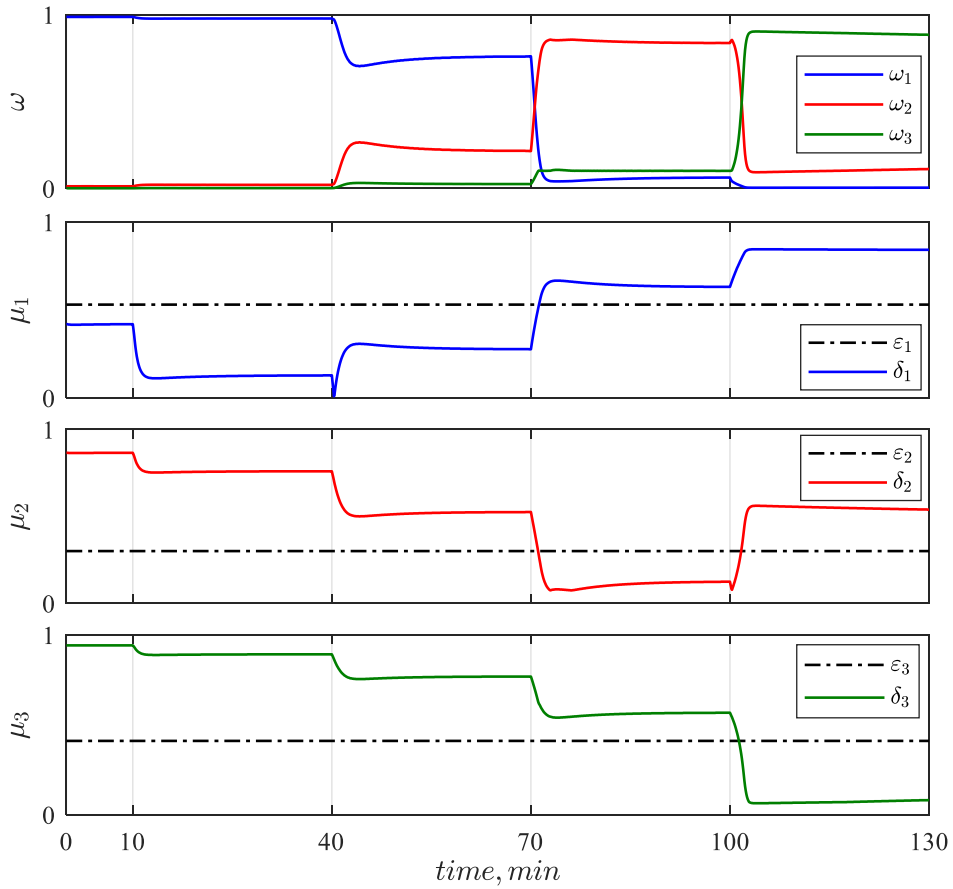


Fig. 6: Weighting functions based only on gap metric.

Thus, it is evident that during certain time intervals, some local controllers that do not satisfy the stability condition are nonetheless assigned non-zero weights in the final control signal, precisely the issue addressed in this paper. To remedy this drawback, we examine the outcome of Scenario 2, shown in Fig. 7. Careful analysis of this figure reveals:

- **From 0 to 70 minutes,** only nominal local model 1 fulfills the stability condition ($\mu_1 \leq b_{G,K}(G_1, K_1)$); consequently, only local controller 1 contributes to the final control signal, while the other two local controllers are assigned zero weights (red and green lines in ω).
- **Between 70 and 100 minutes,** nominal local model 2 meets the stability condition ($\mu_2 \leq b_{G,K}(G_2, K_2)$) and receives a maximum weight of 1, whereas the other two local controllers have zero weights (blue and green lines in ω).
- **Between 100 and 130 minutes,** the stability condition dictates that local controller 3 should be active ($\mu_3 \leq b_{G,K}(G_3, K_3)$) while the other two remain inactive ($\mu_1 > b_{G,K}(G_1, K_1)$ and $\mu_2 > b_{G,K}(G_2, K_2)$), which is precisely what occurs (blue and red lines in ω).

- **Around 70 and 100 minutes,** a hard switching from one local controller to another is observed. The effect of this hard switching on the final control signal is manifested as noticeable oscillations in the manipulated input, as previously described in Fig. 5.

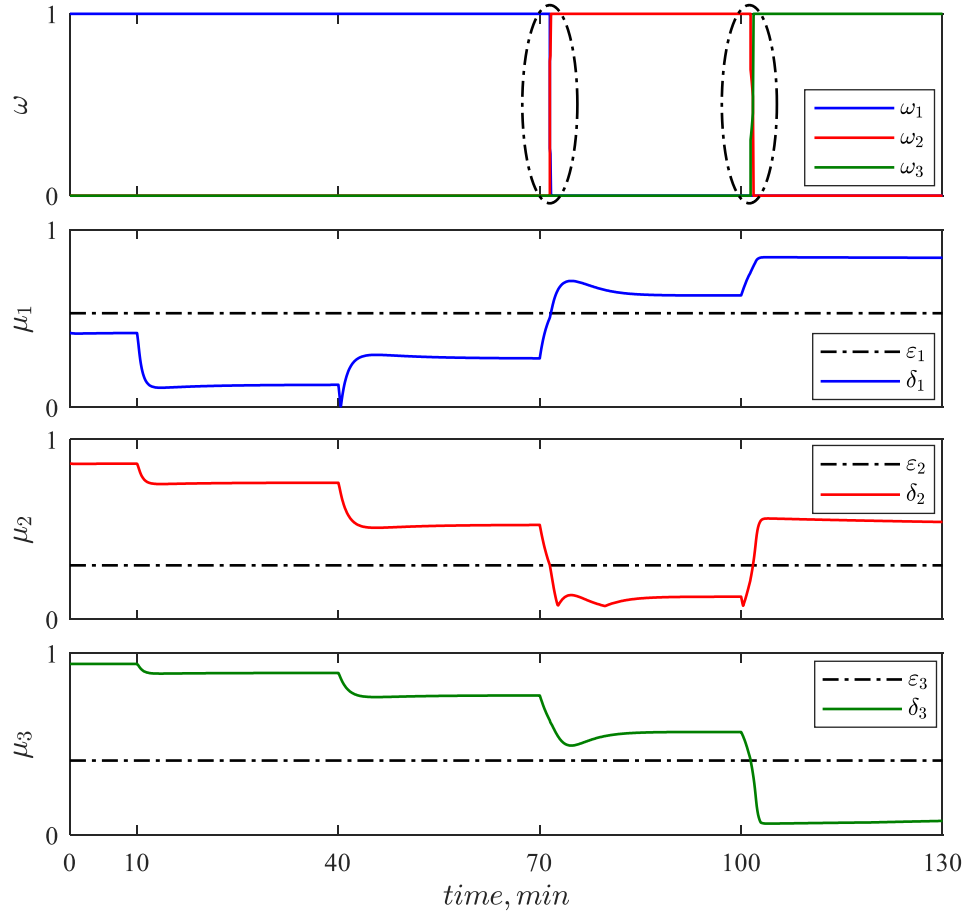


Fig. 7: Weighting functions based on gap metric-stability.

Thus, although the method proposed in Scenario 2 resolves the stability oversight inherent in Scenario 1, the hard switching between operating regions can adversely affect the performance of the global controller. The result of our proposed solution to this issue is displayed in Fig. 8, from which the following conclusions can be drawn:

- **Throughout all time intervals,** the stability condition for selecting local controllers is enforced as in Scenario 2; only stabilizing controllers are assigned non-zero weights in the final control signal.
- **At region transition instants,** the auxiliary local controllers become active, thereby preventing hard switching and abrupt transitions between regions. The resulting effect of this mechanism on the final control signal is a smooth transition across the operating regions, as previously illustrated in Fig. 5.

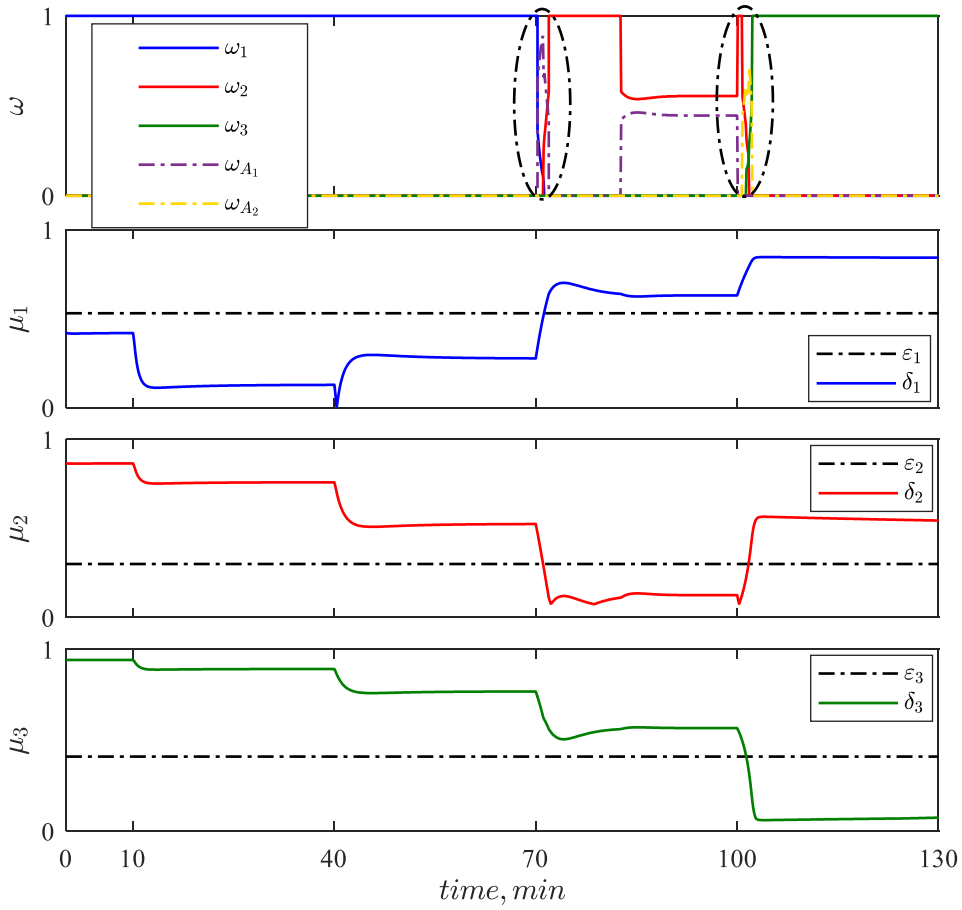


Fig. 8: Weighting functions based on gap metric-stability-auxiliary.

To evaluate the alternative weighting function form given in (19), the closed-loop response of the nonlinear system (22) was simulated using the two weight functions, namely the $1-\overline{\mu}$ and $\frac{1}{\overline{\mu}}$ forms. The results are shown in Fig. 9, alongside the response obtained with the weighting functions of the form (18) for comparison. The corresponding weight values for both cases are reported separately in Fig. 10. In all simulations, the tuning parameter was set to $\beta = 3$, and auxiliary controllers were included in the multiple model control framework. A careful inspection of Fig. 9 reveals that both weighting forms yield very similar reference tracking performance, with only negligible differences observed at certain time instants (e.g., at 70, 80, and 105 minutes). For a more precise comparison, the MIAE values were calculated, giving 0.21 for (18) and 0.20 for (19). Thus, both forms perform comparably in the proposed multiple model control framework, and either can be used effectively.

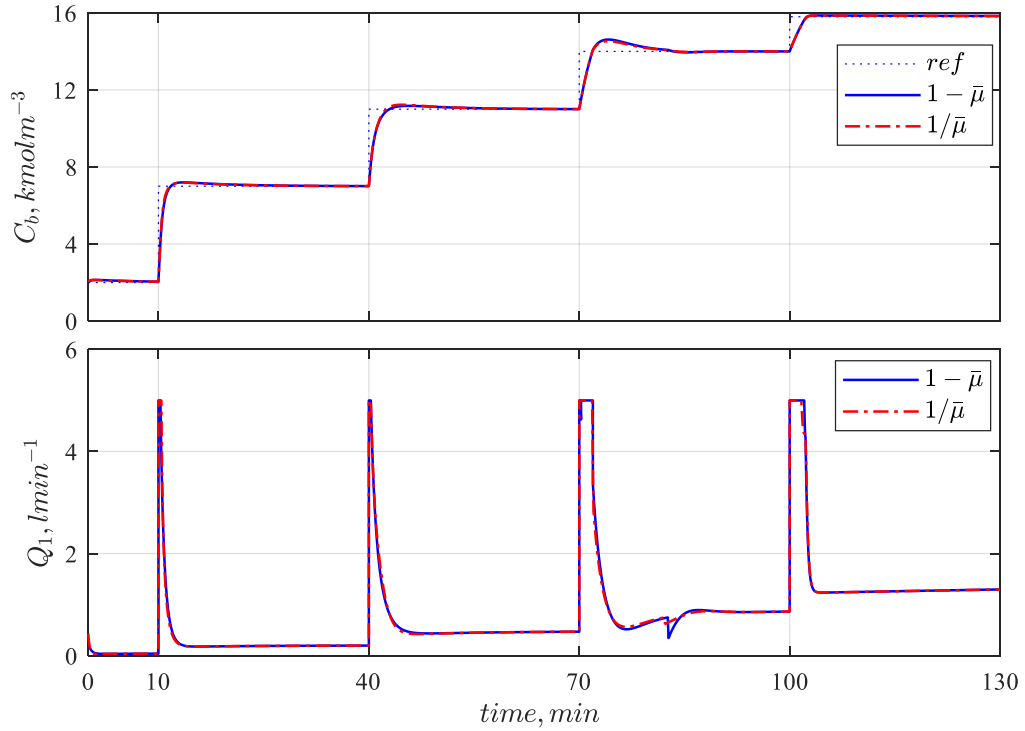


Fig. 9: Closed-loop output response and manipulated input obtained using the two weighting function forms (18) and (19).

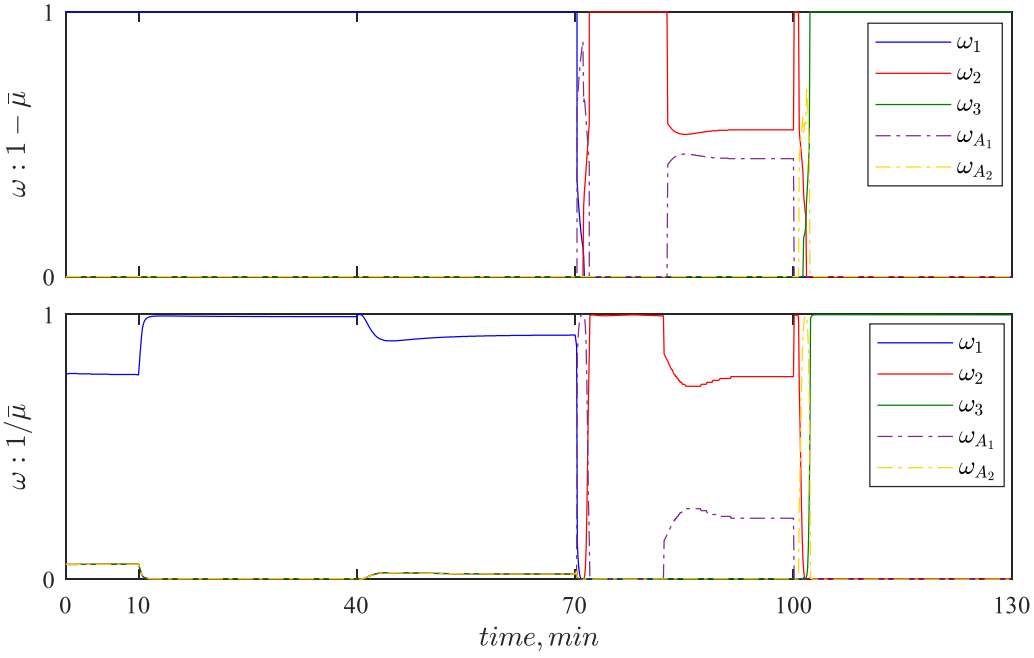


Fig. 10: Weighting functions obtained using the two forms (18) and (19).

In summary, the method proposed in this paper successfully resolves the critical shortcoming of conventional gap metric-based weighting functions, namely, the neglect of local controller stability during the weighting functions calculation stage. Furthermore, the use of auxiliary local controllers plays a pivotal role in smoothing transitions between operating regions. Thus,

the proposed stability-aware gap metric weighting framework not only resolves the critical deficiency inherent in conventional approaches but also delivers effective overall performance.

6. Conclusion

This paper has addressed a critical yet previously overlooked issue in gap metric-based blending approaches for multiple model control: the neglect of closed-loop stability during the weighting functions calculation stage. While conventional methods compute blending weights solely based on the open-loop distance between local models, they offer no guarantee that the resulting global controller preserves stability, even when each local controller is individually stabilizing. This fundamental limitation was identified and rigorously analyzed through a highly nonlinear CSTR case study.

To resolve this shortcoming, a novel stability-aware weighting framework was proposed. The core idea is to explicitly incorporate stability conditions into the gap metric-based weights calculation, ensuring that only locally stabilizing controllers contribute to the global control signal. However, this direct integration was shown to induce hard switching and oscillations at region boundaries. To overcome this side effect, an auxiliary local model strategy was introduced, which activates additional stabilizing controllers during transitions, thereby enabling smooth interpolation without compromising stability.

The proposed framework was evaluated through three comparative scenarios. The results demonstrate that the conventional gap metric–based method (Scenario 1) achieves marginally better tracking performance. However, it cannot theoretically guarantee closed-loop stability. This risk materializes when the system traverses regions where certain nominal local models do not satisfy the stability condition. In such cases, those models receive non-zero weights, potentially compromising stability. The stability-aware method without auxiliary controllers (Scenario 2) successfully eliminates the stability oversight but introduces hard switching and oscillations in the manipulated input. In contrast, the full proposed method (Scenario 3) combines stability-aware weighting with auxiliary local models, achieving both guaranteed stability and smooth transitions. The auxiliary controllers effectively prevent oscillations and ensure seamless interpolation across operating regions.

References

[1] R. Murray-Smith and T. Johansen, Multiple Model Approaches To Nonlinear Modelling And Control. Taylor & Francis, 1997.

[2] M. Ahmadi, “Nonlinearity Measure for Nonlinear Dynamic Systems Using a Multimodel Framework,” Journal of Computational and Nonlinear Dynamics, vol. 20, no. 4, 2025. doi:

10.1115/1.4067913.

[3] A. Lovi, B. Fidan, and C. Nielsen, “Multiple Model Reference Adaptive Control With Blending for Nonsquare Multivariable Systems,” IEEE Transactions on Automatic Control, vol. 70, no. 9, pp. 5780-5791, 2025. doi: 10.1109/TAC.2025.3549291.

[4] H. Yu, Y. Ma, and B. Jiang, “Multiple-Model Prescribed Performance Control on 6-DOF Spacecraft With Actuation Sign Fault,” IEEE Transactions on Aerospace and Electronic Systems, vol. 61, no. 6, pp. 18155-18169, 2025. doi: 10.1109/TAES.2025.3609754.

[5] L. Zhang, K. Xu, J. Yang, M. Han, and S. Yuan, “Transition-Dependent Bumpless Transfer Control Synthesis of Switched Linear Systems,” IEEE Transactions on Automatic Control, vol. 68, no. 3, pp. 1678-1684, 2023. doi: 10.1109/TAC.2022.3152721.

[6] M. Fathi, H. Bolandi, B. G. Vaghei, and S. Ebadollahi, “On Feasibility and Asymptotically Stability of Switched Systems Using Adaptive Multi-Model Predictive Control,” Complex System Modeling and Simulation, vol. 5, no. 2, pp. 155-175, 2025. doi: 10.23919/CSMS.2024.0030.

[7] H. Zengin, N. Zengin, B. Fidan, and A. Khajepour, “Blending based multiple-model adaptive control of multivariable systems with application to lateral vehicle motion control,” European Journal of Control, vol. 58, pp. 1-10, 2021.

[8] L. Dutta and D. K. Das, “Multiple-model adaptive explicit predictive control for nonlinear MIMO system,” Journal of Control and Decision, vol. 12, no. 4, pp. 695-710, 2025.

[9] A. Yewale, R. Methekar, and S. Agrawal, “Dynamic analysis and multiple model control of continuous microbial fuel cell (CMFC),” Chemical Engineering Research and Design, vol. 148, pp. 403-416, 2019/08/01/ 2019. doi: https://doi.org/10.1016/j.cherd.2019.06.007.

[10] H. Zhu, G. Zhao, L. Sun, and K. Y. Lee, “Nonlinear Predictive Control for a Boiler–Turbine Unit Based on a Local Model Network and Immune Genetic Algorithm,” Sustainability, vol. 11, no. 18, p. 5102, 2019. [Online]. Available: https://www.mdpi.com/2071-1050/11/18/5102.

[11] M. Ahmadi, P. Rikhtehgar, and M. Haeri, “A multi-model control of nonlinear systems: A cascade decoupled design procedure based on stability and performance,” Transactions of the Institute of Measurement and Control, vol. 42, no. 7, pp. 1271-1280, 2020. doi: 10.1177/0142331219888368.

[12] M. Ghazali *et al.*, “Energy Management of a Semi-Autonomous Truck Using a Blended Multiple Model Controller Based on Particle Swarm Optimization,” Energies, vol. 18, no. 11, p. 2893, 2025. [Online]. Available: https://www.mdpi.com/1996-1073/18/11/2893.

[13] J. Du and T. A. Johansen, “A gap metric based weighting method for multimodel predictive control of MIMO nonlinear systems,” Journal of Process Control, vol. 24, no. 9, pp. 1346-1357, 2014.

[14] J. Du, L. Zhang, and J. Chen, “A new weighting method based on gap metric for multi-model predictive control,” in 2019 Chinese Control And Decision Conference (CCDC), 2019: IEEE, pp. 1746-1746.

[15] K. Elkhalil and A. Zribi, “Multimodel Control of Nonlinear Systems Using A New Weighting Method Based on Gap Metric,” in 2023 IEEE 11th International Conference on Systems and Control (ICSC), 2023: IEEE, pp. 748-753.

[16] E. Arslan, M. C. Çamurdan, A. Palazoglu, and Y. Arkun, “Multimodel scheduling control of nonlinear systems using gap metric,” Industrial & engineering chemistry research, vol. 43, no. 26, pp. 8275-8283, 2004.

[17] G. M. Prasad and A. S. Rao, “Evaluation of gap-metric based multi-model control schemes for nonlinear systems: An experimental study,” ISA transactions, vol. 94, pp. 246-254, 2019.

[18] M. Ahmadi and M. Haeri, “A new structured multimodel control of nonlinear systems by integrating stability margin and performance,” Journal of Dynamic Systems, Measurement, and Control, vol. 139, no. 9, 2017.

[19] A. El-Sakkary, “The gap metric: Robustness of stabilization of feedback systems,” IEEE Transactions on Automatic Control, vol. 30, no. 3, pp. 240-247, 2003.

[20] T. T. Georgiou, “On the computation of the gap metric,” Systems & Control Letters, vol. 11, no. 4, pp. 253-257, 1988.

[21] K. Zhou and J. C. Doyle, Essentials of robust control. Prentice hall Upper Saddle River, NJ, 1998.

[22] A. A. Jalali and H. Golmohammad, “An optimal multiple-model strategy to design a controller for nonlinear processes: A boiler-turbine unit,” Computers & chemical engineering, vol. 46, pp. 48-58, 2012.

[23] J. Du, C. Song, Y. Yao, and P. Li, “Multilinear model decomposition of MIMO nonlinear systems and its implication for multilinear model-based control,” Journal of Process Control, vol. 23, no. 3, pp. 271-281, 2013.

[24] P. Rikhtehgar, M. Ahmadi, and M. Haeri, “A cascade multiple-model predictive controller of nonlinear systems by integrating stability and performance,” in 2019 27th Iranian conference on electrical engineering (ICEE), 2019: IEEE, pp. 951-955.