Author/s:
Yang, L;Li, D;Ma, A;Xi, Y;Pu, Y;Tan, Y

Title:
Input-mapping based data-driven model predictive control for unknown linear systems via online learning

Date:
2025-05-10

Citation:
Yang, L., Li, D., Ma, A., Xi, Y., Pu, Y. & Tan, Y. (2025). Input-mapping based data-driven model predictive control for unknown linear systems via online learning. International Journal of Robust and Nonlinear Control, 35 (7), pp.2537-2565. https://doi.org/10.1002/rnc.6237.

Received: Added at production | Revised: Added at production | Accepted: Added at production
DOI: xxx/xxxx

SPECIAL ISSUE ARTICLE

Input-mapping based data-driven model predictive control for unknown linear systems via online learning

Lingyi Yang[1] | Dewei Li*[1] | Aoyun Ma[1] | Yugeng Xi[1] | Ye Pu[2] | Ying Tan[2]

[1]the Department of Automation, Shanghai Jiaotong University, Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai Engineering Research Center of Intelligent Control and Management, Shanghai, China
[2]the School of Electrical, Mechanical and Infrastructure Engineering, the University of Melbourne, Victoria, Australia

Correspondence
*Dewei Li, the Department of Automation, Shanghai Jiaotong University, Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai Engineering Research Center of Intelligent Control and Management, Shanghai, 200240. Email: dwli@sjtu.edu.cn

Summary

Data-driven model predictive control (MPC) is an effective control method in controlling unknown constrained systems. The existing data-driven MPC methods either estimate the system online (adaptive) with extra computation efforts, or use the initially measured trajectory from offline trials to design controller. The offline trials are economically expensive for many practical systems. To overcome these limitations, we propose a multi-step input-mapping data-driven scheme. It maps the current and future inputs to the past online measured input-state trajectories and expresses the future state as a linear combination of the past states. A moving data window is used to update the data online. Based on this scheme, two computationally efficient multi-step input-mapping data-driven robust model predictive controllers with guarantee of feasibility and stability are proposed. The simulation results verify the improved performance and better computation efficiency of the resulting methods.

KEYWORDS:
Input-mapping data-driven approach, online learning, robust model predictive control

1 | INTRODUCTION

Model predictive control (MPC) has been widely used to control constrained systems in many engineering applications [1,2]. Traditional MPC uses a model to predict the future state and determine the control sequence. Since the accuracy of prediction depends on the model, having an accurate model is crucial for obtaining good performance. The robust model predictive control (RMPC) approach is known to handle modeling error by using a given bound of uncertainty to predict and design controller [1,3,4]. The control policy is constructed based on a fixed feedback gain. The feedback gain affects the size of feasible region and performance and choice of a single feedback need to compromise these objectives [5]. This can be settled by developing interpolation robust model predictive control (IRMPC) which blends the advantages of different feedback gains [5,6,7]. Yet, the RMPC methods are based on a prior bound of uncertainty, which is conservative for the time-invariant or slowly time-varying plant [8].

For control of unknown systems, data-driven model predictive control has been extensively studied as it can exploit data to gain knowledge of unknown systems. The main research can be classified according to whether the data is online or offline collected. The offline data-driven approaches work on trajectories obtained from offline tests. With the trajectories, either implementing model identification or directly utilizing trajectories as an implicit model to predict future states without identification are both widely investigated. One representative identification based approach is subspace model predictive control which identifies the state space system model from data-Hankel matrix [9]. On the other hand, the research focusing on offline measurements without performing identification includes [10,11,12,13] which directly use the offline measurements to predict future trajectories and design the controller based on behavioural theory [14]. Under the assumption of an initially measured trajectory satisfying persistent excitation condition, [10] presents a computationally efficient data-driven MPC scheme with theoretical guarantee of stability and feasibility. However, for some large scale systems, offline test is economically expensive and lack of insufficient number of data or appropriate order of excitation will lead to poor excitation and control performance [15]. For unstable systems, closed-loop tests are required and lack of excitation is likely to arise [11]. Besides, offline data based methods need to recollect the data when the model dynamic changes, otherwise instability or poor performance will result.

To mention online data-driven methods, recent research has mainly focused on adaptive model predictive control (AMPC) [16], model free adaptive control, learning-based MPC, input-mapping data-driven (IMD) MPC, etc. Based on adaptive control, AMPC takes advantage of identification tools to update prediction model while drawing support from RMPC to ensure robust constraint satisfaction and stability [8,17,18,19,20,21]. An issue arising from online estimation is that the performance of AMPC heavily relies on the excitation effect of data which is not always satisfied and remains an open issue [18,19]. The computation resources spent in implementing estimation of unknown parameter or uncertainty set is also a nontrivial problem [8]. Different from AMPC, in which state prediction is based on an estimated model, the IMD MPC scheme we proposed in [22] directly uses the online measured input-state trajectories to design controller and predict future states and has been successfully applied in a robot manipulator example [23]. In particular, it uses an input knowledge to generate an online data-driven strategy which builds a mapping between the current input-state trajectories and past input-state measurements and uses a linear combination of past trajectories to predict future states. The main characteristics of this method is that no initially measured trajectories are required as in [10,12,13]. The main limitation of this scheme is that it assumes the control horizon to be one and simply referred as IMD-1-step in the following. Meanwhile, it cannot cope with asymmetric constraints as an ellipsoid invariant set is adopted. The use of ellipsoid set results in an semi-definite program to be solved online that is computationally demanding.

In this paper, we propose two new multi-step input-mapping data-driven MPC methods for unknown linear time-invariant (LTI) systems with stability and feasibility guarantees. The main contribution of this article are twofold. First, we extend the IMD scheme of single-step in [22] to a more general case with control horizon of arbitrary length. By mapping the current and future input and state to the past online measured input-state trajectories, the future state can be represented as a linear combination of past online state measurements. This multi-step paradigm enables to introduce more degrees of freedom and improved performance compared to [22]. Without model estimation process, the multi-step IMD scheme directly uses past online measurements to design controller, saving the computation efforts of AMPC in updating uncertainty set [8]. It should be emphasized that this scheme requires no initially online measured trajectories as in [10,12]. Second, we propose two computationally efficient multi-step IMD-RMPC algorithms by blending the multi-step IMD scheme with two RMPC methods using polyhedral invariant set in [6]. One is the RMPC based on fixed feedback gain and an IMD-RMPC scheme is proposed. Since the feedback gain determines the size of attractive region and control performance and the choice of a fixed single feedback gain need to compromise these objectives, the other interpolation RMPC, which merges the advantages of different feedback gains, is combined with IMD scheme and an IMD-IRMPC is hence developed. The IMD-IRMPC can enlarge the feasible region of IMD-RMPC. The recursive feasibility and closed-loop stability of two schemes are ensured. And only a quadratic program is required to be solved online for both IMD-RMPC and IMD-IRMPC. Based on polyhedral rather than ellipsoid invariant set, the proposed two schemes can deal with asymmetric constraint in a non-conservative way and possess higher computation efficiency compared to [22]. With incorporation of IMD scheme, the conservatism of two RMPC methods in [6] can be reduced, yielding an improved performance. Due to the combination with RMPC technique, the proposed two methods can deal with model mismatch which is not considered by [10,12]. We demonstrate the advantages of our proposed methods through numerical examples.

To summarize, the proposed multi-step IMD-RMPC methods (i) have less conservatism than [6] by incorporating an input-mapping data-driven scheme, (ii) can achieve better performance than IMD-1-step in [22] by extending the control horizon to any arbitrary length and developing a new multi-step IMD scheme, (iii) possess better computation efficiency and less conservatism in handling asymmetric constraint than IMD-1-step by combining the IMD scheme with two RMPC techniques using polyhedral invariant set, (iv) can achieve better computation efficiency compared to AMPC approach in [8], (v) do not require an initially measured trajectory as in [10,12] and can cope with model mismatch which is not considered by [10,12].

This paper is organized as follows. Section II introduces the basic principle of the input-mapping data-driven strategy. Main results are stated in Section III. A multi-step input-mapping data-driven interpolation RMPC algorithm is presented in Section IV. Section V provides an illustrative example, followed by conclusions.

Notation The sets $\mathbb{N}, \mathbb{R}$ are respectively the set of non-negative integers and real numbers. For some $c_1, c_2 \in \mathbb{R}, c_2 \geq c_1$, let $\mathbb{R}_{\geq c_1}, \mathbb{R}_{[c_1,c_2]}$ denote the sets $\{t \in \mathbb{R} : t \geq c_1\}, \{t \in \mathbb{R} : c_1 \leq t \leq c_2\}$ respectively. For some $c_1, c_2 \in \mathbb{N}, c_2 \geq c_1$, let $\mathbb{N}_{\geq c_1}, \mathbb{N}_{[c_1,c_2]}$ denote the sets $\{t \in \mathbb{N} : t \geq c_1\}, \{t \in \mathbb{N} : c_1 \leq t \leq c_2\}$ respectively. The notation $\|\cdot\|_\infty$ denotes the vector infinity norm and the corresponding induced matrix norm. The symbol $\|x\|_Q^2 = x^T Q x$ is used to simplify the quadratic form with respect to $Q > 0$.

$Co\{\cdot\}$ denotes the convex hull of finite points. The $\texttt{col}[\cdot]$ operation represents $\texttt{col}\left[x_1, x_2, \cdots, x_n\right] = [x_1^T, x_2^T, \cdots, x_n^T]^T$. The symbol $\mathbf{0}, \mathbf{1}$ and I_n denote the all-zero matrix (vector) with appropriate dimensions to be inferred from context, a column vector of ones and the identity matrix of dimension n, respectively. Let $x(i|k)$ denote the predicted system state i steps ahead from time k where $x(k) = x(0|k)$. The notation $(S)_i$ denotes the i^{th} column (or element) of matrix (or vector) S. $\mathcal{L}_i = \{(j_1, \cdots, j_i) : j_r \in \mathbb{N}_{[1,L]}, r = 1, \cdots, i\}$ denotes the set of all possible values of the sequences $(j_1, \cdots, j_i)$. The $\tilde{j}_{1:i}$ is an abbreviation of sequence $(j_1, \cdots, j_i)$.

2 | PROBLEM FORMULATION AND PRELIMINARIES

2.1 | Problem formulation

This paper focuses on unknown discrete time invariant system,

$$x(k+1) = Ax(k) + Bu(k), \tag{1}$$

subject to a polyhedral constraint

$$Fx(k) + Gu(k) \leq \mathbf{1}, \tag{2}$$

with state $x(k) \in \mathbb{R}^{n_x}$, control input $u(k) \in \mathbb{R}^{n_u}$. The matrices A and B are unknown, $A \in \mathbb{R}^{n_x \times n_x}, B \in \mathbb{R}^{n_x \times n_u}$. And $F \in \mathbb{R}^{n_c \times n_x}, G \in \mathbb{R}^{n_c \times n_u}$ are two assigned matrices.

Assumption 1. The unknown system (1) can be depicted as

$$A = A_0 + \Delta_A^{\mathrm{tr}}, B = B_0 + \Delta_B^{\mathrm{tr}}, [\Delta_A^{\mathrm{tr}}, \Delta_B^{\mathrm{tr}}] \in \Xi_\Delta \triangleq Co\{[\Delta_A^{(1)}, \Delta_B^{(1)}], \cdots, [\Delta_A^{(L)}, \Delta_B^{(L)}]\}, \tag{3}$$

with known matrices A_0, B_0 and true unknown matrices $[\Delta_A^{\mathrm{tr}}, \Delta_B^{\mathrm{tr}}]$ lying in a given convex and compact sets Ξ_Δ which is constituted by a convex hull of known vertex matrices $\{[\Delta_A^{(1)}, \Delta_B^{(1)}], [\Delta_A^{(2)}, \Delta_B^{(2)}], \cdots, [\Delta_A^{(L)}, \Delta_B^{(L)}]\}$ with $L > 0$ denoting the number of vertices. Throughout this paper, we denote $[A^{(j)}, B^{(j)}] \triangleq [A_0 + \Delta_A^{(j)}, B_0 + \Delta_B^{(j)}]$ with $j \in \mathbb{N}_{[1,L]}$ and the true model lies in a given set defined as

$$[A, B] \in \Xi \triangleq Co\{[A^{(1)}, B^{(1)}], \cdots, [A^{(L)}, B^{(L)}]\}. \tag{4}$$

We assume that states are perfectly measured. The problem of interest is to design a stabilizing data-driven MPC controller for the unknown system (1) satisfying Assumption 1 based on the online measurements of the system state and input. The controller is designed to ensure the desirable closed-loop performance and constraint satisfaction.

2.2 | Preliminaries of online input-mapping data-driven learning

In this section, we briefly recall the IMD strategy of 1-step proposed in [22]. At time k, given past N_L input-state measurements $\{u(t), x(t)\}_{t=k-N_L}^{t=k-1}$ of the unknown system (1) and current state $x(k)$, we define matrices $X_k^- \in \mathbb{R}^{n_x \times N_L}, U_k^- \in \mathbb{R}^{n_u \times N_L}, X_k^+ \in \mathbb{R}^{n_x \times N_L}$ which collect these measurements as below,

$$X_k^- := \left[x(k-1), x(k-2), \cdots, x(k-N_L)\right], \tag{5}$$

$$U_k^- := \left[u(k-1), u(k-2), \cdots, u(k-N_L)\right], \tag{6}$$

$$X_k^+ := \left[x(k), x(k-1), \cdots, x(k-N_L+1)\right]. \tag{7}$$

The above matrices play the role of moving window which are updated at each sampling time. At time $k = 0$, the above data sets are initialized as $X_k^- = \mathbf{0}, U_k^- = \mathbf{0}, X_k^+ = \mathbf{0}$ with appropriate dimensions. And we have

$$x(k-i+1) = Ax(k-i) + Bu(k-i). \tag{8}$$

By taking advantage of homogeneity of linear system (1), we multiply a scalar $\beta \in \mathbb{R}$ on left and right sides of (8), then

$$\beta x(k-i+1) = A\beta x(k-i) + B\beta u(k-i). \tag{9}$$

This implies that if the current state and input satisfy $\{x(k), u(k)\} = \{\beta x(k-i), \beta u(k-i)\}$, the next step state prediction equals $x(1|k) = \beta x(k-i+1)$. Further, by making use of additive characteristic of linear system, we sum (9) from $i = 1$ to $i = N_L$, then

$$\sum_{i=1}^{N_L} \beta_i x(k-i+1) = A \sum_{i=1}^{N_L} \beta_i x(k-i) + B \sum_{i=1}^{N_L} \beta_i u(k-i), \tag{10}$$

where $\beta_i \in \mathbb{R}$ with $i \in \mathbb{N}_{[1,N_L]}$. This implies that if the current state and input satisfy $\{x(k), u(k)\} = \{\sum_{i=1}^{N_L} \beta_i x(k-i), \sum_{i=1}^{N_L} \beta_i u(k-i)\}$, then the next step state prediction equals $x(1|k) = \sum_{i=1}^{N_L} \beta_i x(k-i+1)$. Different values of β_i where $i \in \mathbb{N}_{[1,N_L]}$ correspond to various control input and can be used to optimize performance, which will be introduced in the next section.

Equations (9)-(10) show how the IMD strategy builds a mapping between the current input-state signals $\{x(k), u(k), x(1|k)\}$ and the past online input-state measurements $\{x(k-i), u(k-i), x(k-i+1)\}$ collected in X_k^-, U_k^-, X_k^+. This lays a foundation of using past input signals and state online measurements to express the current and future input and state predictions. The advantage of using past data to express the future state is that no online estimation process is needed as in AMPC and the past input signals can be directly used to design the current and future control input, which has great simplicity. It is noted that a similar data-driven MPC approach which represents the future input and output as a linear combination of past input and output trajectories has been proposed in [10]. The requirement of obtaining sufficient measurements satisfying persistent excitation condition through offline trials is economically expensive for some large-scale processes. To tackle this issue, we propose an online multi-step IMD strategy which does not require any offline trials in the following section.

3 | DATA-DRIVEN MPC WITH FIXED FEEDBACK GAIN CONTROL

In this section, we present a computationally efficient online multi-step IMD-RMPC scheme with guarantee of recursive feasibility and stability. With online measurements, the proposed scheme uses the input knowledge to map the current and future inputs and states to the past input and state trajectories. In particular, our proposed method constructs the future input $u(i|k)$ as a linear combination of past input signals $v(i|k)$ and the input residual $\mu(i|k)$, i.e., $u(i|k) = v(i|k) + \mu(i|k)$, and constructs the current and future states as a linear combination of past states $z(i|k)$ and the state residual $e(i|k)$, i.e., $x(i|k) = z(i|k) + e(i|k)$. Further, we combine our proposed data-driven technique with the RMPC method. Section 3.1 contains the multi-step input-mapping data-driven technique. In Section 3.2 and 3.3, we combine the data-driven scheme with the RMPC method in which conditions for ensuring robust constraint satisfaction are presented, as well as the data-driven cost function. The whole algorithm is summarized in Section 3.4 with a theoretical analysis of recursive feasibility and stability.

3.1 | Data-driven control strategy

We use past input signals from the data set U_k^- in (6) to design the control input which is defined as follows,

$$u(i|k) = \begin{cases} v(i|k) + \mu(i|k), & i \in \mathbb{N}_{[0,N-1]}, \\ Kx(i|k), (= v(i|k) + \mu(i|k)), & i \in \mathbb{N}_{\geq N}, \end{cases} \tag{11}$$

in which K is a pre-determined feedback gain such that the closed-loop system matrix

$$\phi^{(j)} = A^{(j)} + B^{(j)} K, \tag{12}$$

is quadratically stable for all $j \in \mathbb{N}_{[1,L]}$ (the method of determining K can be found in [6]-[7]), and the linear combination of past inputs $v(i|k) \in \mathbb{R}^{n_u}$ is denoted as

$$v(i|k) = \textstyle\sum_{l=1}^{N_L-i} (\theta(i|k))_l u(k-l), i \in \mathbb{N}_{[0,N-1]}, \quad (v(i|k) = 0 \text{ for } i \in \mathbb{N}_{\geq N}), \tag{13}$$

where the vector $\theta(i|k) \in \mathbb{R}^{N_L - i}$ is the linear combination parameter optimized online and $(\theta(i|k))_l$ denotes the l^{th} element of vector $\theta(i|k)$. The input residual $\mu(i|k) \in \mathbb{R}^{n_u}$ is parameterized as

$$\mu(i|k) = Ke(i|k) + c(i|k), i \in \mathbb{N}_{\geq 0}, \tag{14}$$

where $c(i|k) \in \mathbb{R}^{n_u}$ denotes the perturbation vector optimized online ($c(i|k) = 0$ for $i \in \mathbb{N}_{\geq N}$). The state residual $e(i|k) \in \mathbb{R}^{n_x}$ is defined from the decomposition of the current and future predicted state $x(i|k)$ as a linear combination of past states $z(i|k)$

and a state residual $e(i|k)$, namely

$$x(i|k) = z(i|k) + e(i|k), i \in \mathbb{N}_{\geq 0}, \tag{15}$$

where the linear combination of past states $z(i|k) \in \mathbb{R}^{n_x}$ is denoted as

$$z(i|k) = \sum_{l=1}^{N_L-i} (\theta(i|k))_l x(k-l), i \in \mathbb{N}_{[0,N-1]}, (z(i|k) = 0 \text{ for } \ i \in \mathbb{N}_{\geq N}), \tag{16}$$

with $(\theta(i|k))_l$ denoting the l^{th} element of vector $\theta(i|k)$, and the state residual evolves according to

$$e(i|k) = W(i-1|k) + (A+BK)\, e(i-1|k) + Bc(i-1|k), i \in \mathbb{N}_{\geq 1}, (c(i|k) = 0 \ \text{ for } \ \ i \in \mathbb{N}_{\geq N}), \tag{17}$$

where $e(0|k) = x(0|k) - z(0|k)$, $W(i-1|k)$ is defined as

$$W(i-1|k) \triangleq \sum_{p=2}^{N_L-i+1} \left[(\theta(i-1|k))_p - (\theta(i|k))_{p-1}\right] x(k-p+1) + (\theta(i-1|k))_1\, x(k), i \in \mathbb{N}_{[1,N-1]}, \tag{18}$$

$$W(N-1|k) \triangleq \sum_{p=1}^{N_L-N+1} (\theta(N-1|k))_p\, x(k-p+1), W(i|k) = 0, i \in \mathbb{N}_{\geq N}, \tag{19}$$

with $x(k)$ known at time k and $(\theta(i|k))_{p-1}$ denoting the $(p-1)^{th}$ element of vector $\theta(i|k)$. On the basis of (15) and (17), the predicted state evolves according to

$$x(i+1|k) = (A+BK)x(i|k) + z(i+1|k) + W(i|k) - (A+BK)z(i|k) + Bc(i|k), i \in \mathbb{N}_{[0,N-1]}. \tag{20}$$

With the terminal state feedback control strategy (11) for $i \geq N$, the state evolves as

$$x(i+1|k) = (A+BK)x(i|k), i \in \mathbb{N}_{\geq N}. \tag{21}$$

Remark 1. (Prediction horizon) The above deduction is derived assuming that the prediction horizon N is less than or equal to to the data length N_L, $N \leq N_L$. For the cases of $N > N_L$, the extension is rather straightforward by respectively evolving the state according to (20) from the current time k to the future time $k + N_L - 1$ where $z(N_L|k) = 0$ and $x(i+1|k) = (A+BK)x(i|k) + Bc(i|k), i \in \mathbb{N}_{[N_L,N-1]}$.

By using the proposed IMD technique, the future state in (15) consist of a known component $z(i|k)$ and an unknown residual signal $e(i|k)$. As the true model parameter $[A, B]$ satisfying Assumption 1 is unknown, the future residual signal $e(i|k)$ in (17) is unknown but bounded. To ensure robust constraint satisfaction, we use RMPC method to construct state sets for $x(i|k)$ in (15). Compared to RMPC which uses polytopic model vertices $\left[A^{(j)}, B^{(j)}\right]$ to predict the whole state $x(i|k)$, we use the knowledge of past data to separate a known part $z(i|k)$ from the future unknown state $x(i|k)$, enabling the conservatism to be reduced.

3.2 | Data-driven state set and constraint satisfaction

In this subsection, we consider constructing state sets to bound the future predicted state in (20). In particular, we construct a sequence of state sets $\{\mathcal{X}(0|k), \mathcal{X}(1|k), \cdots, \mathcal{X}(N|k)\}$ which contain all the possible future predicted state $x(i|k)$ with the form of

$$x(i|k) \in \mathcal{X}(i|k) := \{x(i|k) | x(i|k) = z(i|k) + e(i|k)\}, i \in \mathbb{N}_{[0,N]}. \tag{22}$$

For the state residual $e(i|k)$, we construct a sequence of sets for the future predicted state residual $\{\mathcal{E}(0|k), \mathcal{E}(1|k), \cdots, \mathcal{E}(N|k)\}$ with the form of

$$e(0|k) \in \mathcal{E}(0|k) = \{e(0|k), e(0|k) = x(0|k) - z(0|k)\}, e(i|k) \in \mathcal{E}(i|k) := Co\{e^{\tilde{j}_{1:i}}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}, i \in \mathbb{N}_{[1,N]}, \tag{23}$$

with $x(0|k) = x(k)$ known at time k and $z(0|k)$ defined in (16). The set $\mathcal{E}(i|k)$ is constituted by a convex hull of vertices $e^{\tilde{j}_{1:i}}(i|k) \in \mathbb{R}^{n_x}$. The state residual vertex evolves according to

$$e^{(\tilde{j}_1)}(1|k) = W(0|k) + \phi^{(\tilde{j}_1)} e(0|k) + B^{(\tilde{j}_1)} c(0|k), \tilde{j}_1 \in \mathcal{L}_1, \tag{24}$$

with $\phi^{(\tilde{j}_1)} = A^{(\tilde{j}_1)} + B^{(\tilde{j}_1)} K$ and $W(0|k)$ defined in (18), $\left[A^{(\tilde{j}_1)}, B^{(\tilde{j}_1)}\right]$ and K defined in Assumption 1 and (11), respectively. For $i \in \mathbb{N}_{[2,N]}, \tilde{j}_{1:i-1} \in \ \mathcal{L}_{i-1}, j_i \in \mathbb{N}_{[1,L]}$,

$$e^{(\tilde{j}_{1:i-1}, j_i)}(i|k) = W(i-1|k) + \phi^{(j_i)} e^{\tilde{j}_{1:i-1}}(i-1|k) + B^{(j_i)} c(i-1|k), \tag{25}$$

where $\phi^{(j_i)} = A^{(j_i)} + B^{(j_i)} K$ and $W(i-1|k)$ is defined in (18)-(19).

With (23)-(25), the future state set in (22) can be rewritten as

$$x(0|k) \in \mathcal{X}(0|k) := \{x(0|k)\}, x(i|k) \in \mathcal{X}(i|k) := Co\{x^{\tilde{j}_{1:i}}(i|k), x^{\tilde{j}_{1:i}}(i|k) = z(i|k) + e^{\tilde{j}_{1:i}}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}, i \in \mathbb{N}_{[1,N]}, \tag{26}$$

which is constituted by a convex hull of vertices $x^{\tilde{j}_{1:i}}(i|k) \in \mathbb{R}^{n_x}$. The current state $x(0|k) = x(k)$ and the linear combination of past states $z(i|k)$ is defined in (16).

For the construction of the set in which the future input stays, we construct a sequence of input sets $\{\mathcal{U}(0|k), \mathcal{U}(1|k), \cdots, \mathcal{U}(N-1|k)\}$ with the form of

$$u(i|k) \in \mathcal{U}(i|k) := \{u(i|k)|u(i|k) = v(i|k) + \mu(i|k)\}, i \in \mathbb{N}_{[0,N-1]}. \tag{27}$$

For the input residual, we construct a sequence of sets $\{\Pi(0|k), \Pi(1|k), \cdots, \Pi(N-1|k)\}$ which contain all the possible future predicted input residual with the form of

$$\mu(0|k) \in \Pi(0|k) := \{\mu(0|k)\}, \mu(i|k) \in \Pi(i|k) := Co\{\mu^{\tilde{j}_{1:i}}(i|k), \mu^{\tilde{j}_{1:i}}(i|k) = Ke^{\tilde{j}_{1:i}}(i|k) + c(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}, i \in \mathbb{N}_{[1,N-1]}, \tag{28}$$

which is constituted by a convex hull of vertices $\mu^{\tilde{j}_{1:i}}(i|k) \in \mathbb{R}^{n_u}$. The current input residual $\mu(0|k)$ is defined in (14) and the vertex of state residual set $e^{\tilde{j}_{1:i}}(i|k)$ is defined in (24)-(25).

With (28), the future input set that contains all the possible future input in (27) is rewritten as

$$u(0|k) \in \mathcal{U}(0|k) := \{u(0|k)\}, u(i|k) \in \mathcal{U}(i|k) := Co\{u^{\tilde{j}_{1:i}}(i|k), u^{\tilde{j}_{1:i}}(i|k) = v(i|k) + \mu^{\tilde{j}_{1:i}}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}, i \in \mathbb{N}_{[1,N-1]}, \tag{29}$$

which is constituted by a convex hull of vertices $u^{\tilde{j}_{1:i}}(i|k) \in \mathbb{R}^{n_u}$. The current input $u(0|k)$ is defined in (11) and the linear combination of past inputs $v(i|k)$ is defined in (13).

The following lemma provides the sufficient conditions to ensure robust constraint satisfaction from the current time k to future predicted time $k + N - 1$.

Lemma 1. (Robust constraint satisfaction) For system (1) with Assumption 1, the constraint (2) is robustly ensured from the current time k to future time $k+N-1$ if there exist the state and control set sequence $\{\mathcal{X}(0|k), \cdots, \mathcal{X}(N-1|k)\}$, $\{\mathcal{U}(0|k), \cdots, \mathcal{U}(N-1|k)\}$ whose vertices satisfy

$$Fx(0|k) + Gu(0|k) \leq \mathbf{1}, Fx^{\tilde{j}_{1:i}}(i|k) + Gu^{\tilde{j}_{1:i}}(i|k) \leq \mathbf{1}, i \in \mathbb{N}_{[1,N-1]}, \forall \tilde{j}_{1:i} \in \mathcal{L}_i. \tag{30}$$

Proof. This follows from the definition of future state and input set respectively in (26) and (29). □

To guarantee robust constraint satisfaction from $k+N$ thereafter, we need to ensure that the terminal state enters into a robust positively invariant (RPI) set. Given the polytopic model Ξ and a robustly stabilizing feedback gain K in (11), we can use the algorithm from [24] to obtain a polyhedral RPI set $\Omega = \{x|Tx \leq \mathbf{1}\}$ with $T \in \mathbb{R}^{n_T \times n_x}$ for the system (1) satisfying Assumption 1 with constraint (2) and control law $u = Kx$. The following lemma gives sufficient conditions to ensure that all the possible future terminal states are constrained in the terminal RPI set.

Lemma 2. (Terminal set) If the vertices of terminal state set satisfy the following condition, then the constraint in (2) can be ensured from the future time $k + N$ to infinity for the system (1) with Assumption 1,

$$Tx^{\tilde{j}_{1:N}}(N|k) \leq \mathbf{1}, \forall \tilde{j}_{1:N} \in \mathcal{L}_N. \tag{31}$$

Proof. This can be derived by using (26) with $i = N$ and the definition of RPI set introduced in [1]. □

3.3 | Data-driven cost

In this work, we consider an infinite-horizon quadratic cost function

$$J(k) = \sum_{i=0}^{\infty}(x^T(i|k)Qx(i|k) + u^T(i|k)Ru(i|k)). \tag{32}$$

As the system is unknown and true predicted state is only known to lie within the state sets $\mathcal{X}(i|k)$, we employ the upper bound of $J(k)$ as the objective function which can be expressed as a quadratic form of the decision variables.

To define the objective function, we introduce some notations that will be used in the following derivation. Let

$$\underline{z}(i|k) = \texttt{col}\,[z(i|k), z(i+1|k), \cdots, z(i+N-1|k)]\,, \underline{v}(i|k) = \texttt{col}\,[v(i|k), v(i+1|k), \cdots, v(i+N-1|k)]\,, \tag{33}$$

$$\underline{c}(i|k) = \texttt{col}\,[c(i|k), c(i+1|k), \cdots, c(i+N-1|k)]\,, \underline{W}(i|k) = \texttt{col}\,[W(i|k), W(i+1|k), \cdots, W(i+N-1|k)]\,, \tag{34}$$

$$\varsigma(i|k) = \texttt{col}\left[\underline{z}(i|k), \underline{v}(i|k), e(i|k), \underline{c}(i|k), \underline{W}(i|k)\right], \tag{35}$$

where $i \in \mathbb{N}_{\geq 0}$ and $e(i|k)$ is defined in (17). Define the matrices $E_{n_x} \in \mathbb{R}^{n_x \times N n_x}$, $E_{n_u} \in \mathbb{R}^{n_u \times N n_u}$ with

$$E_{n_x} := \left[\, I_{n_x} \;\; \mathbf{0} \; \cdots \; \mathbf{0} \,\right], E_{n_u} := \left[\, I_{n_u} \;\; \mathbf{0} \; \cdots \; \mathbf{0} \,\right], \tag{36}$$

such that $z(i|k) = E_{n_x}\underline{z}(i|k), v(i|k) = E_{n_u}\underline{v}(i|k), c(i|k) = E_{n_u}\underline{c}(i|k), W(i|k) = E_{n_x}\underline{W}(i|k)$, and shift matrices $T_{n_x} \in \mathbb{R}^{N n_x \times N n_x}, T_{n_u} \in \mathbb{R}^{N n_u \times N n_u}$ defined as

$$T_{n_x} = \begin{bmatrix} \mathbf{0} & I_{n_x} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & I_{n_x} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & I_{n_x} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix}, T_{n_u} = \begin{bmatrix} \mathbf{0} & I_{n_u} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & I_{n_u} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & I_{n_u} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix}, \tag{37}$$

such that

$$\underline{z}(i+1|k) = T_{n_x}\underline{z}(i|k), \underline{W}(i+1|k) = T_{n_x}\underline{W}(i|k), \underline{v}(i+1|k) = T_{n_u}\underline{v}(i|k), \underline{c}(i+1|k) = T_{n_u}\underline{c}(i|k). \tag{38}$$

The following lemma will derive this data-driven cost upper bound.

Lemma 3. (Cost upper bound) Define a new cost function $\bar{J}(k) = \xi(k)^T M(k)\xi(k)$, where $\xi(k) = \mathtt{col}\left[\underline{\theta}(k), 1, \underline{c}(0|k)\right] \in \mathbb{R}^{N\times(N_L+n_u)-\frac{N(N-1)}{2}+1}$, $\underline{c}(0|k)$ is defined in (34) and $\underline{\theta}(k) = \mathtt{col}\left[\theta(0|k), \theta(1|k), \cdots, \theta(N-1|k)\right]$ with $\theta(i|k), i \in \mathbb{N}_{[0,N-1]}$ mentioned in (13), $M(k)$ is a positive definite matrix with

$$M(k) = D(k)^T P D(k), \tag{39}$$

where $D(k) \in \mathbb{R}^{[2N\cdot(n_x+n_u)+n_x]\times[N\times(N_L+n_u)-\frac{N(N-1)}{2}+1]}$ is defined as

$$D(k) = \left[\begin{array}{cccccccc}
(X_k^-)_{1:N_L} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & (X_k^+)_{2:N_L} & \mathbf{0} & \ddots & \vdots & \vdots & \vdots & \vdots \\
\vdots & \mathbf{0} & (X_k^+)_{2:N_L-1} & \ddots & \mathbf{0} & \vdots & \vdots & \vdots \\
\vdots & \vdots & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} & \vdots & \vdots \\
\vdots & & \vdots & \ddots & (X_k^+)_{2:(N_L-N+3)} & \mathbf{0} & \mathbf{0} & \vdots \\
\mathbf{0} & \mathbf{0} & & \vdots & \mathbf{0} & (X_k^+)_{2:(N_L-N+2)} & \mathbf{0} & \mathbf{0} \\
\hdashline
(U_k^-)_{1:N_L} & \mathbf{0} & & \vdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & (U_k^-)_{1:N_L-1} & \mathbf{0} & & \vdots & \mathbf{0} & \mathbf{0} & \mathbf{0} \\
\vdots & \mathbf{0} & (U_k^-)_{1:N_L-2} & \ddots & & \vdots & \mathbf{0} & \mathbf{0} \\
\vdots & & \mathbf{0} & \ddots & \mathbf{0} & & \vdots & \mathbf{0} \\
\vdots & \vdots & & \ddots & (U_k^-)_{1:(N_L-N+2)} & \mathbf{0} & & \vdots \\
\mathbf{0} & & \vdots & & \mathbf{0} & (U_k^-)_{1:(N_L-N+1)} & \mathbf{0} & \vdots \\
-(X_k^-)_{1:N_L} & \mathbf{0} & & \vdots & & \mathbf{0} & x(k) & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & & \vdots & & \mathbf{0} & I_{n_u N} \\
\hdashline
(X_k^+)_{1:N_L} & -(X_k^+)_{2:N_L} & \mathbf{0} & \ddots & & \vdots & & \mathbf{0} \\
\mathbf{0} & (X_k^+)_{1:N_L-1} & -(X_k^+)_{2:N_L-1} & \ddots & \mathbf{0} & & \vdots & \vdots \\
\vdots & \mathbf{0} & (X_k^+)_{1:N_L-2} & \ddots & \mathbf{0} & \mathbf{0} & & \vdots \\
\vdots & \vdots & \mathbf{0} & \ddots & -(X_k^+)_{2:(N_L-N+3)} & \mathbf{0} & \mathbf{0} & \vdots \\
\vdots & \vdots & \vdots & \ddots & (X_k^+)_{1:(N_L-N+2)} & -(X_k^+)_{2:(N_L-N+2)} & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & (X_k^+)_{1:(N_L-N+1)} & \mathbf{0} & \mathbf{0}
\end{array}\right], \tag{40}$$

with $(X_k^-)_{1:i}, (U_k^-)_{1:i}, (X_k^+)_{1:i}$ denoting the 1^{th} to i^{th} columns of the matrices X_k^-, U_k^-, X_k^+ respectively defined in (5)-(7) for $i \in \mathbb{N}_{\geq 1}$, then $\bar{J}(k) \geq J(k)$ defined in (32) if the Lyapunov matrix P satisfies

$$P \geq (\psi^{(j)})^T P \psi^{(j)} + \bar{Q}, \forall j \in \mathbb{N}_{[1,L]}, \tag{41}$$

where for $j \in \mathbb{N}_{[1,L]}$,

$$\psi^{(j)} = \left[\begin{array}{ccc:cc:cc} T_{n_x} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hdashline \mathbf{0} & T_{n_u} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hdashline \mathbf{0} & \mathbf{0} & \phi^{(j)} & B^{(j)} & \mathbf{0} & E_{n_x} & \mathbf{0} \\ \hdashline \mathbf{0} & \mathbf{0} & \mathbf{0} & \multicolumn{2}{c:}{T_{n_u}} & \mathbf{0} & \mathbf{0} \\ \hdashline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \multicolumn{2}{c}{T_{n_x}} \end{array}\right], \quad \bar{Q} = \begin{bmatrix} E_{n_x} & \mathbf{0} & I_{n_x} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & E_{n_u} & K & E_{n_u} & \mathbf{0} \end{bmatrix}^T \begin{bmatrix} Q & \mathbf{0} \\ \mathbf{0} & R \end{bmatrix} \begin{bmatrix} E_{n_x} & \mathbf{0} & I_{n_x} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & E_{n_u} & K & E_{n_u} & \mathbf{0} \end{bmatrix}, \tag{42}$$

with $\phi^{(j)}$ defined in (12), E_{n_x}, E_{n_u} and T_{n_x}, T_{n_u} respectively defined in (36) and (37).

Proof. Define $\psi = \sum_{j=1}^{L} p^{(j)}\psi^{(j)}$ with $p^{(j)}$ denoting the convex combination parameter for Assumption 1 such that the true unknown model $[A, B] = \sum_{j=1}^{L} p^{(j)} \left[A^{(j)}, B^{(j)}\right], \sum_{j=1}^{L} p^{(j)} = 1, p^{(j)} \geq 0$. It is readily known from (17) and (38) that $\varsigma(i+1|k) = \psi\varsigma(i|k)$ where $\varsigma(i|k)$ is defined in (35). Let $V(i|k) = \varsigma(i|k)^T P\varsigma(i|k)$. The inequality (41) implies $P \geq \psi^T P\psi + \bar{Q}$, by pre- and post-multiplication of this inequality by $(\varsigma(i|k))^T$ and its transpose and considering (11), (14)-(15), we have $V(i|k) - V(i+1|k) \geq \|x(i|k)\|_Q^2 + \|u(i|k)\|_R^2$. Summing this inequality for $i = 0, 1, \cdots, \infty$ yields the result that $V(0|k) \geq J(k)$. Since (13), (15)-(16) and (18)-(19) hold, $\varsigma(0|k) = D(k)\xi(k)$. Hence, $\bar{J}(k) = V(0|k) \geq J(k)$ holds by considering (39). □

3.4 | Overall algorithm

The multi-step input-mapping data-driven RMPC method is formulated as a linearly constrained quadratic program as below,

$$\min_{\{c(i|k),\theta(i|k)\}_{i\in\mathbb{N}_{[0,N-1]}}} \quad \xi(k)^T M(k)\xi(k), s.t.\ (23)-(26), (28)-(31). \tag{43}$$

Algorithm 1 lists the off-line and online procedures for the input-mapping data-driven RMPC scheme. Figure 1 shows the block diagram of the proposed IMD-RMPC controller. The feasibility and stability of Algorithm 1 are shown in Theorem 1.

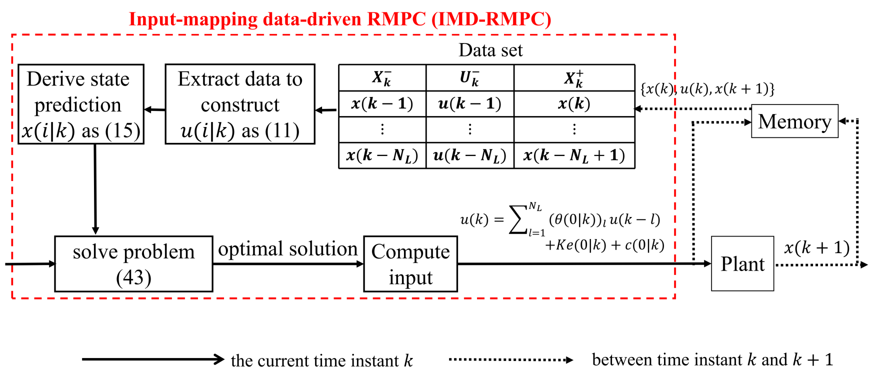


FIGURE 1 Block diagram of the IMD-RMPC controller

Algorithm 1 Input-mapping data-driven robust model predictive control (IMD-RMPC) algorithm

---

Offline:

Step 1. Determine the stabilizing feedback gain K, the polyhedron RPI set Ω and Lyapunov matrix P satisfying Lemma 3.

Online:

Step 1. Compute matrix $M(k)$ and solve problem (43). Apply input $u(k) = \left[\sum_{l=1}^{N_L} (\theta(0|k))_l u(k-l) + Ke(0|k) + c(0|k)\right]$ to system.

Step 2. Update data set X_k^-, U_k^-, X_k^+ in (5)-(7) by storing a new record $\{x(k), u(k), x(k+1)\}$, set $k = k+1$ and back to Step 1.

Theorem 1. (Feasibility and stability) For system (1) satisfying Assumption 1 subject to constraint (2), Algorithm 1 is asymptotically stable and recursively admits a feasible solution if (43) is initially feasible.

Proof. Suppose problem (43) is feasible at time k. Let $\mathbf{c}^*(k) = \{c^*(0|k), \cdots, c^*(N-1|k)\}$ ($c^*(i|k) = \mathbf{0}$ for $i \in \mathbb{N}_{\geq N}$), $\boldsymbol{\theta}^*(k) = \{\theta^*(0|k), \cdots, \theta^*(N-1|k)\}$ be the optimal solution at time k. Denote $\mathbf{z}^*(k) = \{z^*(0|k), \cdots, z^*(N-1|k)\}$ ($z^*(i|k) = \mathbf{0}$ for $i \in \mathbb{N}_{\geq N}$), $\boldsymbol{\nu}^*(k) = \{\nu^*(0|k), \cdots, \nu^*(N-1|k)\}$ ($\nu^*(i|k) = \mathbf{0}$ for $i \in \mathbb{N}_{\geq N}$), as the optimal linear combination of past states and inputs, $\mathbf{W}^*(k) = \{W^*(0|k), \cdots, W^*(N-1|k)\}$ ($W^*(i|k) = \mathbf{0}$ for $i \in \mathbb{N}_{\geq N}$) as the optimal auxiliary variable defined in (18)-(19) at time k.

Define the optimal set of state residual $\mathcal{E}^*(i|k)$ and state $\mathcal{X}^*(i|k)$ at time k satisfying (23)-(26) as follows,

$$e^*(i|k) \in \mathcal{E}^*(i|k) = Co\{e^{(\tilde{j}_{1:i})*}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}, x^*(i|k) \in \mathcal{X}^*(i|k) = Co\{x^{(\tilde{j}_{1:i})*}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}, i \in \mathbb{N}_{[1,N]}, \tag{44}$$

with $\mathcal{E}^*(0|k) = \{e^*(0|k)\}, \mathcal{X}^*(0|k) = \{x^*(0|k)\}$ for $i = 0$. Denote the optimal set of input residual and input at time k by $\Pi^*(i|k), \mathcal{U}^*(i|k)$ which are respectively defined as

$$\mu^*(i|k) \in \Pi^*(i|k) = Co\{\mu^{(\tilde{j}_{1:i})*}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}, u^*(i|k) \in \mathcal{U}^*(i|k) = Co\{u^{(\tilde{j}_{1:i})*}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}, i \in \mathbb{N}_{[0,N-1]}, \tag{45}$$

with $\Pi^*(0|k) = \{\mu^*(0|k)\}, \mathcal{U}^*(0|k) = \{u^*(0|k)\}$ for $i = 0$. The optimal sets $\Pi^*(i|k), \mathcal{U}^*(i|k)$ satisfy (28)-(29).

Recursive feasibility Firstly, we define the candidate perturbation and linear combination parameter sequence at time $k+1$ as

$$\hat{\mathbf{c}}(k+1) = \{c^*(1|k), c^*(2|k), \cdots, c^*(N-1|k), \mathbf{0}\}, \hat{c}(i|k+1) = \mathbf{0}, i \in \mathbb{N}_{\geq N}, \tag{46}$$

$$\hat{\boldsymbol{\theta}}(k+1) = \{\hat{\theta}(0|k+1), \hat{\theta}(1|k+1), \cdots, \hat{\theta}(N-2|k+1), \mathbf{0}\}, \tag{47}$$

where for $i \in \mathbb{N}_{[0,N-2]}$, $\hat{\theta}(i|k+1) = \texttt{col}\left[0_{1\times 1}, \theta^*(i+1|k)\right] \in \mathbb{R}^{N_L - i}$ so that (13), (16) and (18)-(19) respectively give

$$\hat{z}(i|k+1) = z^*(i+1|k), i \in \mathbb{N}_{[0,N-1]}, \hat{z}(i|k+1) = \mathbf{0}, i \in \mathbb{N}_{\geq N}, \tag{48}$$

$$\hat{\nu}(i|k+1) = \nu^*(i+1|k), i \in \mathbb{N}_{[0,N-1]}, \hat{\nu}(i|k+1) = \mathbf{0}, i \in \mathbb{N}_{\geq N}, \tag{49}$$

$$\hat{W}(i|k+1) = W^*(i+1|k), i \in \mathbb{N}_{[0,N-1]}, \hat{W}(i|k+1) = \mathbf{0}, i \in \mathbb{N}_{\geq N}. \tag{50}$$

To construct the candidate set of state residual $\hat{\mathcal{E}}(i|k+1)$ and input residual $\hat{\Pi}(i|k+1)$, implied by Assumption 1 that there exists $p^{(j)} \in \mathbb{R}_{[0,1]}, j \in \mathbb{N}_{[1,L]}$, such that

$$\textstyle\sum_{j=1}^{L} p^{(j)} = 1, [A, B] = \sum_{j=1}^{L} p^{(j)} [A^{(j)}, B^{(j)}]. \tag{51}$$

For the optimal solution defined above in (44)-(45), the superscript $\tilde{j}_{1:i} \in \mathcal{L}_i$ denotes the index of polytopic model sequence used in expressing each vertex of optimal set. For example, the notation $e^{(\tilde{j}_{1:i})*}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i$ implies that the vertex of state residual optimal set is generated by using model sequence of $[A^{(j_1)}, B^{(j_1)}], \cdots, [A^{(j_i)}, B^{(j_i)}]$ with $j_1, \cdots, j_i \in \mathbb{N}_{[1,L]}$, i.e., $[A^{(j_t)}, B^{(j_t)}]$ with $t \in \mathbb{N}_{[1,i]}$ is used at predicted time $k+t$ to compute $e^{(\tilde{j}_{1:i})*}(i|k)$ in (24)-(25). At time $k+1$, the candidate solution is constructed from the shifted optimal solution from previous time k. Hence, for the purpose of the superscript $\tilde{j}_{1:i} \in \mathcal{L}_i$ aligning with the shifted constructed solution, in the proof below, we use the superscript $(j, \tilde{j}_{1:i-1})$ (or only j) with $j \in \mathbb{N}_{[1,L]}, \tilde{j}_{1:i-1} \in \mathcal{L}_{i-1}$, denoting that vertex of optimal set is generated by model sequence $[A^{(j)}, B^{(j)}], [A^{(j_1)}, B^{(j_1)}], \cdots, [A^{(j_{i-1})}, B^{(j_{i-1})}]$, i.e., the polytopic model vertices $[A^{(j)}, B^{(j)}]$ and $[A^{(j_t)}, B^{(j_t)}]$ with $t \in \mathbb{N}_{[1,i-1]}$ are respectively used at predicted time $k+1, k+t+1$ to compute $e^{(j,\tilde{j}_{1:i-1})*}(i|k)$ in (25) (or $[A^{(j)}, B^{(j)}]$ is used at predicted time $k+1$ to compute $e^{(j)*}(1|k)$) in (24)) with $i \in \mathbb{N}_{[2,N]}$. With $(j, \tilde{j}_{1:i-1})$, (24)-(25) are recast as

$$e^{(j)*}(1|k) = W^*(0|k) + (A^{(j)} + B^{(j)}K)e^*(0|k) + B^{(j)}c^*(0|k), \tag{52}$$

$$e^{(j,\tilde{j}_1)*}(2|k) = W^*(1|k) + (A^{(\tilde{j}_1)} + B^{(\tilde{j}_1)}K)e^{(j)*}(1|k) + B^{(\tilde{j}_1)}c^*(1|k), \tag{53}$$

$$e^{(j,\tilde{j}_{1:i-1},j_i)*}(i+1|k) = W^*(i|k) + (A^{(j_i)} + B^{(j_i)}K)e^{(j,\tilde{j}_{1:i-1})*}(i|k) + B^{(j_i)}c^*(i|k), i \in \mathbb{N}_{[2,N-1]}. \tag{54}$$

Hence, we intend to use the the optimal vertex $e^{(j,\tilde{j}_{1:i-1})*}(i|k)$ to construct the candidate vertex of state residual set denoted as $\hat{e}^{\tilde{j}_{1:i-1}}(i-1|k+1)$, which will be defined next. For the vertices of optimal state and input residual sets at time k,

$$\{e^*(0|k), e^{(j)*}(1|k), e^{(j,\tilde{j}_1)*}(2|k), \cdots, e^{(j,\tilde{j}_{1:N-1})*}(N|k)\}, \{\mu^*(0|k), \mu^{(j)*}(1|k), \mu^{(j,\tilde{j}_1)*}(2|k), \cdots, \mu^{(j,\tilde{j}_{1:N-2})*}(N-1|k)\}$$

satisfying (23)-(25) and (28), the vertices of candidate sets of state residual $\hat{\mathcal{E}}(i|k+1) = Co\{\hat{e}^{\tilde{j}_{1:i}}(i|k+1), \tilde{j}_{1:i} \in \mathcal{L}_i\}$ for $i \in \mathbb{N}_{[1,N]}$ ($\hat{\mathcal{E}}(0|k+1) = \{\hat{e}(0|k+1)\}$) and input residual $\hat{\Pi}(i|k+1) = Co\{\hat{\mu}^{\tilde{j}_{1:i}}(i|k+1), \tilde{j}_{1:i} \in \mathcal{L}_i\}$ for $i \in \mathbb{N}_{[1,N-1]}$

($\hat{\Pi}(0|k+1) = \{\hat{\mu}(0|k+1)\}$) defined by

$$\hat{e}(0|k+1) = e^*(1|k) = \textstyle\sum_{j=1}^{L} p^{(j)} e^{(j)*}(1|k), \hat{\mu}(0|k+1) = \mu^*(1|k) = \sum_{j=1}^{L} p^{(j)} \mu^{(j)*}(1|k), \tag{55}$$

$$\hat{e}^{\tilde{j}_{1:i}}(i|k+1) = \textstyle\sum_{j=1}^{L} p^{(j)} e^{(j,\tilde{j}_{1:i})*}(i+1|k), i \in \mathbb{N}_{[1,N-1]}, \hat{\mu}^{\tilde{j}_{1:i}}(i|k+1) = \sum_{j=1}^{L} p^{(j)} \mu^{(j,\tilde{j}_{1:i})*}(i+1|k), i \in \mathbb{N}_{[1,N-2]}, \tag{56}$$

$$\hat{e}^{(\tilde{j}_{1:N-1},j_N)}(N|k+1) = (A^{(j_N)} + B^{(j_N)}K)\hat{e}^{\tilde{j}_{1:N-1}}(N-1|k+1), \hat{\mu}^{\tilde{j}_{1:N-1}}(N-1|k+1) = K\hat{e}^{\tilde{j}_{1:N-1}}(N-1|k+1), \tag{57}$$

for all $\tilde{j}_{1:N-1} \in \mathcal{L}_{N-1}, j_N \in \mathbb{N}_{[1,L]}$, satisfy, at time $k+1$, the constraints (23)-(25) and (28) by considering (46) and (50). Analogously, for the vertices of optimal state and input sets at time k

$$\{x^*(0|k), x^{(j)*}(1|k), x^{(j,\tilde{j}_1)*}(2|k), \cdots, x^{(j,\tilde{j}_{1:N-1})*}(N|k)\}, \{u^*(0|k), u^{(j)*}(1|k), u^{(j,\tilde{j}_1)*}(2|k), \cdots, u^{(j,\tilde{j}_{1:N-2})*}(N-1|k)\}$$

satisfying (26) and (29), the vertices of candidate sets of state residual $\hat{\mathcal{X}}(i|k+1) = Co\{\hat{x}^{\tilde{j}_{1:i}}(i|k+1), \tilde{j}_{1:i} \in \mathcal{L}_i\}$ for $i \in \mathbb{N}_{[1,N]}$ ($\hat{\mathcal{X}}(0|k+1) = \{\hat{x}(0|k+1)\}$) and input residual $\hat{u}(i|k+1) = Co\{\hat{u}^{\tilde{j}_{1:i}}(i|k+1), \tilde{j}_{1:i} \in \mathcal{L}_i\}$ for $i \in \mathbb{N}_{[1,N-1]}$ ($\hat{\mathcal{U}}(0|k+1) = \{\hat{u}(0|k+1)\}$) defined by

$$\hat{x}(0|k+1) = \textstyle\sum_{j=1}^{L} p^{(j)} x^{(j)*}(1|k), \hat{u}(0|k+1) = \sum_{j=1}^{L} p^{(j)} u^{(j)*}(1|k), \tag{58}$$

$$\hat{x}^{\tilde{j}_{1:i}}(i|k+1) = \textstyle\sum_{j=1}^{L} p^{(j)} x^{(j,\tilde{j}_{1:i})*}(i+1|k), i \in \mathbb{N}_{[1,N-1]}, \hat{u}^{\tilde{j}_{1:i}}(i|k+1) = \sum_{j=1}^{L} p^{(j)} u^{(j,\tilde{j}_{1:i})*}(i+1|k), i \in \mathbb{N}_{[1,N-2]}, \tag{59}$$

$$\hat{x}^{(\tilde{j}_{1:N-1},j_N)}(N|k+1) = (A^{(j_N)} + B^{(j_N)}K)\hat{x}^{\tilde{j}_{1:N-1}}(N-1|k+1), \hat{u}^{\tilde{j}_{1:N-1}}(N-1|k+1) = K\hat{x}^{\tilde{j}_{1:N-1}}(N-1|k+1), \tag{60}$$

with $j_N \in \mathbb{N}_{[1,L]}, \tilde{j}_{1:N-1} \in \mathcal{L}_{N-1}$, satisfy, at time $k+1$, the constraints (26) and (29) by considering (48)-(49) and (55)-(57). Therefore, given

$$\{x^*(0|k), x^{(j)*}(1|k), x^{(j,\tilde{j}_1)*}(2|k), \cdots, x^{(j,\tilde{j}_{1:N-1})*}(N|k)\}, \{u^*(0|k), u^{(j)*}(1|k), u^{(j,\tilde{j}_1)*}(2|k), \cdots, u^{(j,\tilde{j}_{1:N-2})*}(N-1|k)\}$$

satisfying (30)-(31), the constraint (30) holds at time $k+1$ by convexity. The constraint (31) holds as $\hat{x}^{(j,\tilde{j}_{1:N-2})}(N-1|k+1) \in \Omega$ caused by (59) and $x^{(j,\tilde{j}_{1:N-1})*}(N|k) \in \Omega$, and Ω is a RPI set for the system (1) subject to the constraint (2) with $u = Kx$. Thus the candidate solution $\{\hat{\mathbf{c}}(\mathbf{k+1}), \hat{\boldsymbol{\theta}}(\mathbf{k+1})\}$ is feasible at time $k+1$.

Closed-loop stability The closed-loop stability is proven by showing monotonic decrease of the optimal cost, denote the optimal cost of problem (43) at time k and cost of the constructed feasible solution at time $k+1$ respectively as

$$V(\xi^*(k)) \triangleq (\xi^*(k))^T M(k)\xi^*(k), V(\hat{\xi}(k+1)) \triangleq (\hat{\xi}(k+1))^T M(k+1)\hat{\xi}(k+1), \tag{61}$$

where

$$\xi^*(k) = \texttt{col}\left[\underline{\theta}^*(k), 1, \underline{c}^*(k)\right], \hat{\xi}(k+1) = \texttt{col}\left[\hat{\underline{\theta}}(k+1), 1, \hat{\underline{c}}(k+1)\right], \tag{62}$$

$$\underline{\theta}^*(k) = \boldsymbol{\theta}^*(k), \hat{\underline{\theta}}(k+1) = \hat{\boldsymbol{\theta}}(k+1), \underline{c}^*(0|k) = \mathbf{c}^*(k), \hat{\underline{c}}(0|k+1) = \hat{\mathbf{c}}(k+1), \tag{63}$$

with $\mathbf{c}^*(k) = \{c^*(0|k), \cdots, c^*(N-1|k)\}, \boldsymbol{\theta}^*(k) = \{\theta^*(0|k), \cdots, \theta^*(N-1|k)\}$ and $\hat{\boldsymbol{\theta}}(k+1), \hat{\mathbf{c}}(k+1)$ respectively defined in (46)-(47). Denote the optimal augment vector defined in (35) at time k as

$$\varsigma^*(i|k) = \texttt{col}\left[\underline{z}^*(i|k), \underline{v}^*(i|k), e^*(i|k), \underline{c}^*(i|k), \underline{W}^*(i|k)\right], i \in \mathbb{N}_{\geq 0}, \tag{64}$$

with $e^*(i|k) \in Co\{\hat{e}^{(\tilde{j}_{1:i})*}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}$, $\underline{z}^*(i|k) = \texttt{col}[z^*(i|k), z^*(i+1|k), \cdots, z^*(i+N-1|k)]$, $\underline{v}^*(i|k) = \texttt{col}[v^*(i|k), v^*(i+1|k), \cdots, v^*(i+N-1|k)]$, $\underline{c}^*(i|k) = \texttt{col}[c^*(i|k), \cdots, c^*(i+N-1|k)]$, $\underline{W}^*(i|k) = \texttt{col}[W^*(i|k), W^*(i+1|k), \cdots, W^*(i+N-1|k)]$ and define the constructed solution at time $k+1$ as

$$\hat{\varsigma}(i|k+1) = \texttt{col}\left[\hat{\underline{z}}(i|k+1), \hat{\underline{v}}(i|k+1), \hat{e}(i|k+1), \hat{\underline{c}}(i|k+1), \hat{\underline{W}}(i|k+1)\right], i \in \mathbb{N}_{\geq 0}, \tag{65}$$

with $\hat{e}(0|k+1)$ and $\hat{e}(i|k+1) \in Co\{\hat{e}^{\tilde{j}_{1:i}}(i|k+1), \tilde{j}_{1:i} \in \mathcal{L}_i\}$ for $i \in \mathbb{N}_{\geq 1}$ respectively defined in (55)-(57),

$$\hat{\underline{z}}(i|k+1) = \texttt{col}\left[\hat{z}(i|k+1), \cdots, \hat{z}(i+N-1|k+1)\right], \hat{\underline{W}}(i|k+1) = \texttt{col}\left[\hat{W}(i|k+1), \cdots, \hat{W}(i+N-1|k+1)\right], \tag{66}$$

$$\hat{\underline{v}}(i|k+1) = \texttt{col}\left[\hat{v}(i|k+1), \cdots, \hat{v}(i+N-1|k+1)\right], \hat{\underline{c}}(i|k+1) = \texttt{col}\left[\hat{c}(i|k+1), \cdots, \hat{c}(i+N-1|k+1)\right]. \tag{67}$$

As indicated by (13), (15)-(16) and (18)-(19), $\varsigma^*(0|k) = D(k)\xi^*(k)$ and $\hat{\varsigma}(0|k+1) = D(k+1)\hat{\xi}(k+1)$ hold where $\xi^*(k)$ and $\hat{\xi}(k+1)$ are defined in (62). Hence, considering (39) and (61),

$$V(\xi^*(k)) = (\varsigma^*(0|k))^T P\varsigma^*(0|k), V(\hat{\xi}(k+1)) = (\hat{\varsigma}(0|k+1))^T P\hat{\varsigma}(k+1). \tag{68}$$

According to (17), (38) and implied by (46), (48)-(50), (55), (64)-(67),

$$\hat{\varsigma}(0|k+1) = \varsigma^*(1|k) = \psi\varsigma^*(0|k), \tag{69}$$

with $\psi = \sum_{j=1}^{L} p^{(j)}\psi^{(j)}, \psi^{(j)}$ defined in (42) and $p^{(j)}$ satisfying (51). Therefore, since (41) implying $P \geq \psi^T P\psi + \bar{Q}$, by pre- and post-multiplication of this inequality by $(\varsigma^*(0|k))^T$ and its transpose and considering (68)-(69), (11) and (14)-(15), we have $V(\xi^*(k)) - V(\hat{\xi}(k+1)) \geq \|x(0|k)\|_Q^2 + \|u(0|k)\|_R^2$ which implies $V(\xi^*(k)) - V(\xi^*(k+1)) \geq \|x(0|k)\|_Q^2 + \|u(0|k)\|_R^2 > 0, \forall x(0|k) \neq 0, u(0|k) \neq 0$ as Q, R are positive definite. Hence, the closed-loop system is asymptotically stable. □

4 | DATA-DRIVEN INTERPOLATION MODEL PREDICTIVE CONTROL

The proposed Algorithm 1 in Section 3 uses a a linear combination of past inputs and a fixed feedback control gain as the control policy. The feedback gain determines the size of feasible region and control performance and a choice of one feedback gain need to compromise these objectives[5]. To solve this issue, we consider the combination of the IMD scheme and interpolation robust model predictive control (IRMPC) which combines the advantages of various control laws[5,6,7].

$$\min_{Y_0,Q_0,\theta_p,S} -\theta_p\,, \tag{70}$$

$$s.t. \begin{bmatrix} 1 & \theta_p x_p^T \\ \theta_p x_p & Q_0 \end{bmatrix} \geq 0, \tag{71}$$

$$\begin{bmatrix} Q_0 & (A^{(j)}Q_0 + B^{(j)}Y_0)^T & (Q^{1/2}Q_0)^T & (R^{1/2}Y_0)^T \\ A^{(j)}Q_0 + B^{(j)}Y_0 & Q_0 & \mathbf{0} & \mathbf{0} \\ Q^{1/2}Q_0 & \mathbf{0} & \gamma I & \mathbf{0} \\ R^{1/2}Y_0 & \mathbf{0} & \mathbf{0} & \gamma I \end{bmatrix} \geq 0, j \in \mathbb{N}_{[1,L]}, \tag{72}$$

$$\begin{bmatrix} S & FQ_0 & GY_0 \\ (FQ_0)^T & Q_0 & \mathbf{0} \\ (GY_0)^T & \mathbf{0} & Q_0 \end{bmatrix} \geq 0, S_{ii} \leq 1. i \in \mathbb{N}_{[1,n_c]}. \tag{73}$$

The key idea of IRMPC lies in the interpolation between various control laws. Throughout this paper, we assume the interpolation between two different controllers. Given the cost index γ and reference point x_p, the feedback gain can be obtained by solving (70) and applying $K = Y_0 Q_0^{-1}$, which is introduced in [6]-[7]. Two feedback gains with different specifications of performance, shape and size of feasible region can be obtained from (70) by setting γ_1, γ_2 and $x_{p,1}, x_{p,2}$ of various value. Two corresponding polyhedral RPI sets Ω_1, Ω_2 with $\Omega_r = \{x | T_r x \leq \mathbf{1}\}, T_r \in \mathbb{R}^{n_{T_r} \times n_x}, r = 1, 2$ for the system (1) with the constraint (2) and feedback control gain $u = K_r x$ can be derived by using the algorithm from [24]. Without abuse of notation, the interpolation index $r = 1, 2$.

Following [6], the current state can be decomposed into the following form,

$$x(k) = x_1(0|k) + x_2(0|k), \tag{74}$$

where the variables $\{x_r(0|k)\}_{r=1,2}$ are decomposition variables and treated as decision variables. We consider the following control policy which is the superposition of two interpolated inputs $u_r(i|k) \in \mathbb{R}^{n_u}$,

$$u(i|k) = \begin{cases} \sum_{r=1}^{2} u_r(i|k), u_r(i|k) = v_r(i|k) + K_r e_r(i|k) + c_r(i|k), & i \in \mathbb{N}_{[0,N-1]}, \\ \sum_{r=1}^{2} u_r(i|k), u_r(i|k) = K_r x_r(i|k), & i \in \mathbb{N}_{\geq N}, \end{cases} \tag{75}$$

where the perturbation sequence is denoted as $\{c_r(0|k), \cdots, c_r(N-1|k)\}_{r=1,2}$, the interpolated linear combination of past input signals $v_r(i|k)$ is revised from (13) by defining two sets of combination parameters $\theta_r(i|k) \in \mathbb{R}^{N_L - i}$ as

$$v_r(i|k) = \sum_{l=1}^{N_L-i} (\theta_r(i|k))_l u(k-l), i \in \mathbb{N}_{[0,N-1]}, (v_r(i|k) = 0 \text{ for } i \in \mathbb{N}_{\geq N}), \tag{76}$$

the interpolated state residual is defined as

$$e_r(i|k) := x_r(i|k) - z_r(i|k), \tag{77}$$

where the interpolated linear combination of past states $z_r(i|k)$ is revised from (16) as follows

$$z_r(i|k) = \sum_{l=1}^{N_L-i} (\theta_r(i|k))_l x(k-l), i \in \mathbb{N}_{[0,N-1]}, (z_r(i|k) = 0 \text{ for } i \in \mathbb{N}_{\geq N}), \tag{78}$$

and the interpolated state residual evolves according to

$$e_r(i|k) = W_r(i-1|k) + \left(A + BK_r\right) e_r(i-1|k) + Bc_r(i-1|k), i \in \mathbb{N}_{\geq 1}, (c_r(i|k) = 0 \text{ for } i \in \mathbb{N}_{\geq N}), \tag{79}$$

where $e_r(0|k) = x_r(0|k) - z_r(0|k)$, $W_r(i-1|k)$ is defined as

$$W_r(i-1|k) \triangleq \sum_{p=2}^{N_L-i+1} \left[\left(\theta_r(i-1|k)\right)_p - \left(\theta_r(i|k)\right)_{p-1} \right] x(k-p+1) + \left(\theta_r(i-1|k)\right)_1 x(k), i \in \mathbb{N}_{[1,N-1]}, \tag{80}$$

$$W_r(N-1|k) \triangleq \sum_{p=1}^{N_L-N+1} \left(\theta_r(N-1|k)\right)_p x(k-p+1), W_r(i|k) = 0, i \in \mathbb{N}_{\geq N}. \tag{81}$$

Similar to (20), the interpolated state evolves according to

$$x_r(i+1|k) = (A+BK_r)x_r(i|k) + z_r(i+1|k) + W_r(i|k) - (A+BK_r)z_r(i|k) + Bc_r(i|k), i \in \mathbb{N}_{[0,N-1]}, r = 1,2. \tag{82}$$

With the terminal state feedback control strategy in (75), the interpolated state evolves as

$$x_r(i+1|k) = (A+BK_r)x_r(i|k), i \in \mathbb{N}_{\geq N}. \tag{83}$$

4.1 | Interpolated sate set and constraint satisfaction

In this section, we define the sets of future true and interpolated state and input.We construct a sequence of state sets $\{\mathcal{X}(0|k), \mathcal{X}(1|k), \cdots, \mathcal{X}(N|k)\}$ to depict the future predicted true state as

$$x(i|k) \in \mathcal{X}(i|k) := \{x(i|k) | x(i|k) = x_1(i|k) + x_2(i|k)\}, i \in \mathbb{N}_{[0,N]}, \tag{84}$$

For the interpolated state $x_r(i|k)$, we can construct a sequence of interpolated state sets $\{\mathcal{X}_r(0|k), \mathcal{X}_r(1|k), \cdots, \mathcal{X}_r(N|k)\}$ to contain all the possible future predicted interpolated state that is defined as follows,

$$x_r(i|k) \in \mathcal{X}_r(i|k) = \{x_r(i|k) | x_r(i|k) = z_r(i|k) + e_r(i|k)\}, i \in \mathbb{N}_{[0,N]}, r = 1,2, \tag{85}$$

where the linear combination of past states for each interpolated state $z_r(i|k)$ is defined in (78). For the interpolated state residual $e_r(i|k)$ with $r = 1,2$, we can construct a sequence of sets to depict the future predicted interpolated state residual $\{\mathcal{E}_r(0|k), \mathcal{E}_r(1|k), \cdots, \mathcal{E}_r(N|k)\}$ with the form of

$$e_r(0|k) \in \mathcal{E}_r(0|k) = \{e_r(0|k)\}, e_r(i|k) \in \mathcal{E}_r(i|k) := Co\{e_r^{\tilde{j}_{1:i}}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}, i \in \mathbb{N}_{[1,N]}, \tag{86}$$

with $e_r(0|k) = x_r(0|k) - z_r(0|k)$, $z_r(0|k)$ defined in (78). The set $\mathcal{E}_r(i|k)$ is constituted by a convex hull of vertices $e_r^{\tilde{j}_{1:i}}(i|k) \in \mathbb{R}^{n_x}$. The state residual vertex evolves according to

$$e_r^{\tilde{j}_1}(1|k) = W_r(0|k) + \left(A^{(\tilde{j}_1)} + B^{(\tilde{j}_1)}K_r\right) e_r(0|k) + B^{(\tilde{j}_1)}c_r(0|k), \tilde{j}_1 \in \mathcal{L}_1, r = 1,2, \tag{87}$$

with $W_r(0|k)$ defined in (80), $A^{(\tilde{j}_1)}$, $B^{(\tilde{j}_1)}$ coming from Assumption 1 and K_r obtained by solving (70). For $i \in \mathbb{N}_{[2,N]}$, $\tilde{j}_{1:i-1} \in \mathcal{L}_{i-1}$, $j_i \in \mathbb{N}_{[1,L]}$,

$$e_r^{(\tilde{j}_{1:i-1},j_i)}(i|k) = W_r(i-1|k) + \left(A^{(j_i)} + B^{(j_i)}K_r\right) e_r^{\tilde{j}_{1:i-1}}(i-1|k) + B^{(j_i)}c_r(i-1|k), r = 1,2, \tag{88}$$

where $W_r(i-1|k)$ is defined in (80)-(81).

With (86)-(88), the interpolated state set in (85) can be further rewritten as, for $r = 1,2$,

$$x_r(0|k) \in \mathcal{X}_r(0|k) := \{x_r(0|k)\}, x_r(i|k) \in \mathcal{X}_r(i|k) := Co\{x_r^{\tilde{j}_{1:i}}(i|k), x_r^{\tilde{j}_{1:i}}(i|k) = z_r(i|k) + e_r^{\tilde{j}_{1:i}}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}, i \in \mathbb{N}_{[1,N]}. \tag{89}$$

This implies the set $\mathcal{X}_r(i|k)$ is constituted by a convex hull of vertices $x_r^{\tilde{j}_{1:i}}(i|k) \in \mathbb{R}^{n_x}$. The linear combination of past states for each interpolated state $z_r(i|k)$ is defined in (78). By using (89), the set of true state in (84) can be further rewritten as

$$x(i|k) \in \mathcal{X}(i|k) := Co\{x^{\tilde{j}_{1:i}}(i|k), x^{\tilde{j}_{1:i}}(i|k) = x_1^{\tilde{j}_{1:i}}(i|k) + x_2^{\tilde{j}_{1:i}}(i|k), x_r^{\tilde{j}_{1:i}}(i|k) \in \mathcal{X}_r(i|k), r = 1,2, \tilde{j}_{1:i} \in \mathcal{L}_i\}, i \in \mathbb{N}_{[1,N]}, \tag{90}$$

where $x(0|k) \in \mathcal{X}(0|k) := \{x(0|k) | x(0|k) = x_1(0|k) + x_2(0|k)\}$.

For the construction of the future input set, we can construct a sequence of input sets $\{\mathcal{U}(0|k), \mathcal{U}(1|k), \cdots, \mathcal{U}(N-1|k)\}$ with $r = 1,2$ to depict all the possible future predicted input which is in the form of

$$u(i|k) \in \mathcal{U}(i|k) := \{u(i|k) | u(i|k) = u_1(i|k) + u_2(i|k)\}, i \in \mathbb{N}_{[0,N-1]}. \tag{91}$$

For the future interpolated input, we can construct a sequence of input sets $\{\mathcal{U}_r(0|k), \mathcal{U}_r(1|k), \cdots, \mathcal{U}_r(N-1|k)\}$ with $r = 1,2$ to depict the set of all the possible future interpolated input that is defined as follows,

$$\mathcal{U}_r(i|k) = \{u_r(i|k) | u_r(i|k) = \nu_r(i|k) + \mu_r(i|k)\}, i \in \mathbb{N}_{[0,N-1]}, r = 1,2. \tag{92}$$

For the interpolated input residual $\mu_r(i|k), r = 1, 2$, we construct a sequence of sets for the future input residual $\{\Pi_r(0|k), \Pi_r(1|k), \cdots, \Pi_r(N-1|k)\}$, constituted by a convex hull of vertices $\mu_r^{\tilde{j}_{1:i}}(i|k) \in \mathbb{R}^{n_u}$, which is in the form of

$$\mu_r(0|k) \in \Pi_r(0|k) := \{\mu_r(0|k)\}, \mu_r(i|k) \in \Pi_r(i|k) := Co\{\mu_r^{\tilde{j}_{1:i}}(i|k), \mu_r^{\tilde{j}_{1:i}}(i|k) = K_r e_r^{\tilde{j}_{1:i}}(i|k) + c_r(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}, \tag{93}$$

for $i \in \mathbb{N}_{[1,N-1]}$, $\mu_r(0|k) = K_r e_r(0|k) + c_r(0|k)$ and $e_r^{\tilde{j}_{1:i}}(i|k)$ defined in (87)-(88). As indicated by (93), the interpolated input set in (92) can be rewritten as, for $r = 1, 2$,

$$u_r(0|k) \in \mathcal{U}_r(0|k) := \{u_r(0|k)\}, u_r(i|k) \in \mathcal{U}_r(i|k) := Co\{u_r^{\tilde{j}_{1:i}}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i, u_r^{\tilde{j}_{1:i}}(i|k) = \nu_r(i|k) + \mu_r^{\tilde{j}_{1:i}}(i|k)\}, i \in \mathbb{N}_{[1,N-1]}, \tag{94}$$

which is constituted by a convex hull of vertices $u_r^{\tilde{j}_{1:i}}(i|k) \in \mathbb{R}^{n_u}$. By using (94), the true input set in (91) can be rewritten as a set constituted by a convex hull of vertices $u^{\tilde{j}_{1:i}}(i|k) \in \mathbb{R}^{n_u}$, which is in the form of

$$u(0|k) \in \mathcal{U}(0|k) := \{u(0|k) | u(0|k) = u_1(0|k) + u_2(0|k)\}, \tag{95}$$

$$u(i|k) \in \mathcal{U}(i|k) := Co\{u^{\tilde{j}_{1:i}}(i|k), u_r^{\tilde{j}_{1:i}}(i|k) = u_1^{\tilde{j}_{1:i}}(i|k) + u_2^{\tilde{j}_{1:i}}(i|k), u_r^{\tilde{j}_{1:i}}(i|k) \in \mathcal{U}_r(i|k), r = 1, 2, \tilde{j}_{1:i} \in \mathcal{L}_i\}, i \in \mathbb{N}_{[0,N-1]}. \tag{96}$$

Lemma 4. (Robust constraint satisfaction) For system (1) with Assumption 1, constraint (2) is robustly ensured from the current time k to future time $k + N - 1$ if there exist interpolated state and control set sequence $\{\mathcal{X}_r(0|k), \cdots, \mathcal{X}_r(N-1|k)\}, \{\mathcal{U}_r(0|k), \cdots, \mathcal{U}_r(N-1|k)\}$ whose vertices satisfy

$$F \sum_{r=1}^{2} x_r(0|k) + G \sum_{r=1}^{2} u_r(0|k) \leq \mathbf{1}, \tag{97}$$

$$F \sum_{r=1}^{2} x_r^{\tilde{j}_{1:i}}(i|k) + G \sum_{r=1}^{2} u_r^{\tilde{j}_{1:i}}(i|k) \leq \mathbf{1}, i \in \mathbb{N}_{[1,N-1]}, \forall \tilde{j}_{1:i} \in \mathcal{L}_i. \tag{98}$$

Proof. This lemma follows from the definition of future state and input set respectively in (90) and (95)-(96). □

The following lemma gives sufficient conditions to ensure that constraint (2) can be satisfied from the predicted time $k + N$ to infinity.

Lemma 5. (Interpolated terminal set) If the vertices of interpolated terminal state set satisfy the following conditions, then constraint (2) can be ensured from $k + N$ to infinity for the system (1) with Assumption 1,

$$T_r x_r^{\tilde{j}_{1:N}}(N|k) \leq \lambda_r, \forall \tilde{j}_{1:N} \in \mathcal{L}_N, \sum_{r=1}^{2} \lambda_r = 1, \tag{99}$$

where $\Omega_r = \{T_r x \leq \mathbf{1}\}$ are two polyhedral RPI sets for system (1) with the constraint (2) and feedback control gain $u = K_r x$.

Proof. This follows from using (75) with $i \in \mathbb{N}_{\geq N}$ and (83) and the definition of RPI set introduced in [1]. □

Remark 2. (Enlarged feasible region) The utilization of the interpolation controller (75) will yield an enlarged terminal set (convex hull of two individual terminal sets corresponding to K_1, K_2) and hence a larger region of attraction.

4.2 | Interpolation cost

According to [6], the cost defined in (32) is upper bounded by

$$J(k) \leq 2\left[\sum_{r=1}^{2} J_r(x_r(0|k), \mathbf{u}_r(k))\right] \triangleq 2\left(\sum_{r=1}^{2} J_r(k)\right), \tag{100}$$

where $J_r(x_r(0|k), \mathbf{u}_r(k)) = \sum_{i=0}^{\infty} \left[(x_r(i|k))^T Q x_r(i|k) + (u_r(i|k))^T R u_r(i|k)\right]$ with $\mathbf{u}_r(k) = \left[u_r(0|k), u_r(1|k), \cdots, u_r(\infty|k)\right]$, is denoted as $J_r(k)$ to simplify the notation.

Lemma 6. (Interpolation cost upper bound) Define the interpolated cost function $\bar{J}_r(k) = (\xi_r(k))^T M_r(k) \xi_r(k)$, where $\xi_r(k) = \mathtt{col}[\underline{\theta}_r(k), 1, \mathbf{c}_r(k)]$ with $\underline{\theta}_r(k) = \mathtt{col}\left[\theta_r(0|k), \theta_r(1|k), \cdots, \theta_r(N-1|k)\right], \theta_r(i|k)$ mentioned above (76), $\mathbf{c}_r(k) = \mathtt{col}\left[c_r(0|k), c_r(1|k), \cdots, c_r(N-1|k)\right]$, and

$$M_r(k) = D(k)^T P_r D(k), \tag{101}$$

is a positive definite matrix with $D(k)$ defined in (40), then $\bar{J}_r(k) \geq J_r(k)$ if the Lyapunov matrices $P_r, r = 1, 2$, satisfy

$$P_r \geq (\psi_r^{(j)})^T P_r \psi_r^{(j)} + \bar{Q}_r, j \in \mathbb{N}_{[1,L]}, \tag{102}$$

with $\phi_r^{(j)} = A^{(j)} + B^{(j)} K_r$, E_{n_x}, E_{n_u} and T_{n_x}, T_{n_u} respectively defined in (36) and (37),

$$\psi_r^{(j)} = \left[\begin{array}{ccc|cc|cc} T_{n_x} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & T_{n_u} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \phi_r^{(j)} & B^{(j)} & \mathbf{0} & E_{n_x} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \multicolumn{2}{c|}{T_{n_u}} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \multicolumn{2}{c}{T_{n_x}} \end{array}\right], \bar{Q}_r = \begin{bmatrix} E_{n_x} & \mathbf{0} & I_{n_x} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & E_{n_u} & K_r & E_{n_u} & \mathbf{0} \end{bmatrix}^T \begin{bmatrix} Q & \mathbf{0} \\ \mathbf{0} & R \end{bmatrix} \begin{bmatrix} E_{n_x} & \mathbf{0} & I_{n_x} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & E_{n_u} & K_r & E_{n_u} & \mathbf{0} \end{bmatrix}. \tag{103}$$

Proof. This lemma can be proven in a similar manner as Lemma 3, hence omitted. □

The input-mapping data-driven interpolation RMPC is formulated as the following linearly constrained quadratic program,

$$\begin{aligned} &\min_{\{x_r(0|k),\{c_r(i|k),\theta_r(i|k)\}_{i\in\mathbb{N}_{[0,N-1]}}\}_{r=1,2}} 2\sum_{r=1}^{2}\left[(\xi_r(k))^T M_r(k)\xi_r(k)\right] \\ &s.t. \;\; (74), (86)-(90), (93)-(99). \end{aligned} \tag{104}$$

Algorithm 2 lists off-line and online procedures for the input-mapping data-driven interpolation RMPC scheme. Figure 2 shows the block diagram of IMD-IRMPC controller. The feasibility and stability of Algorithm 2 is shown in Theorem 2.

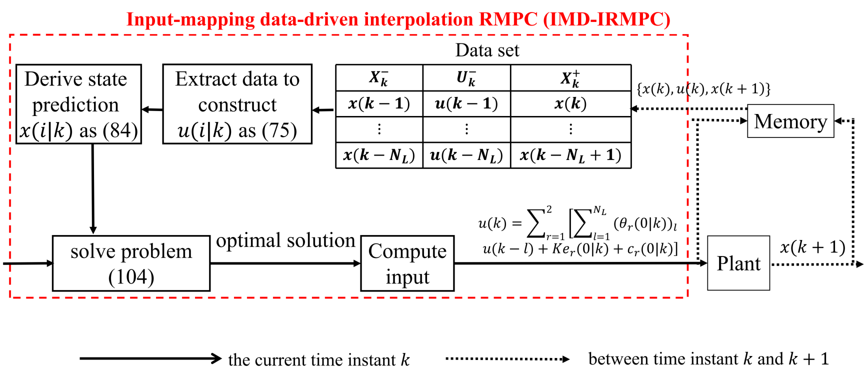


FIGURE 2 Block diagram of the IMD-IRMPC controller

Theorem 2. (Feasibility and stability) For system (1) satisfying Assumption 1 subject to constraint (2), Algorithm 2 is asymptotically stable and recursively admits a feasible solution if (104) is initially feasible.

Proof. For not abuse of notation, the interpolation number is taken as $r = 1, 2$. Suppose problem (104) is feasible at time k. Let

$$\mathbf{c}_\mathbf{r}^*(k) = \{c_r^*(0|k), \cdots, c_r^*(N-1|k)\}(c_r^*(i|k) = \mathbf{0}, i \in \mathbb{N}_{\geq N}), \boldsymbol{\theta}_\mathbf{r}^*(k) = \{\theta_r^*(0|k), \cdots, \theta_r^*(N-1|k)\} \tag{105}$$

be the optimal solution at time k. Denote $\mathbf{z}_\mathbf{r}^*(k) = \{z_r^*(0|k), \cdots, z_r^*(N|k)\}$ ($z_r^*(i|k) = \mathbf{0}$ for $i \in \mathbb{N}_{\geq N}$), $\boldsymbol{\nu}_\mathbf{r}^*(k) = \{\nu_r^*(0|k), \cdots, \nu_r^*(N-1|k)\}$ ($\nu_r^*(i|k) = \mathbf{0}$ for $i \in \mathbb{N}_{\geq N}$) as the optimal linear combination of past states and inputs, $\mathbf{W}_\mathbf{r}^*(k) = \{W_r^*(0|k), \cdots, W_r^*(N-1|k)\}$ ($W_r^*(i|k) = \mathbf{0}$ for $i \in \mathbb{N}_{\geq N}$) as the optimal auxiliary variable defined in (80)-(81) at time k. Define the optimal set of interpolated state residual $\mathcal{E}_r^*(i|k)$ and state $\mathcal{X}_r^*(i|k)$ at time k respectively satisfying (86)-(88) and (89) as follows,

$$e_r^*(i|k) \in \mathcal{E}_r^*(i|k) = Co\{e_r^{(\tilde{j}_{1:i})*}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}, x_r^*(i|k) \in \mathcal{X}_r^*(i|k) = Co\{x_r^{(\tilde{j}_{1:i})*}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}, i \in \mathbb{N}_{[1,N]}, \tag{106}$$

with $\mathcal{E}_r^*(0|k) = \{e_r^*(0|k)\}, \mathcal{X}_r^*(0|k) = \{x_r^*(0|k)\}$ for $i = 0$. Denote the optimal set of interpolated input residual and input at time k by $\Pi_r^*(i|k), \mathcal{U}_r^*(i|k)$ which is defined as

$$\mu_r^*(i|k) \in \Pi_r^*(i|k) = Co\{\mu_r^{(\tilde{j}_{1:i})*}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}, u_r^*(i|k) \in \mathcal{U}_r^*(i|k) = Co\{u_r^{(\tilde{j}_{1:i})*}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}, i \in \mathbb{N}_{[0,N-1]},$$

with $\Pi_r^*(0|k) = \{\mu_r^*(0|k)\}, \mathcal{U}_r^*(0|k) = \{u_r^*(0|k)\}$ for $i = 0$. The optimal sets $\Pi_r^*(i|k), \mathcal{U}_r^*(i|k)$ satisfy (93)-(94).

Recursive feasibility Firstly, we define the candidate perturbation and linear combination parameter sequence at time $k+1$ as

$$\hat{\mathbf{c}}_{\mathbf{r}}(k+1)=\{c_r^*(1|k), c_r^*(2|k), \cdots, c_r^*(N-1|k), \mathbf{0}\}, \hat{c}_r(i|k+1)=\mathbf{0}, i\in\mathbb{N}_{\geq N}, \tag{107}$$

$$\hat{\boldsymbol{\theta}}_{\mathbf{r}}(k+1)=\{\hat{\theta}_r(0|k+1), \hat{\theta}_r(1|k+1),\cdots, \hat{\theta}_r(N-2|k+1), \mathbf{0}\}, \tag{108}$$

where for $i\in\mathbb{N}_{[0,N-2]}$, $\hat{\theta}_r(i|k+1)=\texttt{col}\left[0_{1\times 1}, \theta_r^*(i+1|k)\right]\in\mathbb{R}^{N_L-i}$ so that (76), (78) and (80)-(81) respectively give

$$\hat{z}_r(i|k+1)=z_r^*(i+1|k), i\in\mathbb{N}_{[0,N-1]}, \hat{z}_r(i|k+1)=\mathbf{0}, i\in\mathbb{N}_{\geq N}, \tag{109}$$

$$\hat{v}_r(i|k+1)=v_r^*(i+1|k), i\in\mathbb{N}_{[0,N-1]}, \hat{v}_r(i|k+1)=\mathbf{0}, i\in\mathbb{N}_{\geq N}, \tag{110}$$

$$\hat{W}_r(i|k+1)=W_r^*(i+1|k), i\in\mathbb{N}_{[0,N-1]}, \hat{W}_r(i|k+1)=\mathbf{0}, i\in\mathbb{N}_{\geq N}. \tag{111}$$

To construct the candidate set of interpolated state residual $\hat{\mathcal{E}}_r(i|k+1)$ and input residual $\hat{\Pi}_r(i|k+1)$, we first define these two sets as follows,

$$\hat{\mathcal{E}}_r(i|k+1)=Co\{\hat{e}_r^{\tilde{j}_{1:i}}(i|k+1), \tilde{j}_{1:i}\in\mathcal{L}_i\}, i\in\mathbb{N}_{[1,N]}, \hat{\Pi}_r(i|k+1)=Co\{\hat{\mu}_r^{\tilde{j}_{1:i}}(i|k+1), \tilde{j}_{1:i}\in\mathcal{L}_i\}, i\in\mathbb{N}_{[1,N-1]}, \tag{112}$$
$$\hat{\mathcal{E}}_r(0|k+1)=\{\hat{e}_r(0|k+1)\}, \hat{\Pi}_r(0|k+1)=\{\hat{\mu}_r(0|k+1)\}.$$

Similar with the discussion after (51) in Theorem 1, we use the notation $(j,\tilde{j}_{1:i-1})$ to replace $(\tilde{j}_{1:i})$ with $j\in\mathbb{N}_{[1,L]}, \tilde{j}_{1:i-1}\in\mathcal{L}_{i-1}, \tilde{j}_{1:i}\in\mathcal{L}_i$ in defining the vertices of optimal sets at time k for notational clarity in defining the constructed solution at time $k+1$, i.e., respectively using $e_r^{(j)*}(1|k)$ and $e_r^{(j,\tilde{j}_{1:i-1})*}(i|k)$ to replace $e_r^{(\tilde{j}_1)*}(1|k)$ and $e_r^{(\tilde{j}_{1:i})*}(i|k)$ with $i\in\mathbb{N}_{[1,N]}$.

The vertices of optimal interpolated state and input residual sets at time k defined as

$$\{e_r^*(0|k), e_r^{(j)*}(1|k), e_r^{(j,\tilde{j}_1)*}(2|k), \cdots, e_r^{(j,\tilde{j}_{1:N-1})*}(N|k)\}, \{\mu_r^*(0|k), \mu^{(j)*}(1|k), \mu_r^{(j,\tilde{j}_1)*}(2|k), \cdots, \mu_r^{(j,\tilde{j}_{1:N-2})*}(N-1|k)\},$$

satisfy (86)-(88) and (93). Similar to the construction of $\hat{e}_r^{\tilde{j}_{1:i}}(i|k+1), \hat{\mu}_r^{\tilde{j}_{1:i}}(i|k+1)$ in (55)-(57), it can be similarly verified that the vertices defined by

$$\hat{e}_r(0|k+1)=e_r^*(1|k)=\textstyle\sum_{j=1}^L p^{(j)}e_r^{(j)*}(1|k), \hat{\mu}_r(0|k+1)=\mu_r^*(1|k)=\sum_{j=1}^L p^{(j)}\mu_r^{(j)*}(1|k), \tag{113}$$

$$\hat{e}_r^{\tilde{j}_{1:i}}(i|k+1)=\textstyle\sum_{j=1}^L p^{(j)}e_r^{(j,\tilde{j}_{1:i})*}(i+1|k), i\in\mathbb{N}_{[1,N-1]}, \hat{\mu}_r^{\tilde{j}_{1:i}}(i|k+1)=\sum_{j=1}^L p^{(j)}\mu_r^{(j,\tilde{j}_{1:i})*}(i+1|k), i\in\mathbb{N}_{[1,N-2]}, \tag{114}$$

with $p^{(j)}$ satisfying (51) and $\tilde{j}_{1:i}\in\mathcal{L}_i$,

$$\hat{e}_r^{(\tilde{j}_{1:N-1},j_N)}(N|k+1)=(A^{(j_N)}+B^{(j_N)}K_r)\hat{e}_r^{\tilde{j}_{1:N-1}}(N-1|k+1), \hat{\mu}_r^{\tilde{j}_{1:N-1}}(N-1|k+1)=K_r\hat{e}_r^{\tilde{j}_{1:N-1}}(N-1|k+1), \tag{115}$$

for all $\tilde{j}_{1:N-1}\in\mathcal{L}_{N-1}, j_N\in\mathbb{N}_{[1,L]}$, satisfy, at time $k+1$, the constraints (86)-(88) and (93) by considering (107) and (111).

To construct the candidate set of interpolated state $\hat{\mathcal{X}}_r(i|k+1)$ and input $\hat{\mathcal{U}}_r(i|k+1)$, we first define these two sets as follows,

$$\hat{\mathcal{X}}_r(i|k+1)=Co\{\hat{x}_r^{\tilde{j}_{1:i}}(i|k+1), \tilde{j}_{1:i}\in\mathcal{L}_i\}, i\in\mathbb{N}_{[1,N]}, \hat{\mathcal{U}}_r(i|k+1)=Co\{\hat{u}_r^{\tilde{j}_{1:i}}(i|k+1), \tilde{j}_{1:i}\in\mathcal{L}_i\}, i\in\mathbb{N}_{[1,N-1]},$$
$$\hat{\mathcal{X}}_r(0|k+1)=\{\hat{x}_r(0|k+1)\}, \hat{\mathcal{U}}_r(0|k+1)=\{\hat{u}_r(0|k+1)\}.$$

And the vertices of optimal interpolated state and input sets at time k defined as

$$\{x_r^*(0|k), x_r^{(j)*}(1|k), x_r^{(j,\tilde{j}_1)*}(2|k), \cdots, x_r^{(j,\tilde{j}_{1:N-1})*}(N|k)\}, \{u_r^*(0|k), u_r^{(j)*}(1|k), u_r^{(j,\tilde{j}_1)*}(2|k), \cdots, u_r^{(j,\tilde{j}_{1:N-2})*}(N-1|k)\},$$

satisfy (89) and (94). Similar to the construction of $\hat{x}^{\tilde{j}_{1:i}}(i|k+1), \hat{u}^{\tilde{j}_{1:i}}(i|k+1)$ in (58)-(60), it can be verified that the vertices defined by

$$\hat{x}_r(0|k+1)=\textstyle\sum_{j=1}^L p^{(j)}x_r^{(j)*}(1|k), \hat{u}_r(0|k+1)=\sum_{j=1}^L p^{(j)}u_r^{(j)*}(1|k), \tag{116}$$

$$\hat{x}_r^{\tilde{j}_{1:i}}(i|k+1)=\textstyle\sum_{j=1}^L p^{(j)}x_r^{(j,\tilde{j}_{1:i})*}(i+1|k), i\in\mathbb{N}_{[1,N-1]}, \hat{u}_r^{\tilde{j}_{1:i}}(i|k+1)=\sum_{j=1}^L p^{(j)}u_r^{(j,\tilde{j}_{1:i})*}(i+1|k), i\in\mathbb{N}_{[1,N-2]}, \tag{117}$$
$$\hat{x}_r^{(\tilde{j}_{1:N-1},j_N)}(N|k+1)=(A^{(j_N)}+B^{(j_N)}K_r)\hat{x}_r^{\tilde{j}_{1:N-1}}(N-1|k+1), \hat{u}_r^{\tilde{j}_{1:N-1}}(N-1|k+1)=K_r\hat{x}_r^{\tilde{j}_{1:N-1}}(N-1|k+1)$$

with $\tilde{j}_{1:i}\in\mathcal{L}_i, j_N\in\mathbb{N}_{[1,L]}, \tilde{j}_{1:N-1}\in\mathcal{L}_{N-1}$, satisfy, at time $k+1$, the constraints (89) and (94) by considering (109)-(110) and (113)-(115). Therefore, given

$$\{x_r^*(0|k), x_r^{(j)*}(1|k), x_r^{(j,\tilde{j}_1)*}(2|k), \cdots, x_r^{(j,\tilde{j}_{1:N-1})*}(N|k)\}, \{u_r^*(0|k), u_r^{(j)*}(1|k), u_r^{(j,\tilde{j}_1)*}(2|k), \cdots, u_r^{(j,\tilde{j}_{1:N-2})*}(N-1|k)\}$$

satisfying (97)-(99), the constraint (97)-(98) hold at time $k+1$ by convexity. The constraint (99) holds with $\hat{\lambda}_r^*(k+1)=\lambda_r^*(k)$ as $\hat{x}_r^{\tilde{j}_{1:N-1}}(N-1|k+1)\in\lambda_r^*(k)\Omega_r$ caused by (117) and $x_r^{(j,\tilde{j}_{1:N-1})*}(N|k)\in\lambda_r^*(k)\Omega_r$, and Ω_r is a RPI set for system (1) subject to the constraint (2) with $u=K_r x$. (74) is ensured by noting $x(k+1)=\sum_{j=1}^L p^{(j)}x^{(j)*}(1|k)$ with $p^{(j)}$ satisfying (51) and

$x^{(j)*}(1|k)$ defined in (90) with $\tilde{j}_1 = j$ and $\hat{x}_r(0|k+1) = \sum_{j=1}^{L} p^{(j)} x_r^{(j)*}(1|k)$ holds by construction. Thus the candidate solution $\{\hat{x}_r(0|k+1), \hat{\lambda}_r(k+1), \hat{\mathbf{c}}_{\mathbf{r}}(\mathbf{k+1}), \hat{\boldsymbol{\theta}}_{\mathbf{r}}(\mathbf{k+1})\}_{r=1,2}$ is feasible at time $k+1$.

Closed-loop stability The closed-loop stability is proven by showing monotonic decrease of the optimal cost, denote the optimal cost of problem (104) at time k and cost of the constructed feasible solution at time $k+1$ respectively as

$$V(\xi_1^*(k), \xi_2^*(k)) \triangleq \sum_{r=1}^{2} V_r(\xi_r^*(k)), V(\hat{\xi}_r(k+1)) \triangleq \sum_{r=1}^{2} V_r(\hat{\xi}_r(k+1)), \tag{118}$$

with $V_r(\xi_r^*(k)) = (\xi_r^*(k))^T M_r(k) \xi_r^*(k), V_r(\hat{\xi}_r(k+1)) = (\hat{\xi}_r(k+1))^T M_r(k+1) \hat{\xi}_r(k+1)$,

$$\xi_r^*(k) = \texttt{col}\left[\underline{\theta}_r^*(k), 1, \underline{c}_r^*(0|k)\right], \hat{\xi}_r(k+1) = \texttt{col}\left[\underline{\hat{\theta}}_r(k+1), 1, \underline{\hat{c}}_r(0|k+1)\right], \tag{119}$$

$$\underline{\theta}_r^*(k) = \boldsymbol{\theta}_r^*(k), \underline{\hat{\theta}}_r(k+1) = \hat{\boldsymbol{\theta}}_r(k+1), \underline{c}_r^*(0|k) = \mathbf{c}_{\mathbf{r}}^*(k), \underline{\hat{c}}_r(0|k+1) = \hat{\mathbf{c}}_{\mathbf{r}}(k+1), \tag{120}$$

with $\boldsymbol{\theta}_{\mathbf{r}}^*(k), \mathbf{c}_{\mathbf{r}}^*(k)$ and $\hat{\boldsymbol{\theta}}_{\mathbf{r}}(k+1), \hat{\mathbf{c}}_{\mathbf{r}}(k+1)$ respectively defined in (105), (107)-(108). Denote the optimal interpolated augment vector revised from (35) at time k as

$$\varsigma_r^*(i|k) = \texttt{col}\left[\underline{z}_r^*(i|k), \underline{v}_r^*(i|k), e_r^*(i|k), \underline{c}_r^*(i|k), \underline{W}_r^*(i|k)\right], \tag{121}$$

with $e_r^*(i|k) \in Co\{\hat{e}_r^{(\tilde{j}_{1:i})*}(i|k), \tilde{j}_{1:i} \in \mathcal{L}_i\}$, $\underline{z}_r^*(i|k) = \texttt{col}[z_r^*(i|k), z_r^*(i+1|k), \cdots, z_r^*(i+N-1|k)]$, $\underline{v}_r^*(i|k) = \texttt{col}[v_r^*(i|k), v_r^*(i+1|k), \cdots, v_r^*(i+N-1|k)], \tilde{j}_{1:i} \in \mathcal{L}_i\}$, $\underline{c}_r^*(i|k) = \texttt{col}[c_r^*(i|k), \cdots, c_r^*(i+N-1|k)]$, $\underline{W}_r^*(i|k) = \texttt{col}[W_r^*(i|k), W_r^*(i+1|k), \cdots, W_r^*(i+N-1|k)]$ and define the interpolated constructed solution at time $k+1$ as

$$\hat{\varsigma}_r(i|k+1) = \texttt{col}\left[\underline{\hat{z}}_r(i|k+1), \underline{\hat{v}}_r(i|k+1), \hat{e}_r(i|k+1), \underline{\hat{c}}_r(i|k+1), \underline{\hat{W}}_r(i|k+1)\right], \tag{122}$$

with $\hat{e}_r(i|k+1) \in Co\{\hat{e}_r^{\tilde{j}_{1:i}}(i|k+1), \tilde{j}_{1:i} \in \mathcal{L}_i\}$ for $i \in \mathbb{N}_{\geq 1}$ and $\hat{e}_r(0|k+1)$ defined in (113)-(115),

$$\underline{\hat{z}}_r(i|k+1) = \texttt{col}\left[\hat{z}_r(i|k+1), \cdots, \hat{z}_r(i+N-1|k+1)\right], \underline{\hat{c}}_r(i|k+1) = \texttt{col}\left[\hat{c}_r(i|k+1), \cdots, \hat{c}_r(i+N-1|k+1)\right], \tag{123}$$

$$\underline{\hat{v}}_r(i|k+1) = \texttt{col}\left[\hat{v}_r(i|k+1), \cdots, \hat{v}_r(i+N-1|k+1)\right], \underline{\hat{W}}_r(i|k+1) = \texttt{col}\left[\hat{W}_r(i|k+1), \cdots, \hat{W}_r(i+N-1|k+1)\right]. \tag{124}$$

As indicated by (76)-(78) and (80)-(81), $\varsigma_r^*(0|k) = D(k)\xi_r^*(k)$ and $\hat{\varsigma}_r(0|k+1) = D(k+1)\hat{\xi}_r(k+1)$ with $\xi_r^*(k)$ and $\hat{\xi}_r(k+1)$ defined in (119). Hence, considering (101) and (118),

$$V_r(\xi_r^*(k)) = (\varsigma_r^*(0|k))^T P_r \varsigma_r^*(0|k), V_r(\hat{\xi}_r(k+1)) = (\hat{\varsigma}_r(0|k+1))^T P_r \hat{\varsigma}_r(0|k+1). \tag{125}$$

From (79), (107), (109)-(111), (113), (121)-(124) and noting that T_{n_x}, T_{n_u} defined in (37) satisfy $\underline{z}_r^*(1|k) = T_{n_x}\underline{z}_r^*(0|k), \underline{W}_r^*(1|k) = T_{n_x}\underline{W}_r^*(0|k), \underline{v}_r^*(1|k) = T_{n_u}\underline{v}_r^*(0|k), \underline{c}_r^*(1|k) = T_{n_u}\underline{c}_r^*(0|k)$ which is similar with (38),

$$\hat{\varsigma}_r(0|k+1) = \varsigma_r^*(1|k) = \psi_r \varsigma_r^*(0|k), \tag{126}$$

with $\psi_r = \sum_{j=1}^{L} p^{(j)} \psi_r^{(j)}, \psi_r^{(j)}$ defined in (103) and $p^{(j)}$ satisfying (51). Therefore, since (102) implying $P_r \geq \psi_r^T P_r \psi_r + \bar{Q}_r$, by pre- and post-multiplication of this inequality by $(\varsigma_r^*(0|k))^T$ and its transpose and considering (75), (77), (125)-(126) and E_{n_x}, E_{n_u} defined in (36) satisfy $z_r^*(0|k) = E_{n_x}\underline{z}_r^*(0|k), v_r^*(0|k) = E_{n_u}\underline{v}_r^*(0|k), W_r^*(0|k) = E_{n_x}\underline{W}_r^*(0|k)$, we have $V_r(\xi_r^*(k)) - V_r(\hat{\xi}_r(k+1)) \geq \|x_r(0|k)\|_Q^2 + \|u_r(0|k)\|_R^2$ which implies $V_r(\xi_r^*(k)) - V_r(\xi_r^*(k+1)) \geq \|x_r(0|k)\|_Q^2 + \|u_r(0|k)\|_R^2$, $V(\xi_1^*(k), \xi_2^*(k)) - V(\xi_1^*(k+1), \xi_2^*(k+1)) \geq \sum_{r=1}^{2}\left[\|x_r(0|k)\|_Q^2 + \|u_r(0|k)\|_R^2\right] > 0, \forall x_r(0|k) \neq 0, u_r(0|k) \neq 0 (x(k) = x_1(k) + x_2(k) \neq 0, u(k) = u_1(0|k) + u_2(0|k) \neq 0)$ as Q, R are positive definite. Hence, the closed-loop system is asymptotically stable. □

5 | NUMERICAL EXAMPLES

In this section, we demonstrate the advantages of our proposed data-driven methods by two examples. First, the proposed data-driven controller is applied to a four dimensional system. We show that our proposed scheme can achieve desirable control performance with good computational efficiency (Section 5.1). Next, we show the potential of our proposed scheme in controlling nonlinear plant through a CSTR chemical process benchmark example (Section 5.2).

Algorithm 2 Input-mapping data-driven interpolation robust model predictive control (IMD-IRMPC) algorithm

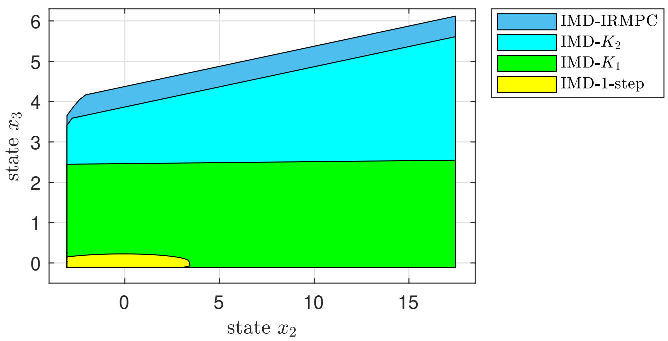
Offline:

Step 1. Choose the cost indexes γ_1 and γ_2 and two reference points $x_{p,1}$ and $x_{p,2}$, use (70) to calculate the stabilizing feedback gains $\{K_1, K_2\}$ and apply the algorithm in [24] to calculate the corresponding polyhedral RPI sets $\{\Omega_1, \Omega_2\}$. Compute the Lyapunov matrices P_1, P_2 satisfying Lemma 6.

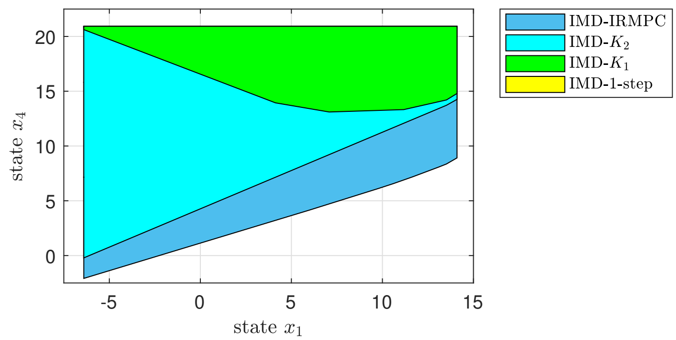
Online:

Step 1. Compute matrix $M_r(k)$ and solve problem (104). Apply input $u(k) = \sum_{r=1}^{2}\left[\sum_{l=1}^{N_L}(\theta_r(0|k))_l u(k-l) + K_r e_r(0|k) + c_r(0|k)\right]$ to system.

Step 2. Update data set X_k^-, U_k^-, X_k^+ in (5)-(7) by storing a new record $\{x(k), u(k), x(k+1)\}$, set $k = k+1$ and back to Step 1.



(a) The $x_2 - x_3$ cross section of feasible region



(b) The $x_1 - x_4$ cross section of feasible region

FIGURE 3 The comparison of feasible region under IMD-IRMPC, IMD-K_1, IMD-K_2 and IMD-1-step. (a): The $x_2 - x_3$ cross section of feasible region with $(x_1, x_4) = (1, 2)$ (b): The $x_1 - x_4$ cross section of feasible region with $(x_2, x_3) = (7, 5)$ In (b), the $x_1 - x_4$ cross section of feasible region of IMD-1-step is empty (contains no points)

5.1 | Example 1: numerical example

We apply our method to the following fourth-order system. The discrete time state space model is in the form of

$$x(k+1) = (A_0 + \Delta_A^{\mathrm{tr}})x(k) + (B_0 + \Delta_B^{\mathrm{tr}})u(k), [\Delta_A^{\mathrm{tr}}, \Delta_B^{\mathrm{tr}}] \in \Xi \triangleq Co\{[\Delta_A^{(1)}, \Delta_B^{(1)}], [\Delta_A^{(2)}, \Delta_B^{(2)}]\},$$

$$A_0 = \begin{bmatrix} 0.8623 & 0 & 0.1472 & 0 \\ 0 & 0.8359 & 0 & 0.1052 \\ 0 & 0 & 0.6434 & 0 \\ 0 & 0 & 0 & 0.8130 \end{bmatrix}, B_0 = \begin{bmatrix} 0.0647 & 0 \\ 0 & 0.0647 \\ 0 & 0.0680 \\ 0.0619 & 0 \end{bmatrix},$$

$$\Delta_A^{(1)} = \begin{bmatrix} 0 & 0 & -0.1177 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0.3163 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \Delta_A^{(2)} = \begin{bmatrix} 0 & 0 & 0.1177 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -0.3163 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \Delta_B^{(1)} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & -0.0204 \\ 0 & 0 \end{bmatrix}, \Delta_B^{(2)} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0.0204 \\ 0 & 0 \end{bmatrix},$$

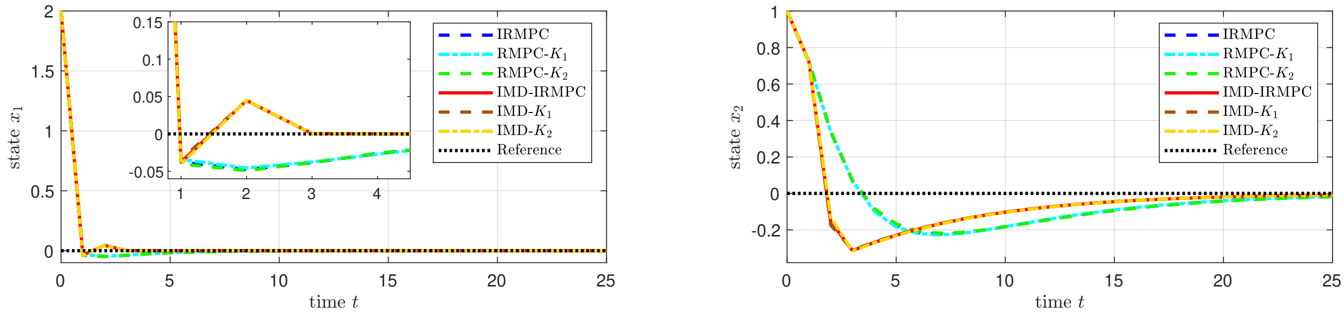
$$A^{(1)} = A_0 + \Delta_A^{(1)}, A^{(2)} = A_0 + \Delta_A^{(2)}, B^{(1)} = B_0 + \Delta_B^{(1)}, B^{(2)} = B_0 + \Delta_B^{(2)}.$$

The constraints of the system are

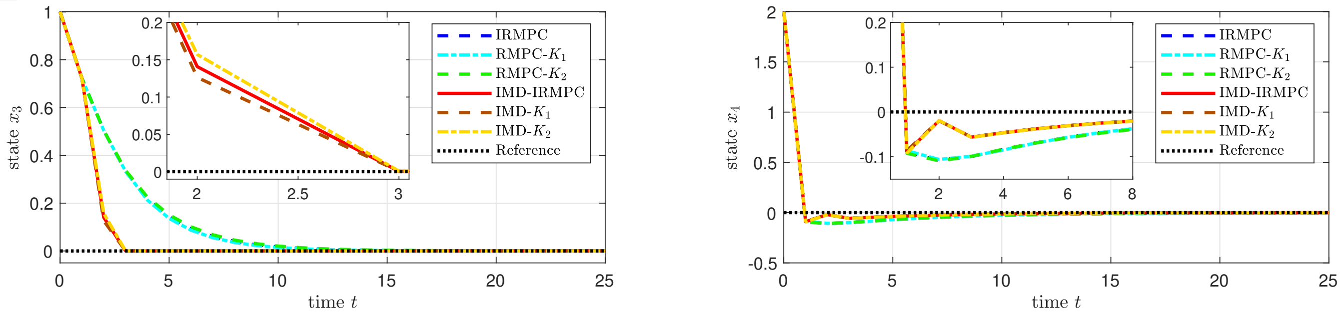
$$U = \{u \in \mathbb{R}^2 | -42.7506 \le u_1 \le 17.2494, -30.5563 \le u_2 \le 29.4437\},$$

$$X = \{x \in \mathbb{R}^4 | -6.4033 \le x_1 \le 14.0967, -3.0544 \le x_2 \le 17.4456, -0.1144 \le x_3 \le 24.3856, -2.5748 \le x_4 \le 20.9252\}. \quad (127)$$

The weighting matrices are $Q = diag\{20, 0.01, 200, 0\}$, $R = 0.001 I_2$. The data length and control horizon are chosen as $N_L = 4$, $N = 2$. For the input-mapping interpolation RMPC algorithm (Algorithm 2), we firstly solve (70) to get two feedback



(a) Trajectories of x_1 starting from initial state $x_0 = (2, 1, 1, 2)^T$ (b) Trajectories of x_2 starting from initial state $x_0 = (2, 1, 1, 2)^T$



(c) Trajectories of x_3 starting from initial state $x_0 = (2, 1, 1, 2)^T$ (d) Trajectories of x_4 starting from initial state $x_0 = (2, 1, 1, 2)^T$

FIGURE 4 The comparison of state evolution starting from $x_0 = (2, 1, 1, 2)^T$ under IMD-IRMPC, IMD-K_1, IMD-K_2 and their robust counterparts IRMPC, RMPC-K_1, RMPC-K_2. (a): Trajectories of state x_1. (b): Trajectories of state x_2. (c): Trajectories of state x_3. (d): Trajectories of state x_4

TABLE 1 Comparison of accumulated cost between robust and data-driven methods for Exampe 1

Initial state	IMD-IRMPC	IRMPC	IMD-K_1	RMPC-K_1	IMD-K_2	RMPC-K_2
$x_0 = \begin{pmatrix} 2 & 1 & 1 & 2 \end{pmatrix}^T$	389.3	474.0	388.5	473.7	390.2	477.3

gains with different properties. For the tuning parameters in (70), we set the cost indices and the reference points of the high performance and low performance feedback respectively as $\gamma_1 = 10000, \gamma_2 = 14600$ and $x_{p,1} = (135, 35, 10, 8)^T, x_{p,2} = (111, 24, 0.01, 5)^T$. By solving (70), the resulting high-gain and low-gain feedback controllers are respectively selected as $K_1 = \begin{bmatrix} -2.5141 & -0.0808 & -2.2485 & 0.1793 \\ 0.2034 & 0.0093 & -5.8007 & 0.3582 \end{bmatrix}$, $K_2 = \begin{bmatrix} -1.6614 & -0.0879 & -0.2392 & -0.3194 \\ -0.0518 & 0.0537 & -4.2735 & 0.0693 \end{bmatrix}$.

The feasible region of our proposed data-driven methods IMD-K_1, IMD-K_2 (Algorithm 1 with two feedback laws K_1, K_2), IMD-IRMPC (Algorithm 2, IMD-interpolation robust model predictive control) and IMD-1-step (IMD method with control horizon $N = 1$ in [22]) is shown in Figure 3. Since it is difficult to plot the feasible region for 4 dimensional system, we randomly choose some points within the constraint admissible region, respectively fix x_1, x_4 and x_2, x_3 and plot the $x_2 - x_3$ and $x_1 - x_4$ cross sections of feasible region. It can be observed that IMD-IRMPC enlarges the feasible region of IMD-K_1 and IMD-K_2 as a result of the interpolation between two control laws K_1, K_2, as stated in Remark 2. And due to the fact that ellipsoid invariant set deals with the asymmetric constraint of state x_3 defined in (127) conservatively by treating it as a symmetric constraint (i.e., $-0.1144 \leq x_3 \leq 0.1144$), IMD-1-step has a relatively small attractive region on $x_2 - x_3$ plane with the fixed coordinate $(x_1, x_4) = (1, 2)$ and even infeasible on $x_1 - x_4$ plane with the fixed coordinate $(x_2, x_3) = (7, 5)$. Since its attractive region is negligible and near the origin, we do not perform comparison with IMD-1-step in this example but in Example 2.

TABLE 2 Comparison of accumulated cost and average (total) computation time between data-driven controllers for Example 1 (- = infeasible)

Initial state x_0	IMD-IRMPC		IMD-K_1		AMPC-K_1		AMPCs-K_1		IMD-K_2		AMPC-K_2		AMPCs-K_2	
	Time/s	Cost	Time/s	Cost	Time/s	Cost	Time/s	Cost	Time/s	Cost	Time/s	Cost	Time/s	Cost
$(5\ 6\ 4\ 5)^T$	0.0082	6095.6	-	-	-	-	-	-	0.0070	6095.6	3.28	6097.4	0.22	6841.7
$(2\ 1\ 1\ 2)^T$	0.0083	389.3	0.0075	388.5	3.38	390.8	0.22	440.0	0.0069	390.2	3.29	391.1	0.20	451.0
$(10\ 7\ 6\ 12)^T$	0.0085	17466	-	-	-	-	-	-	-	-	-	-	-	-

TABLE 3 Comparison of computation efficiency between online data-driven controllers with initial state $x_0 = (2, 1, 1, 2)^T$ for Example 1

Algorithm	IMD-IRMPC	IMD-K_1	AMPC-K_1	AMPCs-K_1	IMD-K_2	AMPC-K_2	AMPCs-K_2
Updating time(s)	0.0009	0.0007	3.31	0.14	0.0007	3.23	0.14
Solving time(s)	0.0074	0.0068	0.07	0.08	0.0062	0.06	0.06
Total time(s)	0.0083	0.0075	3.38	0.22	0.0069	3.29	0.20

5.1.1 | The comparison with RMPC methods

In this subsection, we compare the performance of our proposed data-driven methods IMD-K_1, IMD-K_2 (Algorithm 1 with two feedback laws K_1, K_2), IMD-IRMPC (Algorithm 2, IMD-interpolation robust model predictive control) and its robust counterparts (RMPC-K_1, RMPC-K_2, IRMPC) introduced in [6].

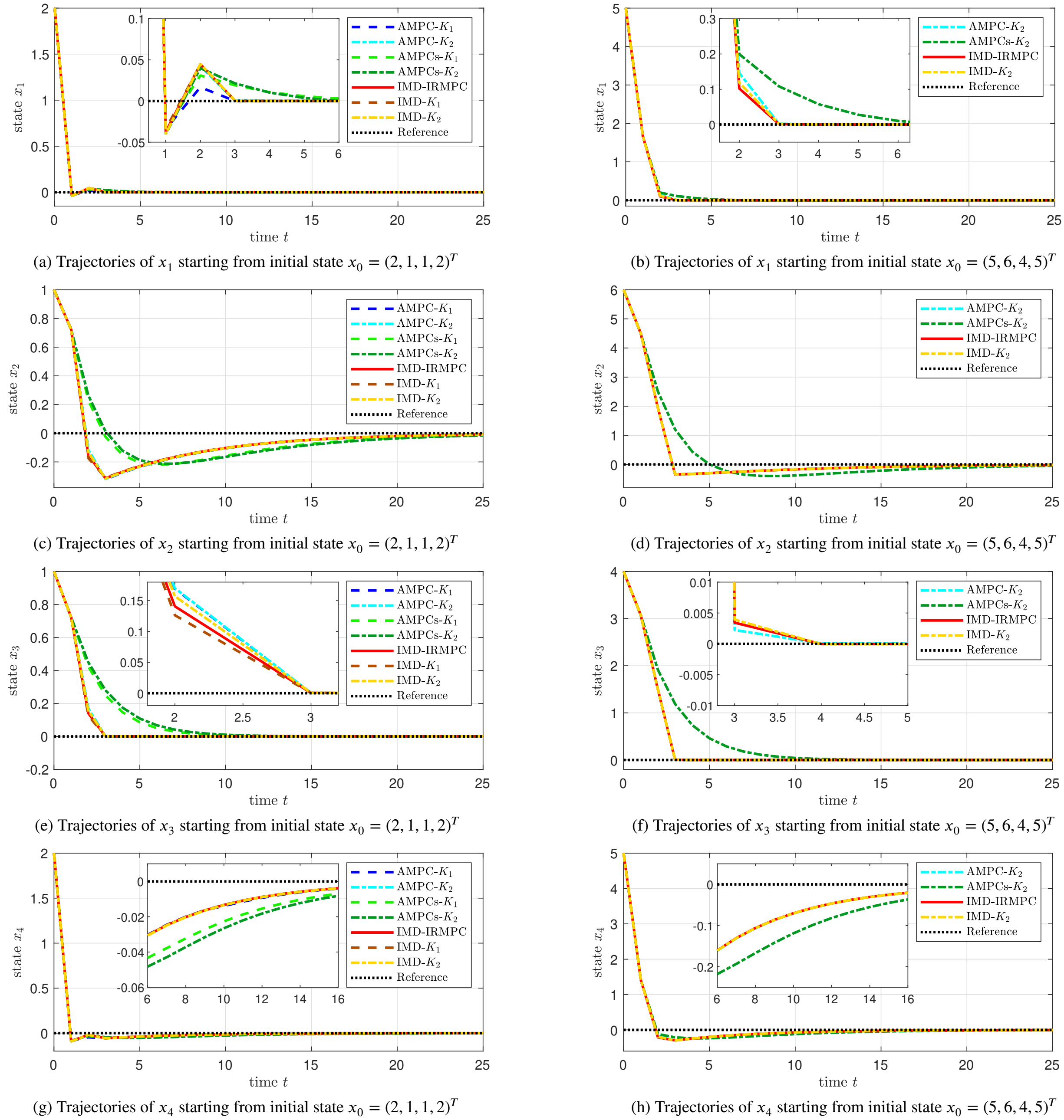
To compare performance, we select an initial state $x_0 = (2, 1, 1, 2)^T$ and the state trajectories of different methods are shown in Figure 4. It can be seen that the proposed IMD-IRMPC converge faster than its counterpart IRMPC, which also applies for IMD-K_1 and IMD-K_2 compared with their robust counterparts. Moreover, as the feedback controller of K_1 is designed to have better performance than K_2, IMD-K_1 performs better than IMD-K_2 and IMD-IRMPC achieves almost the same desirable performance with IMD-K_1. Further, Table 1 lists the accumulated cost index which is denoted as $J_a = \sum_{k=0}^{T_{step}} (x^T(k)Qx(k) + u^T(k)Ru(k))$ with $T_{step} = 25$ denoting the simulation time. It can be readily seen that the proposed input-mapping data-driven methods improve the performance of robust methods.

5.1.2 | The comparison with online data-driven MPC methods

In this subsection, we make a comparison between two online data-driven methods in terms of closed-loop performance and computation efficiency, namely our proposed input-mapping data-driven methods and AMPC (adaptive MPC) approach in [8] which uses online measurements to reduce the size of polytopic model set in (4). As the AMPC is based on the fixed feedback gain, AMPC-K_1 and AMPC-K_2 are both taken into comparison. And the simplified version of AMPC, denoted as AMPCs-K_1 and AMPCs-K_2 (which is introduced in [8] to simply update model set without recomputing the terminal set and Lyapunov weighting matrix), are also considered. To give a fair comparison, the control horizon in these two approaches are kept the same as mentioned above, $N = 2$. The same prior polytopic set Ξ in (4) is adopted in all cases.

To compare control performance and computation efficiency, we select three initial points, respectively $x_0 = (5, 6, 4, 5)^T, (2, 1, 1, 2)^T$ and $(10, 7, 6, 12)^T$. The performance is evaluated through an accumulated cost for $T_{step} = 50$ time steps. To assess the computation efficiency, the average (total) computation time over 50 time steps is measured which includes the time spent for online learning (in updating the estimated model, model uncertainty set, terminal set and Lyapunov matrix for AMPC and updating the data set $X_k^-.U_k^-, X_k^+$ in (5)-(7) for the proposed IMD methods) and solving the optimization problem.

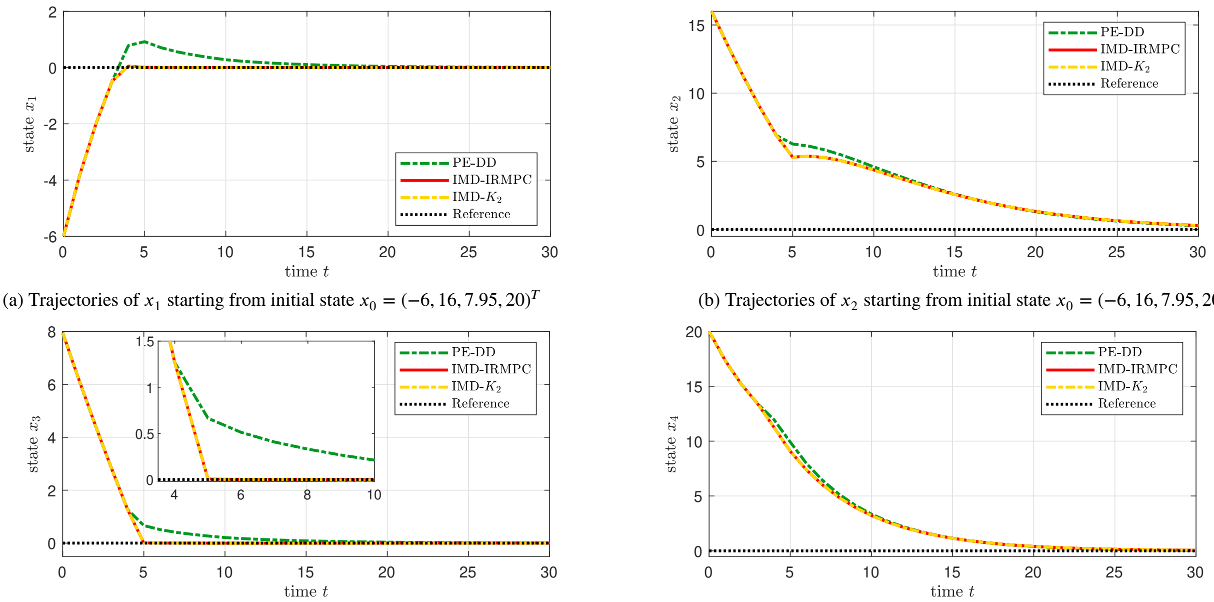
The results are reported as follows. Table 2 shows the performance and average (total) computation time with regard to different initial points under IMD-IRMPC, IMD-K_1, IMD-K_2, AMPC-K_1, AMPC-K_2, AMPCs-K_1 and AMPCs-K_2. As can be seen from Table 2, the methods based on one feedback gain $K_r(r = 1, 2)$ are infeasible for some initial points or only one K_r provides a feasible solution while the proposed IMD-IRMPC always provides feasibility due to the enlarged attractive region. We also observe that IMD-IRMPC, IMD-K_1 and IMD-K_2 can achieve superior control performance and computational efficiency simultaneously compared to AMPC-K_1, AMPC-K_2, AMPCs-K_1 and AMPCs-K_2. And AMPC-K_r with $r = 1, 2$ improves the control performance of AMPCs-K_r at the cost of more computation burden which includes not only updating the estimated



(a) Trajectories of x_1 starting from initial state $x_0 = (2, 1, 1, 2)^T$
(b) Trajectories of x_1 starting from initial state $x_0 = (5, 6, 4, 5)^T$
(c) Trajectories of x_2 starting from initial state $x_0 = (2, 1, 1, 2)^T$
(d) Trajectories of x_2 starting from initial state $x_0 = (5, 6, 4, 5)^T$
(e) Trajectories of x_3 starting from initial state $x_0 = (2, 1, 1, 2)^T$
(f) Trajectories of x_3 starting from initial state $x_0 = (5, 6, 4, 5)^T$
(g) Trajectories of x_4 starting from initial state $x_0 = (2, 1, 1, 2)^T$
(h) Trajectories of x_4 starting from initial state $x_0 = (5, 6, 4, 5)^T$

FIGURE 5 The comparison of state evolution starting from initial state $x_0 = (2, 1, 1, 2)^T$ (left) and $x_0 = (5, 6, 4, 5)^T$ (right) under IMD-IRMPC, IMD-K_1, IMD-K_2, AMPC-K_1, AMPC-K_2, AMPCs-K_1 and AMPCs-K_2

model and model uncertainty set, but also recomputing the terminal set and Lyapunov weighting matrix. It is noteworthy that our proposed multi-step IMD based methods have much less computation time than AMPC and AMPCs methods by directly using data to design controller without estimating the model online as in AMPC and AMPCs methods, contributing to the obvious advantages of both performance and computation efficiency. Table 3 compares the time spent in online updating and solving the optimization problem between different algorithms. This reveals that the proposed multi-step IMD methods implement online learning by simply updating the data set $X_k^-.U_k^-, X_k^+$ as defined in (5)-(7) while AMPC needs to at least update the estimated model and model uncertainty set for each sampling time. For the time spent in solving optimization, AMPC and AMPCs have more optimization variables and constraints resulting from the decision variables of tube design and an extending control horizon.



(a) Trajectories of x_1 starting from initial state $x_0 = (-6, 16, 7.95, 20)^T$

(b) Trajectories of x_2 starting from initial state $x_0 = (-6, 16, 7.95, 20)^T$

(c) Trajectories of x_3 starting from initial state $x_0 = (-6, 16, 7.95, 20)^T$

(d) Trajectories of x_4 starting from initial state $x_0 = (-6, 16, 7.95, 20)^T$

FIGURE 6 The comparison of state evolution with initial state starting from $x_0 = (-6, 16, 7.95, 20)^T$ under IMD-IRMPC, IMD-K_2, PE-DD in the case of nominal plant

TABLE 4 Comparison of accumulated cost and average computation time in the case of nominal plant under IMD-IRMPC, IMD-K_2, PE-DD and non-PE-DD for Example 1 (- = infeasible)

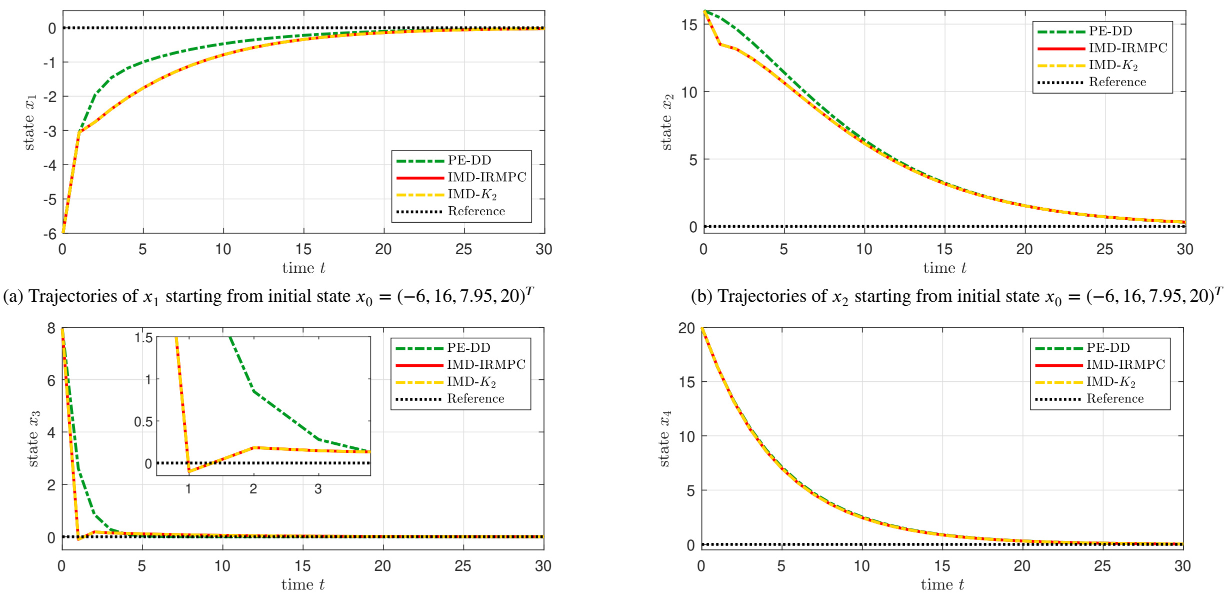
Algorithm	IMD-IRMPC	IMD-K_2	PE-DD	non-PE-DD
Cost	27307	27307	27602	-
Computation time(s)	0.0074	0.0055	0.0103	-

To display the stabilization performance more clearly, the closed-loop trajectories with two initial points $x_0 = (5, 6, 4, 5)^T, (2, 1, 1, 2)^T$ over the former 25 time steps are plotted in Figure 5. The closed-loop trajectories of the initial state $x_0 = (10, 7, 6, 12)^T$ are not shown as only the IMD-IRMPC is feasible for this initial state and convergence to the origin is evident. It can be seen that IMD-IRMPC, IMD-K_1 and IMD-K_2 converge faster than the AMPC-K_r and AMPCs-K_r methods with $r = 1, 2$, which clearly verifies the effectiveness of the proposed input-mapping data-driven scheme.

5.1.3 | The comparison with offline data-driven based methods

In this subsection, we make a comparison with an offline data-driven based method, namely the PE-DD (persistently exciting data-driven) approach which collects a certain amount of offline data satisfying PE condition in [10], in terms of closed-loop performance, computation efficiency and robustness in presence of model mismatch.

We perform the comparison under two cases, respectively the nominal and model mismatch. For the nominal case, the online real model parameter coincides with the model used in collecting data during offline trials, i.e., $[A^{\text{offline}}, B^{\text{offline}}] = [A^{\text{online}}, B^{\text{online}}] = [A^{(1)}, B^{(1)}]$. In the case of model mismatch, the online model is inconsistent with the model used in offline trials, i.e., $\left[A^{\text{offline}}, B^{\text{offline}}\right] = \left[A^{(1)}, B^{(1)}\right], \left[A^{\text{online}}, B^{\text{online}}\right] = \left[A^{(2)}, B^{(2)}\right]$. To give a fair comparison, we replace the min-max objective function of IMD-IRMPC in (104) (IMD-RMPC in (43)) into a nominal cost function to be consistent with PE-DD. Moreover, to guarantee recursive feasibility and stability, we consider the robust state constraint satisfaction. In consideration of insufficient data not satisfying PE condition, we also make a comparison with non-PE-DD where the not persistently exciting data is collected by reducing the data length such that the PE condition is not satisfied.



(a) Trajectories of x_1 starting from initial state $x_0 = (-6, 16, 7.95, 20)^T$

(b) Trajectories of x_2 starting from initial state $x_0 = (-6, 16, 7.95, 20)^T$

(c) Trajectories of x_3 starting from initial state $x_0 = (-6, 16, 7.95, 20)^T$

(d) Trajectories of x_4 starting from initial state $x_0 = (-6, 16, 7.95, 20)^T$

FIGURE 7 The comparison of state evolution with initial state starting from $x_0 = (-6, 16, 7.95, 20)^T$ under IMD-IRMPC, IMD-K_2, PE-DD in the case of model mismatch

TABLE 5 Comparison of accumulated cost in presence of model mismatch under IMD-IRMPC, IMD-K_2 and PE-DD for Example 1

Algorithm	**IMD-IRMPC**	**IMD-K_2**	**PE-DD**
Cost	14161	14161	15303

To compare control performance and computation efficiency, the initial state is chosen as $x_0 = (-6, 16, 7.95, 20)^T$ which is near the boundary of the feasible region. For PE-DD, the control horizon is set as $N = 15$ which is the minimal control length that can provide feasibility for the chosen initial state x_0. The length of data is set as $N_L = 56$ which is the lower bound of data length satisfying the persistent excitation condition. For non-PE-DD, the data length is reduced to $N_L = 46$. The tuning parameters regarding control horizon and data length of our proposed methods are consistent with Section 5.1.2.

The results are reported as follows. Table 4 shows the comparison between different algorithms in terms of accumulated cost and computation efficiency. For the performance cost, it can be observed that IMD-IRMPC and IMD-K_2 have better performance than PE-DD in the cases of nominal plant. The stabilization performance is plotted in Figure 6. It can be seen that our proposed methods perform better than PE-DD. The advantage can be explained by noting that a conservative terminal equality constraint is used in [10] which requires longer control horizon to provide feasibility and desirable performance. In the case of insufficient data, non-PE-DD becomes infeasible. This further implies the advantage of our methods over PE-DD lies in that we do not need to implement offline trials to collect sufficient amount of data satisfying PE condition, making it possible to be applied to large-scale systems which have are economically expensive to implement offline trials.

To compare computation efficiency, the average computation time is listed in Table 4. The results show that IMD-IRMPC and IMD-K_2 has better computation efficiency than PE-DD. This is due to the fact that the control horizon of IMD-IRMPC nad IMD-K_2 are set as $N = 2$, much less than PE-DD which requires $N = 15$ to provide feasibility.

In the presence of model mismatch, as can be seen from Table 5, our proposed methods IMD-IRMPC and IMD-K_2 both have better performance than PE-DD. This can be explained by noting that our methods use the readily available prior knowledge (polytopic model in (4)) and is combined with RMPC technique to provide asymptotic stability for all the unknown LTI system lying within a given set Ξ defined in (4) which is not considered by [10]. To display the stabilization performance more clearly,

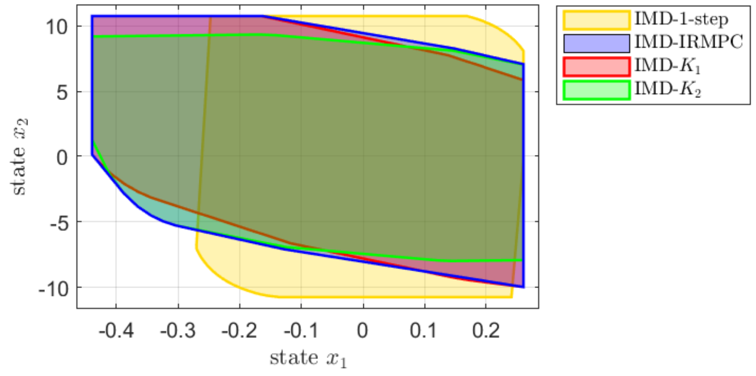


FIGURE 8 The comparison of feasible region between IMD-IRMPC, IMD-K_1, IMD-K_2 and IMD-1-step

TABLE 6 Normalized size of attractive region with respect to different direction for Example 2

Direction	IMD-IRMPC	IMD-K_1	IMD-K_2	IMD-1 step
x_1	1	1	1	0.757
x_2	0.964	0.964	0.796	1

the closed-loop trajectories in presence of model mismatch are plotted in Figure 7. It can be observed that IMD-IRMPC and IMD-K_2 converge faster than PE-DD.

5.2 | Example 2: CSTR

Consider the following Continuous Stirred Tank Reactor (CSTR) from [2] with the dynamics:

$$\dot{C}_A = \frac{q}{V}(C_{Af} - C_A) - k_0 e^{(\frac{-E}{RT})} C_A,$$
$$\dot{T} = \frac{q}{V}(T_f - T) + \frac{(-\Delta H)}{\rho C_p} k_0 e^{(\frac{-E}{RT})} C_A + \frac{UA}{V\rho C_p}(T_c - T),$$

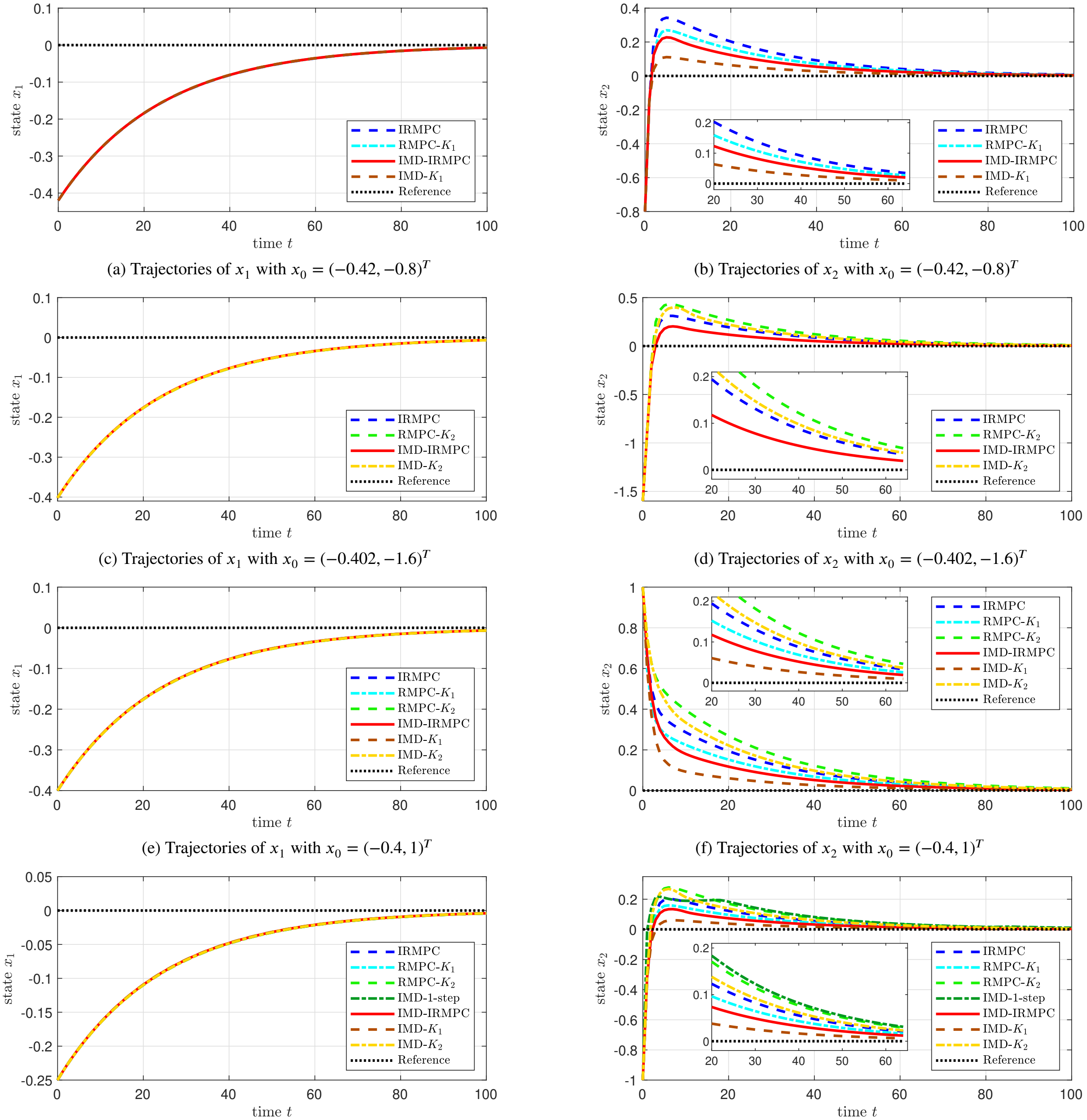
where two states C_A, T denote the concentration of reactant A and the reactor temperature respectively, control input T_c denotes the coolant temperature. All physical constants are listed in [2]. Constraints are $X = \{\begin{bmatrix} C_A \\ T \end{bmatrix} \in \mathbb{R}^2 | 0.3mol/L \leq C_A \leq 1mol/L, 370K \leq T \leq 392K\}, U = \{T_c \in \mathbb{R} | 250K \leq T_c \leq 360K\}$. The target unstable steady point is $x_s = (0.7393, 380.7463)^T$. We shift the origin, equivalently derive an LPV model from [25] and adopt the first-order discretization with the sampling time $T_s = 0.03min$. A time-varying polytopic system follows

$$x(k+1) = (A_0 + \Delta_A^{\mathrm{tr}})x(k) + (B_0 + \Delta_B^{\mathrm{tr}})u(k), [\Delta_A^{\mathrm{tr}}, \Delta_B^{\mathrm{tr}}] \in \Xi \triangleq Co\{[\Delta_A^{(1)}, \Delta_B^{(1)}], \cdots, [\Delta_A^{(4)}, \Delta_B^{(4)}]\},$$
$$A_0 = \begin{bmatrix} 0.9577 & -0.0005 \\ 2.4613 & 1.0302 \end{bmatrix}, B_0 = \begin{bmatrix} 0 \\ 0.03 \end{bmatrix}, \Delta_A^{(1)} = \begin{bmatrix} -0.0131 & 0 \\ 2.6208 & 0 \end{bmatrix}, \Delta_A^{(2)} = \begin{bmatrix} 0.0131 & 0 \\ -2.6208 & 0 \end{bmatrix},$$
$$\Delta_A^{(3)} = \begin{bmatrix} 0 & -0.0002 \\ 0 & 0.0471 \end{bmatrix}, \Delta_A^{(4)} = \begin{bmatrix} 0 & 0.0002 \\ 0 & -0.0471 \end{bmatrix}, \Delta_B^{(1)} = \Delta_B^{(2)} = \Delta_B^{(3)} = \Delta_B^{(4)} = 0.$$

By shifting origin, the constraints for the new defined states are

$$-0.4393 \leq x_1 \leq 0.2607, -10.7463 \leq x_2 \leq 10.7463, -60 \leq u \leq 50. \tag{128}$$

The weighting matrices are $Q = diag\{0.1, 40.5\}$, $R = 0.1$. The data length and control horizon are set to be $N_L = 2, N = 2$. To compute two stabilizing feedback gains with different characteristics, the cost indices and the reference points of the high-performance and low-performance feedback in (70) are respectively set as $\gamma_1 = 10000, \gamma_2 = 5300$ and $x_{p,1} = (-0.4 \;\; 9)^T$, $x_{p,2} = (-1 \;\; 191)^T$. By solving (70), the resulting high-performance and low-performance feedback controllers are respectively selected as $K_1 = [-60.7186 \;\; -8.7233]$ and $K_2 = [2.6129 \;\; -9.2618]$.



(a) Trajectories of x_1 with $x_0 = (-0.42, -0.8)^T$

(b) Trajectories of x_2 with $x_0 = (-0.42, -0.8)^T$

(c) Trajectories of x_1 with $x_0 = (-0.402, -1.6)^T$

(d) Trajectories of x_2 with $x_0 = (-0.402, -1.6)^T$

(e) Trajectories of x_1 with $x_0 = (-0.4, 1)^T$

(f) Trajectories of x_2 with $x_0 = (-0.4, 1)^T$

(g) Trajectories of x_1 with $x_0 = (-0.25, -1)^T$

(h) Trajectories of x_2 with $x_0 = (-0.25, -1)^T$

FIGURE 9 The state evolution under IMD-IRMPC, IMD-K_1, IMD-K_2, their robust counterparts IRMPC, RMPC-K_1, RMPC-K_2 and IMD-1-step

We compare the feasible region, performance and computation efficiency of our proposed data-driven methods (IMD-K_1, IMD-K_2, IMD-IRMPC), their robust counterparts (RMPC-K_1, RMPC-K_2, IRMPC) introduced in [6] and the IMD-1 step (the input mapping data-driven method with control horizon $N = 1$ in [22]) . To give a fair comparison, the control horizon in all those methods are set the same as $N = 2$. And the data length in IMD based methods are set as $N_L = 2$.

To compare the region of attraction, we display the attractive attraction for four IMD based methods in Figure 8. We can observe that IMD-IRMPC has a larger attractive region than IMD-K_1 and IMD-K_2 as stated in Remark 2. Compared with the IMD-1-step scehme in [22], the methods proposed in this paper has a wider feasibility along the negative x_1 direction due to

TABLE 7 Comparison of accumulated cost for Example 2 (- = infeasible)

Initial state x_0	IMD-IRMPC	IRMPC	IMD-K_1	RMPC-K_1	IMD-K_2	RMPC-K_2	IMD-1 step
$\left(-0.42\ \ -0.8\right)^T$	1398.7	1470.3	1355.6	1423.7	-	-	-
$\left(-0.402\ \ -1.6\right)^T$	1585.7	1652.4	-	-	1688.5	1740.2	-
$\left(-0.4\ \ 1\right)^T$	980.0	1036.8	941.8	995.7	1084.3	1128.8	-
$\left(-0.25\ \ -1\right)^T$	630.7	654.1	614.7	639.2	677.8	689.6	722.5

TABLE 8 Comparison of average and maximal computation time with an initial state $x_0 = (-0.25, -1)^T$ for Example 2

Algorithm	IMD-IRMPC	IRMPC	IMD-K_1	RMPC-K_1	IMD-K_2	RMPC-K_2	IMD-1-step
Average Time(s)	0.0088	0.0079	0.0071	0.0059	0.0069	0.0058	0.1288
Maximum Time(s)	0.0099	0.0089	0.0080	0.0067	0.0079	0.0070	0.1744

the use of polyhedral invariant set which can cope with asymmetric constraint of state x_1 defined in (128) in a non-conservative way than the ellipsoid set used in IMD-1-step. The normalized size of attractive region projected onto different axes is shown in Table 6 where the maximum taken for comparison is chosen as the range of physical constraint for each state components defined in (128).

To compare the closed-loop performance, we select four initial points, respectively $x_0 = (-0.42, -0.8)^T$, $(-0.402, -1.6)^T$, $(-0.4, 1)^T$, $(-0.25, -1)^T$. The results are reported as follows. Table 7 shows the comparison of accumulated cost with regard to different initial points for different algorithms. The cost is evaluated through $J_a = \sum_{k=0}^{T_{step}} (x^T(k)Qx(k) + u^T(k)Ru(k))$ where $T_{step} = 120$ denotes the simulation time. It can be observed that IMD-IRMPC always provides a feasible solution for different initial points which cannot be achieved for methods based on only one feedback gain. It is also shown that IMD-IRMPC improves the performance of IRMPC, which also applies for IMD-K_1 and IMD-K_2 compared with their robust counterparts. Moreover, it clearly shows the advantages of the proposed multi-step IMD schemes of IMD-IRMPC, IMD-K_1 and IMD-K_2 over the IMD-1-step scheme due to the increased degrees of freedom. This clearly demonstrates the effectiveness of the proposed muti-step IMD approach. To display the stabilization performance more clearly, the closed-loop trajectories are plotted in Figure 9. It explicitly shows that IMD-IRMPC, IMD-K_1 and IMD-K_2 show better stabilization performance than other methods.

To compare computation efficiency, Table 8 shows the computation time. It shows that the proposed IMD based methods IMD-IRMPC, IMD-K_1 and IMD-K_2 consume a few more computational resources than their robust counterparts IRMPC, RMPC-K_1 and RMPC-K_2 due to the increase of optimization variables. In addition, IMD-IRMPC adds a few computation burden to IMD-K_1 and IMD-K_2 due to the use of interpolation control and still has good computation efficiency. It can also be shown that the multi-step IMD schemes IMD-IRMPC, IMD-K_1 and IMD-K_2 have less computation burden than IMD-1-step. This results from the fact that only a quadratic program is required to be solved online rather than a semi-definite program in IMD-1-step. We code our program on Matlab 2018a and run our code on Intel Core I7 with 3.30GHz. The optimization is solved by Mosek.

6 | CONCLUSION

This paper proposes two new multi-step input-mapping data-driven MPC methods for constrained control of unknown linear time invariant systems. By mapping the current and future input to the past online input state measurements, the future state can be expressed as a linear combination of past trajectories. This multi-step IMD scheme is respectively combined with two RMPC techniques using polyhedral invariant set. One is the RMPC with a fixed feedback gain and an IMD-RMPC is proposed. The other is the interpolation RMPC and an IMD-IRMPC is developed. The interpolation RMPC technique can combine the advantages (performance, size and shape of feasible region) of different feedback gains where compromise is a necessity for the RMPC technique with a single fixed feedback gain. The IMD-IRMPC has larger feasible region than IMD-RMPC. The recursive feasibility and stability of the resulting two methods are ensured. By incorporating this multi-step IMD scheme, the conservatism of two RMPC techniques introduced in [6] can be reduced, resulting in a less conservative performance. A comparison simulation with the AMPC method in [8] shows that the proposed two algorithms can achieve superior performance with higher computation efficiency. Meanwhile, compared to the PE-DD approach in [10,12], our proposed two schemes can deal with model mismatch (multiplicative uncertainty) which is not considered by [10,12]. No initially measured trajectories are required in our proposed methods as in [10,12]. Compared with the previous work in [22], the IMD-RMPC and IMD-IRMPC methods can deal with asymmetric constraint and have higher computation efficiency by using polyhedral invariant set. Performance can be improved by introducing more degrees of freedom through multi-step paradigm.

ACKNOWLEDGMENTS

This work was partially supported by the National Science Foundation of China (grant no. 61973214, 61963030, 62103271).

CONFLICT OF INTEREST

The authors have no conflict of interest to disclose.

DATA AVAILABILITY STATEMENT

There is no data available to be shared.

References

1. Kouvaritakis B, Cannon M. Model predictive control. *Switzerland: Springer International Publishing* 2016.

2. Wan Z, Kothare MV. Efficient scheduled stabilizing output feedback model predictive control for constrained nonlinear systems. *IEEE Transactions on Automatic Control* 2004; 49(7): 1172–1177.

3. Fleming J, Kouvaritakis B, Cannon M. Robust Tube MPC for Linear Systems With Multiplicative Uncertainty. *IEEE Transactions on Automatic Control* 2015; 60(4): 1087-1092.

4. Hanema J, Lazar M, Tóth R. Heterogeneously parameterized tube model predictive control for LPV systems. *Automatica* 2020; 111: 108622.

5. Liu C, Li H, Shi Y, Xu D. Codesign of Event Trigger and Feedback Policy in Robust Model Predictive Control. *IEEE Transactions on Automatic Control* 2020; 65(1): 302-309.

6. Li D, Xi Y. The synthesis of robust model predictive control with QP formulation. *International Journal of Modelling, Identification and Control* 2011; 13(1-2): 1–8.

7. Nguyen HN, Olaru S, Gutman PO, Hovd M. Constrained Control of Uncertain, Time-varying Linear Discrete-Time Systems Subject to Bounded Disturbances. *IEEE Transactions on Automatic Control* 2015; 60(3): 831-836.

8. Zhang K, Shi Y. Adaptive model predictive control for a class of constrained linear systems with parametric uncertainties. *Automatica* 2020; 117: 108974.

9. Garg A, Mhaskar P. Subspace identification and predictive control of batch particulate processes. In: Proceedings of American Control Conference. ; 2017: 505–510.

10. Berberich J, Köhler J, Müller MA, Allgöwer F. Data-Driven Model Predictive Control With Stability and Robustness Guarantees. *IEEE Transactions on Automatic Control* 2021; 66(4): 1702-1717.

11. De Persis C, Tesi P. Formulas for Data-Driven Control: Stabilization, Optimality, and Robustness. *IEEE Transactions on Automatic Control* 2020; 65(3): 909-924.

12. Coulson J, Lygeros J, Dörfler F. Data-enabled predictive control: In the shallows of the DeePC. In: Proceedings of European Control Conference. ; 2019: 307–312.

13. Elokda E, Coulson J, Beuchat PN, Lygeros J, Dörfler F. Data-enabled predictive control for quadcopters. *International Journal of Robust and Nonlinear Control* 2021; 31(18): 8916-8936.

14. Willems JC, Rapisarda P, Markovsky I, De Moor BL. A note on persistency of excitation. *Systems & Control Letters* 2005; 54(4): 325–329.

15. Prívara S, Cigler J, Váňa Z, Ferkl L, Šebek M. Subspace identification of poorly excited industrial systems. In: Proceedings of Conference on Decision and Control. ; 2010: 4405-4410.

16. Manzano JM, Peña M. d. lD, Calliess J, Limon D. Online learning constrained model predictive control based on double prediction. *International Journal of Robust and Nonlinear Control* 2021; 31(18): 8813-8829.

17. Terzi E, Fagiano L, Farina M, Scattolini R. Learning-based predictive control for linear systems: A unitary approach. *Automatica* 2019; 108: 108473.

18. Lorenzen M, Cannon M, Allgöwer F. Robust MPC with recursive model update. *Automatica* 2019; 103: 461–471.

19. Lu X, Cannon M, Koksal-Rivet D. Robust adaptive model predictive control: Performance and parameter estimation. *International Journal of Robust and Nonlinear Control* 2021; 31(18): 8703–8724.

20. Köhler J, Kötting P, Soloperto R, Allgöwer F, Müller MA. A robust adaptive model predictive control framework for nonlinear uncertain systems. *International Journal of Robust and Nonlinear Control* 2021; 31(18): 8725–8749.

21. Di Cairano S, Danielson C. Indirect adaptive model predictive control and its application to uncertain linear systems. *International Journal of Robust and Nonlinear Control* 2021; 31(18): 8678-8702.

22. Yang L, Lu J, Xu Y, Li D, Xi Y. Constrained robust model predictive control embedded with a new data-driven technique. *IET Control Theory & Applications* 2020; 14(16): 2395–2405.

23. He S, Chen W, Li D, Xi Y, Xu Y, Zheng P. Iterative Learning Control With Data-Driven-Based Compensation. *IEEE Transactions on Cybernetics* 2021: 1-12.

24. Pluymers B, Rossiter JA, Suykens JA, De Moor B. The efficient computation of polyhedral invariant sets for linear systems with polytopic uncertainty. In: Proceedings of American Control Conference. ; 2005: 804–809.

25. Ding B, Xie L, Cai W. Robust MPC for polytopic uncertain systems with time-varying delays. *International Journal of Control* 2008; 81(8): 1239–1252.