Robust Hierarchical MPC for Handling Long Horizon Demand Forecast Uncertainty with Application to Automotive Thermal Management*

Mohammad Reza Amini¹, Ilya Kolmanovsky², and Jing Sun¹

Abstract—This paper presents a robust hierarchical MPC (H-MPC) for dynamic systems with slow states subject to demand forecast uncertainty. The H-MPC has two layers: (i) the scheduling MPC at the upper layer with a relatively long prediction/planning horizon and slow update rate, and (ii) the piloting MPC at the lower layer over a shorter prediction horizon with a faster update rate. The scheduling layer MPC calculates the optimal slow states, which will be tracked by the piloting MPC, while enforcing the system constraints according to a long-range and approximate prediction of the future demand/load, e.g., traction power demand for driving a vehicle. In this paper, to enhance the H-MPC robustness against the long-term demand forecast uncertainty, we propose to use the high-quality preview information enabled by the connectivity technology over the short horizon to modify the planned trajectories via a constraint tightening approach at the scheduling layer. Simulation results are presented for a simplified vehicle model to confirm the effectiveness of the proposed robust H-MPC framework in handling demand forecast uncertainty.

I. INTRODUCTION

For dynamic systems with multiple time scales, centralized optimization could lead to non-uniform performance as a centralized controller inherently assumes that all the system outputs are equally reactive [1]. However, the outputs of such dynamic systems tend to react and respond at different rates. This is a common observation for dynamic systems with slow and storage states, e.g., microgrids [2], vehicle [3]–[6] and aircraft [7] thermal management systems.

One approach to address the aforementioned issue in a centralized scheme is to select a relatively long prediction horizon to ensure satisfactory performance of the slow states. While this approach could fulfill the overall performance requirements, it has several significant issues for practical applications:

- Firstly, it will significantly increase the computational complexity. For fast states, the computational complexity is often not justifiable as the information downstream of the horizon has minimal influence on the optimal evolution of the fast state trajectories.
- Secondly, for most systems operating in a dynamic environment, accurate prediction of the demand/load profile over a long prediction horizon is not feasible. For example, vehicle speed cannot be predicted accurately

over a long prediction horizon based on vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) information [4], [5].

The common approach exploited in the literature [6], [8], [9] for systems with multi-timescales is the hierarchical control with multiple layers. The hierarchical framework allows for optimization over different planning horizons at each layer based on the requirements of different system outputs. It also helps to significantly reduce the computational cost compared to its centralized counterpart [4], [6], [10].

Whether centralized or hierarchical, the uncertainty associated with the long-term prediction of the demand/load is the common problem for systems with slow dynamics. As the optimization often pushes the system to work at its limit for best efficiency, it is important to be able to enforce the constraints robustly in the presence of demand forecast uncertainty. This is because for slow states, once the violation of the constraint occurs, the controller has to put more efforts to compensate for this violation, thereby reducing the overall system efficiency. Unlike the most common techniques in the literature on the robust design of hierarchical and distributed MPC [6], [9], [11] which mainly focus on robustifying the controller under the presence of external unknown disturbances, the robustness to long horizon demand forecast uncertainty has not been fully studied.

From a new perspective, in this paper, we consider the long horizon demand forecast uncertainty challenge which plays a critical role in planning and optimization of slow/storage states with wide application in vehicle and aircraft thermal management, building HVAC control, and power grid optimization. To address this challenge, we exploit the H-MPC optimization framework as the hierarchical architecture provides a unique capability of using the demand forecast with different accuracies over different prediction horizons. Our proposed H-MPC has two layers as scheduling and piloting layers. The scheduling layer at the top computes the optimal control input at a slow update rate over a relatively long prediction horizon. The output of the scheduling layer is the optimal trajectories of the "slow" states of interest according to an approximate prediction of the demand over the long planning horizon. In the lower layer, piloting layer, over a much shorter prediction horizon and using a faster update rate, an MPC is designed to follow the scheduled slow state trajectories coming from the upper layer while fulfilling the overall system performance requirements.

It is envisioned that over a relatively short prediction horizon, the demand/load profile can be accurately predicted. As an example for connected and automated vehicles, the

^{*}This paper is based upon the work supported by the United States Department of Energy (DOE) under award No. de-ar0000797.

¹M.R. Amini and J. Sun are with the Department of Naval Architecture & Marine Engineering, University of Michigan, Ann Arbor, MI 48109 USA. Emails: {mamini, jingsun}@umich.edu

²I. Kolmanovsky is with the Dept. of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109 USA. Email: ilya@umich.edu

V2V/V2I communications can be used to predict the future vehicle speed and the associated traction power demand over a short time window accurately [12]. We propose to use this short horizon accurate demand preview information to predict the system state evolution over the prediction horizon of the piloting layer MPC while tracking the planned trajectory coming from the scheduling layer MPC. Computing the evolution of the system state with a faster update rate when the accurate demand preview is accessible allows for detecting the constraint violation proactively over the short prediction horizon. Particularly, we can predict the deviation of the slow state from the planned trajectory and realize the possibility of constraint violation over the short prediction horizon. It is noted that, assuming a perfect model, the deviation in the reference tracking and constraint violation at the piloting layer are mainly attributed to the mismatch between the actual demand and the approximate preview incorporated for planning at the scheduling layer. Once these deviations and possible constraint violation are predicted over the shorter prediction horizon, the constraint set at the scheduling layer is re-computed and tightened to make the H-MPC prepared for the upcoming situation during which the chances of constraint violation are high.

The main contribution of this paper is to develop a robust H-MPC for a class of dynamic systems with slow states subject to demand preview uncertainty. We propose to utilize short and long-range demand preview information to compute robust constraint set to enhance the conventional H-MPC robustness against the demand preview uncertainty while improving the overall system efficiency. The performance of the proposed robust H-MPC is studied through simulations for a vehicle thermal management case study.

II. HIERARCHICAL MPC FORMULATION

We consider the following discrete-time linear timeinvariant system,

$$\boldsymbol{x}_{k+1} = A\boldsymbol{x}_k + B_1\tilde{\boldsymbol{u}}_k + B_2\hat{\boldsymbol{u}}_k, \qquad (1)$$

where $\boldsymbol{x} = [x_1, \dots, x_n]^{\mathsf{T}} \in \mathbb{R}^n$, $\tilde{\boldsymbol{u}} \in \mathbb{R}^m$, and $\hat{\boldsymbol{u}} \in \mathbb{R}^h$ are the system state, control input, and demand/load on the system, respectively, and $A \in \mathbb{R}^{n \times n}$, $B_1 \in \mathbb{R}^{n \times m}$, and $B_2 \in \mathbb{R}^{n \times h}$. The integer k denotes the time step and the sampling period of (1) is assumed to be T.

We further assume that different states of the system $(x_{i_x}, i_x = 1, \dots, n)$ respond at different rates. Depending on the relative time scales of different states, they can be categorized into "fast" and "slow" states: $\boldsymbol{x} = [(\boldsymbol{x^{fast}})^{\mathsf{T}}, (\boldsymbol{x^{slow}})^{\mathsf{T}}]^{\mathsf{T}}$. Note that $\tilde{\boldsymbol{u}}$ contains the physical control inputs to the system that are optimized to regulate the overall performance. On the other hand, $\hat{\boldsymbol{u}}$ is the vector of non-adjustable inputs to the system representing measured external loads. Since $\hat{\boldsymbol{u}}$ is a measured input to the system, it affects the system states, and the knowledge about $\hat{\boldsymbol{u}}$ can be incorporated as a preview for optimization of $\tilde{\boldsymbol{u}}$. In automotive applications, traction power demand (P_{trac}) for driving a vehicle is an example of $\hat{\boldsymbol{u}}$.

The state and control input of the linear system in (1) are constrained: $x \in \mathcal{X}$, $\tilde{u} \in \mathcal{U}$, where $\mathcal{X} = \{x_k | P_x x_k \leq q_x\}$ and $\mathcal{U} = \{ \tilde{\boldsymbol{u}}_k | P_u \tilde{\boldsymbol{u}}_k \leq \boldsymbol{q}_u \}$ are polyhedral sets.

A. Single-Layer MPC (S-MPC) Formulation

For the system (1), we first consider a centralized singlelayer MPC (S-MPC) defined according to the following finite-horizon optimization problem:

$$\min_{\tilde{\boldsymbol{U}}_{k}} \sum_{j=0}^{N} \ell_{\text{S-MPC}}(k+j|k),$$
s.t. $\boldsymbol{x}(k+j+1|k) = A\boldsymbol{x}(k+j|k) + B_{1}\tilde{\boldsymbol{u}}(k+j|k) + B_{2}\hat{\boldsymbol{u}}(k+j|k),$
 $\boldsymbol{x}(k+j|k) \in \mathcal{X}, \ \tilde{\boldsymbol{u}}(k+j|k) \in \mathcal{U},$
 $\boldsymbol{x}(k|k) = \boldsymbol{x}_{k},$
(2)

where $\ell_{\text{S-MPC}}$ is the S-MPC stage cost, $\tilde{U}_k = [\tilde{u}(k|k)^{\mathsf{T}}, \cdots, \tilde{u}(k+N-1|k)^{\mathsf{T}}]^{\mathsf{T}}$ is the control input sequence, and $\hat{U}_k = [\hat{u}(k|k)^{\mathsf{T}}, \cdots, \hat{u}(k+N-1|k)^{\mathsf{T}}]^{\mathsf{T}}$ is the sequence of the "predicted" demand preview. The cost function $\ell_{\text{S-MPC}}$ is defined as [13]:

$$\ell_{\text{S-MPC}} = \left\| \boldsymbol{r}(k+j|k) - \boldsymbol{r}^d(k+j) \right\|_{\Lambda}^2$$
(3)

where $\mathbf{r}_k = E\mathbf{x}_k + F\tilde{\mathbf{u}}_k$, and $\mathbf{r}^d \in \mathbb{R}^{n_r}$ represents the desired operation of the system. When using the solution of the optimization problem (2) to control the system (1), the MPC feedback law defined by $\tilde{\mathbf{u}}(k|k)$ inherently assumes that all the system outputs are equally reactive [1], however, in the case considered here, the outputs of the system respond over different time scales.

B. Hierarchical MPC (H-MPC) Formulation

The H-MPC considered in this paper has two layers [9]:

- scheduling layer at the top with an MPC that computes the optimal control inputs and the "slow" state trajectories at a slower update rate of $T_s = \nu T$, where ν is a positive integer, over a relatively long prediction horizon (H_s). An approximate prediction of the demand sequence is incorporated at this layer.
- **piloting layer** with an MPC implemented over a much shorter prediction horizon (H_p) and using a faster update rate (T) to track the planned trajectories of the slow states from the scheduling layer. The piloting layer MPC has access to a more accurate prediction of the demand profile over H_p .

The scheduling layer MPC is based on the following finitehorizon optimization problem:

$$\begin{split} \min_{\hat{U}_{k_{s}}^{s}} & \sum_{j=0}^{H_{s}} \ell_{\text{scheduling}}(k_{s}+j|k_{s}), \\ \text{s.t.} & \boldsymbol{x}^{s}(k_{s}+j+1|k_{s}) = A^{s}\boldsymbol{x}^{s}(k_{s}+j|k_{s}) \\ & + B_{1}^{s}\tilde{\boldsymbol{u}}_{s}(k_{s}+j|k_{s}) + B_{2}^{s}\hat{\boldsymbol{u}}_{s}(k_{s}+j|k_{s}), \\ & \boldsymbol{x}^{s}(k_{s}+j|k_{s}) \in \mathcal{X}, \ \tilde{\boldsymbol{u}}^{s}(k_{s}+j|k_{s}) \in \mathcal{U}, \\ & \boldsymbol{x}^{s}(k_{s}|k_{s}) = \boldsymbol{x}_{k_{s}}^{s}, \end{split}$$
(4)

where \boldsymbol{x}^s and $\boldsymbol{U}_{k_s}^s = [\boldsymbol{\tilde{u}}^s(k_s|k_s)^{\mathsf{T}}, \cdots, \boldsymbol{\tilde{u}}^s(k_s+H_s-1|k_s)^{\mathsf{T}}]^{\mathsf{T}}$ are the state vector and control input sequence updated at the slower sampling time of $T_s = \nu T$, which is indexed by k_s . The sequence of the approximate demand profile $\hat{U}_{k_s}^s = [\hat{u}^s(k_s|k_s)^{\intercal}, \cdots, \hat{u}^s(k_s + H_s - 1|k_s)^{\intercal}]^{\intercal}$ is also incorporated at the scheduling layer as the preview. The stage cost function $(\ell_{\text{scheduling}})$ of (4) is the same as the one considered in (3) while it is updated every T_s second. The prediction model in (4) is based on the down-sampled version of (1) with slower sampling rate of T_s as follows:

$$A^{s} = (A)^{\nu}, \quad B_{1}^{s} = \sum_{j=0}^{\nu-1} (A)^{j} B_{1}, \quad B_{2}^{s} = \sum_{j=0}^{\nu-1} (A)^{j} B_{2}, \quad (5)$$

Once the scheduling layer optimization problem is solved, based on the computed control sequence $U_{k_s}^s$, the sequence of the scheduled slow states at the subsequent time steps over the horizon of the piloting layer MPC (H_p) can be calculated: $*x^{\text{slow}}(k_s + 1|k_s), \dots, *x^{\text{slow}}(k_s + H_p|k_s)$, where $*x^{\text{slow}}$ is the vector of scheduled slow states of interest. Note that unlike the approach presented in [9], [13], the horizon of the second layer, piloting layer (H_p) , can be longer than $H_p > \nu$. In the next step, the sequence of $*x^{\text{slow}}$ is passed to the piloting layer as a piecewise constant function to account for the faster sampling rate of the lower layer controller (T):

$${}^{*}\boldsymbol{X}^{\mathrm{slow}}(k|k_{s}) = [{}^{*}\boldsymbol{x}^{\mathrm{slow}}(k_{s}|k_{s})^{\mathsf{T}}, {}^{*}\boldsymbol{x}^{\mathrm{slow}}(k+1|k_{s})^{\mathsf{T}}, \cdots, \\ {}^{*}\boldsymbol{x}^{\mathrm{slow}}(k+\nu-1|k_{s})^{\mathsf{T}}, {}^{*}\boldsymbol{x}^{\mathrm{slow}}(k_{s}+1|k_{s})^{\mathsf{T}}, \cdots, \\ {}^{*}\boldsymbol{x}^{\mathrm{slow}}(k+2\nu-1|k_{s})^{\mathsf{T}}, \cdots, {}^{*}\boldsymbol{x}^{\mathrm{slow}}(k+H_{p}-1|k_{s})^{\mathsf{T}}]^{\mathsf{T}}$$
(6)

where * X^{slow} is the sequence of the scheduled slow states to be tracked at the piloting layer. Note that the sequence of * X^{slow} between the slow updates are assumed to be constant, e.g., * $x^{\text{slow}}(k_s|k_s) = *x^{\text{slow}}(k+1|k_s) = \cdots = *x^{\text{slow}}(k+\nu-1|k_s)$. The piloting layer MPC is based on the solution of the following optimal control problem:

$$\min_{\tilde{\boldsymbol{U}}_{k}} \sum_{j=0}^{H_{p}} \ell_{\text{piloting}}(k+j|k),$$
s.t. $\boldsymbol{x}(k+j+1|k) = A\boldsymbol{x}(k+j|k) + B_{1}\tilde{\boldsymbol{u}}(k+j|k) + B_{2}\hat{\boldsymbol{u}}(k+j|k),$
 $\boldsymbol{x}(k+j|k) \in \mathcal{X}, \ \tilde{\boldsymbol{u}}(k+j|k) \in \mathcal{U},$
 $\boldsymbol{x}(k|k) = \boldsymbol{x}_{k}.$
(7)

where the stage cost ℓ_{piloting} is a combination of the original cost in (3) and extra tracking terms to enforce the slow states $(\boldsymbol{x^{\text{slow}}})$ to track the scheduled slow state trajectories $({}^*\boldsymbol{x^{\text{slow}}})$ presented in (6). $\tilde{\boldsymbol{U}}_k = [\tilde{\boldsymbol{u}}(k|k)^{\intercal}, \cdots, \tilde{\boldsymbol{u}}(k+H_p-1|k)^{\intercal}]^{\intercal}$ is the control input sequence of the piloting layer MPC. It is assumed that the incorporated demand preview at the piloting layer $(\hat{\boldsymbol{U}}_k = [\hat{\boldsymbol{u}}(k|k)^{\intercal}, \cdots, \hat{\boldsymbol{u}}(k+H_p-1|k)^{\intercal}]^{\intercal}$ is more accurate than the demand preview used at the scheduling layer $(\hat{\boldsymbol{U}}_{k_s}^s)$. When implementing the H-MPC, at the end of each optimization iteration, $\tilde{\boldsymbol{u}}(k|k)$ is commanded to the system and the horizon is shifted by one time step (T).

III. ROBUST HIERARCHICAL MPC

An intuitive approach for improving the constraint satisfaction at the piloting layer is to tighten the constraints at the scheduling layer. If the constraint tightening is done conservatively, the overall efficiency of the system may be degraded. To avoid unnecessary conservatism, in this paper, we propose to compute a robust constraint set (denoted by $\mathcal{X}^{\text{robust}}$) at each time step k. With a faster update rate and more accurate incorporated demand preview, the evolution of the slow state trajectories over the short horizon of the piloting layer MPC can be predicted while tracking the planned trajectories from the scheduling layer. This prediction of the slow state evolution is then used to compute $\mathcal{X}^{\text{robust}}$ via estimating the deviation in tracking the planned slow state trajectory and potential violation of the constraint over H_p . This concept is shown in Fig. 1.



Fig. 1. Schematic of the constraint violation prediction to compute \mathcal{X}^{robust} . At time step k, the scheduling layer MPC plans the slow state trajectories (b) based on an approximate knowledge of the demand preview (a). The piloting layer MPC is designed to not only regulate the fast states but track the planned slow state trajectory while accessing the accurate demand preview at a faster update rate. The evolution of the slow states over the short prediction horizon is computed (d). If any deviation or constraint violation is predicted over H_p , the scheduling layer constraint set is tightened proactively.

According to the accurate demand preview $(\hat{U}_k = [\hat{u}(k|k)^{\intercal}, \cdots, \hat{u}(k+H_p-1|k)^{\intercal}]^{\intercal})$, the evolution of the system states x over the piloting layer horizon H_p can be predicted based on (1):

$$\boldsymbol{x}(k+j+1|k) = A\boldsymbol{x}(k+j|k) + B_1 \hat{\boldsymbol{u}}^*(k+j|k) + B_2 \hat{\boldsymbol{u}}(k+j|k)$$
(8)

where $j = 0, \dots, H_p - 1$ and $\tilde{u}^*(k+j|k)$ is the vector of the control inputs at k+j computed based on the piloting layer optimal control problem (7) solution at time step k. Based on the state predictions (8), the constraint violation of the slow states of interest can also be predicted at subsequent time steps, which will be used to tighten the state constraint set for the optimization problem at the scheduling layer. We denote the tightened constraint set over the piloting layer horizon at k + j + 1, $j = 0, \dots, H_p - 1$, by \mathcal{X}_j , which is computed as follows:

$$\mathcal{X}_{j} = \{ \boldsymbol{x}(k|k) | \qquad (9)$$
$$P_{x}\boldsymbol{x}(k+j+1|k) \leq (\boldsymbol{q}_{x} - \bar{\mathbf{q}}_{x}(k+j+1|k)) \},$$

where $q_x = [q_x^1, \dots, q_x^{n_q}]^{\mathsf{T}} \in \mathbb{R}^{n_q}$ represents the state constraints, and is used in the definition of the original state constraint set $\mathcal{X} = \{x_k | P_x x_k \leq q_x\}$. Note that if we assume q_x represents the upper and lower bounds on each state and each row of P_x corresponds to one constraint, then non-

zero elements of P_x are either 1 (upper limit) or -1 (lower limit). We then define $\bar{\mathbf{q}}_x(k+j+1|k)$ in (9) as the vector of predicted constraint violation at k+j+1 based on the state prediction made at k (8). If $q_x^{i_q}$ ($i_q = 1, \dots, n_q$) corresponds to x_{i_x} ($i_x = 1, \dots, n$), then the i_q^{th} element of $\bar{\mathbf{q}}_x$ at k+j+1, i.e., $\bar{q}_x^{i_q}(k+j+1|k)$, is calculated as follows:

$$\vec{q}_{x}^{i_{q}}(k+j+1|k) =$$

$$\begin{cases} 0 \quad \text{if } x_{i_{x}} \text{ is fast} \\ 0 \quad \text{if } x_{i_{x}}(k+j+1|k) < q_{x}^{i_{q}} \& x_{i_{x}} \text{ is slow} \\ x_{i_{x}}(k+j+1|k) - q_{x}^{i_{q}} \\ \text{if } x_{i_{x}}(k+j+1|k) \ge q_{x}^{i_{q}} \& x_{i_{x}} \text{ is slow} \end{cases}$$

$$(10)$$

Note that since we are interested in the slow states constraint enforcement, the elements of $\bar{\mathbf{q}}_x$ corresponding to fast states are set to zero in (10).

Eventually and based on the computed \mathcal{X}_j s in (9), the robust constraint set ($\mathcal{X}^{\text{robust}}$) to be implemented at the next time step k + 1 at the scheduling layer is computed as:

$$\mathcal{X}^{\text{robust}} = \mathcal{X} \cap \mathcal{X}_1 \cap \dots \cap \mathcal{X}_{k+H_p-1}, \quad (11)$$

based on which, the scheduling layer MPC of the robust H-MPC is re-formulated as follows:

$$\begin{array}{ll}
\min_{\tilde{\boldsymbol{U}}_{k_{s}}^{s}} & \sum_{j=0}^{H_{s}} \ell_{\text{scheduling}}(k_{s}+j|k_{s}), \\
\text{s.t.} & \boldsymbol{x}^{s}(k_{s}+j+1|k_{s}) = A^{s}\boldsymbol{x}^{s}(k_{s}+j|k_{s}) \\
& + B_{1}^{s}\tilde{\boldsymbol{u}}_{s}(k_{s}+j|k_{s}) + B_{2}^{s}\hat{\boldsymbol{u}}_{s}(k_{s}+j|k_{s}), \\
& \boldsymbol{x}^{s}(k_{s}+j|k_{s}) \in \mathcal{X}^{\text{robust}}, \\
& \tilde{\boldsymbol{u}}^{s}(k_{s}+j|k_{s}) \in \mathcal{U}, \ \boldsymbol{x}^{s}(k_{s}|k_{s}) = \boldsymbol{x}_{k_{s}}^{s},
\end{array} \tag{12}$$

Remark: Note that one can propose to tighten the constraint whenever a violation of the constraint is detected at the piloting layer at time step k. This event-triggered "passive" constraint tightening approach can be incorporated in the H-MPC at the scheduling layer (4) by replacing the original constraint set \mathcal{X} with $\mathcal{X}^{\text{passive}}$, which is computed as follows:

$$\mathcal{X}^{\text{passive}} = \{ \boldsymbol{x}(k|k) | P_x \boldsymbol{x}(k|k) \le (q_x - \bar{\mathbf{q}}_x(k|k)) \}, \quad (13)$$

where $\bar{\mathbf{q}}_x(k|k)$ is the vector of measured (detected) constraint violation at each time step k. While $\mathcal{X}^{\text{passive}}$ is easy to compute, it leads to marginal improvement in the H-MPC robustness, see [4] for an example. This can be explained with respect to the slow dynamic characteristics which call for a long prediction horizon with enough lead time to ensure an effective constraint enforcement. In other words, if the H-MPC constraint set is tightened based on (13) after the occurrence of the violation, compensating for the occurred violation in the subsequent time steps may not be effective as it takes extra effort from the controller, potentially reducing the system efficiency.

IV. CASE STUDY: VEHICLE THERMAL MANAGEMENT

A simplified model of a vehicle is considered in this section to demonstrate the effectiveness of the proposed robust H-MPC. The main structure of the model is adopted from [13] and a new state has been added to the model to

represent the slow state. The model has four states, including the position of the vehicle (x_1) , vehicle speed (x_2) , onboard stored energy (x_3) , e.g., battery state of charge, and thermal index of the vehicle energy storage (x_4) , e.g., battery temperature. Compared to the first three states, the fourth state has a relatively slower dynamics. The control inputs to the model are acceleration (\tilde{u}_1) , deceleration (\tilde{u}_2) , and the power consumed for thermal management of the energy storage (\tilde{u}_3) . The power to an external load (\hat{u}) should also be delivered by the energy storage. The power demand is assumed to be non-adjustable, e.g., an auxiliary load on the battery to fulfill a demand within the vehicle.

The simplified vehicle model has the following structure:

$$\boldsymbol{x}_{k+1} = \begin{bmatrix} 1 & 1 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1 \end{bmatrix} \boldsymbol{x}_{k}$$
(14)
+
$$\begin{bmatrix} 1 & 1 & 0\\ 1 & -1 & 0\\ -0.8 & 0.8 & -0.15\\ 1 & 1 & -0.85 \end{bmatrix} \tilde{\boldsymbol{u}}_{k} + \begin{bmatrix} 0\\ 0\\ -0.25\\ 1 \end{bmatrix} \hat{\boldsymbol{u}}_{k}$$

where $\boldsymbol{x} = [x_1, x_2, x_3, x_4]^{\mathsf{T}}$ and $\tilde{\boldsymbol{u}} = [\tilde{u}_1, \tilde{u}_2, \tilde{u}_3]^{\mathsf{T}}$. The controller design objective is to track a given desired position (x_1^d) while enforcing the state and input constraints. Specifically, the thermal index of the energy storage (x_4) is preferred to be maintained below its upper optimum operation limit x_4^{UL} . It is noted that while the vehicle tracks the reference position, the energy storage needs to deliver power to satisfy the external load (\hat{u}) . Moreover, operating the vehicle by adjusting $\tilde{\boldsymbol{u}}$ and responding to $\hat{\boldsymbol{u}}$ consumes the on-board vehicle energy. The operation of the vehicle is required to be optimized such that the on-board energy lasts until the end of the vehicle mission. The state and input constraints are listed below:

$$[-1, -20, 0, 0]^{\mathsf{T}} \le \boldsymbol{x}_k \le [100, 20, 100, 30]^{\mathsf{T}}$$
(15)

$$[-1, -1, -1]^{\mathsf{T}} \le \tilde{\boldsymbol{u}}_k \le [1, 1, 1]^{\mathsf{T}}.$$
 (16)

In practice, the exact demand preview is assumed to be unknown over the entire vehicle mission, however, an approximate knowledge of the demand is available to the vehicle. Moreover, we assume the vehicle is connected to a server which can provide short-range and accurate demand predictions to the vehicle. The actual and approximate profiles of the power demand as the load (\hat{u}) on the on-board energy storage are shown in Fig. 1-(a,c).

In order to evaluate the H-MPC performance for the vehicle model (14), first a centralized single-layer MPC (S-MPC) is designed. To this end, we consider an S-MPC formulated over a finite-horizon (N) with $\tilde{u}_1, \tilde{u}_2, \tilde{u}_3$ being the optimization variables according to (2). The stage cost of S-MPC, $\ell_{\text{S-MPC}}$, is defined as follows:

$$\ell_{\text{S-MPC}} = \lambda_1 \tilde{u}_1^2 + \lambda_2 \tilde{u}_2^2 + \lambda_3 (x_1 - x_1^d)^2$$
(17)

where x_1^d is the desired and known position trajectory. \mathcal{X} and \mathcal{U} are the convex sets properly defined according to (15) and (16), respectively. The S-MPC optimization problem is solved at every time step, then the horizon is shifted by one

step ($T = 1 \ sec$), and only the current control is commanded to the system. The S-MPC simulation is carried out on a desktop computer, with an Intel[®] Core i7@2.60 GHz processor, in MATLAB[®]/SIMULINK[®] using YALMIP [14] for formulating the optimization problem, and IPOPT [15] for solving the optimization problem numerically. Additionally, in order to ensure the feasibility of the S-MPC solution, slack optimization variables are used to soften the constraints on the position (x_1) and thermal index (x_4). In all simulated cases, a very large weighting factor (e.g., 10^6) has been considered for the slack variable term in the MPC stage cost to ensure a comparable performance.

Fig. 2 shows the performing of the S-MPC with short $(N = 10 \ (10 \ sec))$ and long (N = 100) prediction horizons. Under the unrealistic assumption that the controller has the exact knowledge of the demand profile shown in Fig. 1-(c), Fig. 2-(a) shows that the S-MPC with long horizon tracks the desired position accurately with no x_1 constraint violation while enforcing the constraint on the thermal index state x_4 . When the shorter horizon is used, the S-MPC cannot mitigate the violations of x_4 constraint; thus, it compromises x_1 tracking (Fig. 2-(b)) to avoid further increase in the thermal index. The reason for this deviation in the S-MPC performance with shorter prediction horizon is that the controller does not have enough lead time to take proactive actions in response to the external power demand early on. Moreover, compared to the long horizon S-MPC, the short horizon controller consumes more energy (Fig. 2-(c)).



Fig. 2. Performance of the S-MPC with different prediction horizon lengths: (a) tracking the desired position trajectory, (b) thermal index, and (c) on-board energy.

It was shown that extending the prediction horizon improves S-MPC tracking and constraint satisfaction performance. While this observation is not a surprise, as discussed in Sec II, long horizon optimization for slow states is associated with demand forecast uncertainty and computational complexity. Fig. 2 also shows the S-MPC performance with N = 100 when the incorporated preview is based on the approximate knowledge of the demand profile (Fig. 1-(a)). As can be seen, the uncertainty in the demand preview

significantly alters the long horizon S-MPC performance. The desired position is not being tracked preferably and the thermal index constraint violation occurs more often. Additionally, according to Fig. 2-(c), the S-MPC with the average demand preview has the highest energy consumption.

To address the demand forecast uncertainty impact on the S-MPC performance, a robust H-MPC is designed for the vehicle model (14) according to the proposed framework in Secs. II and III. The scheduling layer MPC is based on the optimization problem formulated in (4) with the same stage cost defined in (17), and with a slower update rate. The long prediction horizon of the scheduling layer MPC is set to $H_s = 20 \ (100 \ sec)$. The sampling time of the scheduling layer MPC is $T_s = 5 \ sec$, thus $\nu = 5$. The robust constraint set $\mathcal{X}^{\text{robust}}$ is calculated according to the procedure (9)-(11) proposed in Sec. III.

The scheduling layer MPC optimization problem is solved according to the average demand preview shown in Fig. 1-(a). This solution is used to calculate the optimal thermal index trajectory (* x_4^{slow}). This scheduled * x_4^{slow} trajectory is then incorporated in the stage cost of the piloting layer, which is formulated according to (7) as a finite-horizon optimization problem over a relatively shorter prediction horizon ($H_p =$ 20 (20 sec)) and with a faster sampling period of $T_p = T =$ 1 sec. Note that it is assumed the exact knowledge of the demand preview is available to the piloting-layer MPC over the receding horizon $H_p = 20$ (20 sec). The stage cost of the piloting layer MPC (ℓ_{piloting}) is defined as:

$$\ell_{\text{piloting}} = \lambda_1 \tilde{u}_1^2 + \lambda_2 \tilde{u}_2^2 + \lambda_3 (x_1 - x_1^d)^2 + \lambda_4 (x_4 - {}^*x_4^{\text{slow}})^2.$$
(18)

Since $T_s > T_p$, the scheduled $*x_4^{slow}$ trajectory is passed on as a piecewise constant function according to (6). Fig. 3 shows the comparison between the S-MPC and the baseline H-MPC with \mathcal{X} as the constraint set. The H-MPC, unlike the S-MPC with average demand preview, manages to deliver an acceptable x_1 tracking performance (Fig. 3-(a)). Since the robust constraint set \mathcal{X}^{robust} is not being incorporated in the baseline H-MPC (4), often violation of thermal index state x_4 is still observed in Fig. 3-(b). Additionally, compared to the S-MPC with long prediction horizon and average demand preview, the H-MPC shows less energy consumption (Fig. 3-(c)) Overall, the H-MPC slightly relaxes the accurate x_1 tracking requirement by putting part of the effort into tracking the scheduled $*x_{4}^{slow}$. However, due to the uncertainty in the long-range demand forecast used at the scheduling layer, x_4 constraint enforcement is not achieved. Note that operating the vehicles while the thermal index is maintained close to the upper limit x_4^{UL} is equivalent to less energy consumption. Since the demand preview is uncertain, steering x_4 to x_4^{UL} (e.g., 30^oC) increases the chances of constraint violation once the vehicle faces the actual power demand.

Upon incorporation of $\mathcal{X}^{\text{robust}}$ in (4), the robust H-MPC (12) becomes able to take advantage of the short horizon and more accurate demand preview over the piloting layer horizon. As shown in Fig. 4, compared to the baseline H-MPC



Fig. 3. Comparison between performances of S-MPC and H-MPC: (a) tracking the desired position trajectory, (b) thermal index, and (c) on-board energy ($H_p = 20 \ sec$, $H_s = N = 100 \ sec$).

(4) with \mathcal{X} as the constraint set at the scheduling layer, the robust H-MPC (12) with $\mathcal{X}^{\text{robust}}$ shows significantly better x_4 constraint enforcement under the uncertainty associated with long horizon demand forecast. As shown in Fig. 4, the robust H-MPC starts to tighten the x_4^{UL} constraint early on to ensure its enforcement during the period that the demanded load on the battery is higher and any uncertainty in the demand preview could lead to violation of the thermal index limit.



Fig. 4. Comparison between performances of H-MPC and robust H-MPC in regulating the energy storage thermal index .

The performance of the robust H-MPC is also compared with the H-MPC with $\mathcal{X}^{\text{passive}}$, which is computed according to (13). Unlike $\mathcal{X}^{\text{robust}}$, in the passive approach \mathcal{X} is tightened whenever the violation of x_4^{UL} is detected by the piloting layer MPC. This means the scheduling layer MPC is informed about the constraint violation only after its occurrence, then it manages to update the scheduled trajectories to reduce the chances of constraint violation in subsequent time steps. Fig. 4 shows that compared to the baseline H-MPC, the H-MPC with $\mathcal{X}^{\text{passive}}$ can only slightly decrease the x_4 constraint violation. This can be explained with respect to the slower dynamics of x_4 which call for a proactive constraint tightening algorithm to effectively



Fig. 5. Comparison between performances of H-MPC and robust H-MPC: (a) tracking the desired position trajectory, and (b) on-board stored energy $(H_p = 20 \ sec, \ H_s = N = 100 \ sec)$

improve the H-MPC robustness. This requirement has been addressed by the proposed robust H-MPC framework in this paper. Note that with baseline and passive H-MPCs, enforcement of the original x_4^{UL} constraint is infeasible. As a results, the computed \mathcal{X} and $\mathcal{X}^{\text{passive}}$ shown in Fig. 4 are based on the softened x_4 constraint using slack variables.

The other interesting observation about the robust H-MPC is its performance in tracking x_1^d (Fig. 5-(a)) while enforcing x_4^{UL} constraint. Given the limited energy stored on-board of the vehicle, the robust H-MPC slightly relaxes the x_1 trajectory tracking objective (while it enforces x_1 constraint) during the period it puts more effort to mitigate x_4^{UL} violation by decreasing (cooling) the energy storage thermal index from around t = 10 sec to t = 60 sec. After this time period, since the robust H-MPC does not see any other major external power demand over the long prediction horizon, it increases the position tracking priority, see Fig. 5.

The energy consumption results of different studied H-MPCs are shown in Fig. 5-(b). It can be seen that the robust H-MPC has the lowest energy consumption, as compared to the baseline and passive H-MPCs. This is an interesting observation as it shows the robust H-MPC framework does not necessarily lead to a conservative controller design which is usually expected when designing a robust MPC under the influence of unknown external disturbances. We recall that for the specific class of dynamic systems with slow states considered in this paper, steering the slow state to its limit usually leads to lower energy consumption. The constraint violation often occurs when the external power demand on the energy storage is higher, meaning during the time periods that the energy storage is required to deliver power for operation of the vehicle, it also has to provide extra power for regulating the thermal state. The robust H-MPC, on the other hand, effectively shifts the thermal management load to those periods during which the power demand is relatively lower. Thanks to this intelligent thermal management load shift by the robust H-MPC, not only less often thermal constraint violations are observed, but the overall efficiency (fuel economy) of the vehicle is also higher, compared to the baseline H-MPC.

V. SUMMARY AND CONCLUSIONS

The problem of robust hierarchical MPC (H-MPC) design for constrained systems with multi-timescales and exposed to external demands was considered in this paper. From a new prospective, we proposed to use demand preview information with different accuracies at different layers of an H-MPC to proactively predict the (chances of) constraint violations and compute a robust constrain set to enhance the overall robustness of the H-MPC against the uncertainty associated with long horizon demand forecast. The computation of the robust constraint set is built upon the information that is available at each layer of an H-MPC, but they are not often being communicated between the layers. By leveraging this information and accessing to demand preview information with different accuracies, we showed that the robustness of conventional H-MPC can be effectively improved. We demonstrated the application of the proposed robust H-MPC for a simplified vehicle thermal management case study. The simulation results confirmed the effectiveness of the proposed approach in regulating the thermal state of the vehicle based on an uncertain knowledge of the demand preview while improving the overall system efficiency, as compared to a conventional H-MPC framework.

REFERENCES

 E.D.M. Medagoda and P.W. Gibbens. Multiple Horizon Model Predictive Flight Control. *Journal of Guidance, Control, and Dynamics*, 37(3):946–951, 2014.

- [2] W.C. Clarke, C. Manzie, and M.J. Brear. Hierarchical Economic MPC for Systems with Storage States. *Automatica*, 94:138–150, 2018.
- [3] M.R. Amini, X. Gong, H. Wang, Y. Feng, I. Kolmanovsky, and J. Sun. Sequential Optimization of Speed, Thermal Load, and Power Split in Connected HEVs. In ACC, 2019. Philadelphia, PA, USA.
- [4] M.R. Amini, I. Kolmanovsky, and J. Sun. Two-Layer Model Predictive Battery Thermal and Energy Management Optimization for Connected and Automated Electric Vehicles. In 57th IEEE CDC, 2018. Miami Beach, FL, USA.
- [5] M.R. Amini, H. Wang, X. Gong, D. Liao-McPherson, I. Kolmanovsky, and J. Sun. Cabin and Battery Thermal Management of Connected and Automated HEVs for Improved Energy Efficiency Using Hierarchical Model Predictive Control. *IEEE Transactions on Control Systems Technology*, 2019.
- [6] J.P. Koeln and A.G. Alleyne. Robust Hierarchical Model Predictive Control of Graph-Based Power Flow Systems. *Automatica*, 96:127– 133, 2018.
- [7] J. Koeln, H. Pangborn, M. Williams, M. Kawamura, and A. Alleyne. Hierarchical Control of Aircraft Electro-Thermal Systems. *IEEE Transactions on Control Systems Technology*, 2019.
- [8] D. Barcelliy, A. Bemporad, and G. Ripaccioliy. Hierarchical Multi-Rate Control Design for Constrained Linear Systems. In 49th CDC, 2010. Atlanta, GA, USA.
- [9] R. Scattolini and P. Colaneri. Hierarchical Model Predictive Control. In 46th IEEE CDC, 2007. New Orleans, LA, USA.
- [10] R. Scattolini. Architectures for Distributed and Hierarchical Model Predictive Control–A Review. Process Control, 19(5):723–731, 2009.
- [11] B. Picasso, D. De Vito, R. Scattolini, and P. Colaneri. An MPC Approach to the Design of Two-Layer Hierarchical Control Systems. *Automatica*, 46(5):823–831, 2010.
- [12] Z. Yang, Y. Feng, X. Gong, D. Zhao, and J. Sun. Eco-trajectory Planning with Consideration of Queue Along Congested Corridor for Hybrid Electric Vehicles. In *TRB*, 2019. Washington, D.C., USA.
- [13] J.P. Koeln and A.G. Alleyne. Two-Level Hierarchical Mission-Based Model Predictive Control. In ACC, 2018. Milwaukee, WI, USA.
- [14] J. Lofberg. YALMIP: A Toolbox for Modeling and Optimization in MATLAB. In *IEEE CACSD*, 2004. New Orleans, LA, USA.
- [15] A. Wächter and L. Biegler. On the Implementation of an Interior-Point Filter Line-Search Algorithm for Large-Scale Nonlinear Programming. *Mathematical Programming*, 106(1):25–57, 2006.