Hierarchical MPC for coupled subsystems using adjustable tubes

Vignesh Raghuraman and Justin P. Koeln

University of Texas at Dallas, Richardson, TX, 75080, United States
(e-mail: vignesh.raghuraman@utdallas.edu, justin.koeln@utdallas.edu)

Preprint submitted for Elsevier

Abstract
A hierarchical Model Predictive Control (MPC) formulation is presented for coupled discrete-time linear systems with state and input constraints. Compared to a centralized approach, a two-level hierarchical controller, with one controller in the upper-level and one controller per subsystem in the lower-level, can significantly reduce the computational cost associated with MPC. Hierarchical coordination is achieved using adjustable tubes, which are optimized by the upper-level controller and bound permissible lower-level controller deviations from the system trajectories determined by the upper-level controller. The size of these adjustable tubes determines the degree of uncertainty between subsystems and directly affects the required constraint tightening under a tube-based robust MPC framework. Sets are represented as zonotopes to enable the ability to optimize the size of these adjustable tubes and perform the necessary constraint tightening online as part of the MPC optimization problems. State and input constraint satisfaction is proven for the two-level hierarchical controller with an arbitrary number of controllers at the lower-level and a numerical example demonstrates the key features and performance of the approach.

Key words: Hierarchical control, Robust model predictive control, Set-based computing, Zonotopes.

1 Introduction
Model Predictive Control (MPC) of constrained dynamic systems provides the ability to satisfy both input and state constraints to guarantee safe and reliable system operation. This is particularly important for systems where the desired operation requires both transient and steady-state input and state trajectories to approach these constraints. Examples include the control of water distribution networks [1], aircraft power systems [2], smart power grids [3, 4], and hybrid electric vehicles [5, 6]. However, centralized MPC approaches are not well-suited for the control of these complex multi-timescale systems, where the system is comprised of many dynamically coupled subsystems and achieving the desired operation requires both fast control update rates and long prediction horizons.

For these complex systems, hierarchical MPC can be used to decompose control decision across multiple levels of controllers [7]. Typically, upper-level controllers are designed with large time step sizes to optimize system operation over long prediction horizons while lower-level controllers use small time step sizes to resolve the fast dynamics of the system over short prediction.
horizons. With a single controller per level, vertical hierarchical MPC is a computationally efficient approach for controlling multi-timescale systems with a relatively low number of states and inputs [8]. For more complex systems, comprised of multiple dynamically-coupled subsystems, full hierarchical MPC utilizes multiple controllers at each of the lower-levels to reduce the number of control decisions per controller [9][11].

Many hierarchical MPC formulations [9][12] have been developed with a focus on either practical application or theoretical guarantees. Specifically, the two-level hierarchical controller in [9] with an upper-level MPC and a lower-level linear controller achieves state and input constraint satisfaction through communication of optimal references and reference rate changes between controller levels and also guarantees closed-loop stability. The controller developed in [12] provides guaranteed persistent controller feasibility and closed-loop stability for cascaded system with actuator dynamics subject to input constraints. In this case, coordination is achieved through the appropriate choice of contractive terminal constraint sets and terminal control laws, which overall guarantee stability of the error between inner-loop and outer-loop reference models to the origin. The works in [10,11] extend the vertical hierarchical architecture to a full two-level hierarchical controller with one upper-level controller and multiple controllers at the lower-level, one for each subsystem operating at the same timescale in [10] and different timescales in [11], and guarantees closed-loop stability and input constraint satisfaction while driving the system to a desired steady state. Note that [9,11,12] focus on driving the system to a desired steady-state equilibrium. However, for systems with finite operation, such steady-state equilibrium might not exist, as in the case of systems whose operation is based on the utilization of a finite resource (e.g. battery state of charge in an electric vehicle [6,13] or fuel in an aircraft [14]).

Similar to [8,15], this work focuses on the notion of completion, with the goal of maximizing transient performance by satisfying state, input, and terminal constraints during system operation. While the multi-rate hierarchical MPC proposed in [11] achieves real-time computational performance using a full hierarchical MPC architecture with a reduced-order model at the upper-level, guarantees on closed-loop state constraint satisfaction are not shown explicitly. Additionally, the amount of control flexibility provided to the upper- and lower-level controllers along with the resulting uncertainty sets, robust positive invariant (RPI) sets, and tightened constraint sets are determined offline and might not be the optimal choice for systems that need a time-varying control flexibility. Moreover, guaranteed convergence might not be possible for a wide range of systems due to underlying assumptions on the slow timescale of the upper-level controller. To address these challenges, this work focuses on development of a set-based hierarchical MPC architecture for linear systems of dynamically-coupled subsystems that guarantees state and input constraint satisfaction.

One of the fundamental considerations for coordination in hierarchical MPC is how to provide lower-level controllers the flexibility to use their fast update rates and the fast dynamics of the system to improve upon the control decisions made by upper-level controllers without introducing unnecessary conservatism to account for this flexibility. In the authors’ prior work [8], set-based vertical hierarchical MPC was proposed, where waysets were used as the primary coordination mechanism to provide both control performance and guaranteed constraint satisfaction. Strategically designed terminal costs were added to complement the waysets to guarantee that the lower-level controllers can only improve control performance compared to the upper-level controller trajectories [16].

For full hierarchical MPC of systems of dynamically-coupled subsystems, providing lower-level controllers the flexibility to deviate from the trajectories planned by upper-level controllers introduces uncertainty between subsystems. Therefore, the desired degree of flexibility balances the benefits of allowing lower-level controller to improve control performance within their own subsystem with the cost of creating unknown disturbances for neighboring subsystems. This trade off can be time-varying, where certain system operations might require a high level of
coordination between subsystems, resulting in very little flexibility for lower-level subsystem controllers to deviate from the upper-level system-wide control plan. Alternatively, other system operations might not require much coordination between subsystems and lower-level controllers should be permitted a high degree of flexibility to further improve control performance.

The proposed two-level hierarchical MPC framework provides this time-varying subsystem coordination flexibility using an adjustable tube set-based coordination mechanism. Specifically, while planning system state and input trajectories, the upper-level controller simultaneously optimizes the permissible deviations from these trajectories provided to the lower-level subsystem controllers and the corresponding constraint tightening needed to be robust to these deviations. These time-varying permissible deviations are communicated to the lower-level controllers that use this flexibility to further optimize subsystem operation. The ability to embed the optimization of these permissible deviation bounds within the upper-level MPC optimization problem is enabled by zonotopes and the recent work on computing Robust Positive Invariant (RPI) sets and Pontryagin difference set operations using linear constraints [17,18].

The specific contributions of this paper are: (1) the development a two-level hierarchical framework with $M$ lower-level controllers, one for each of the $M$ dynamically-coupled subsystems; (2) the definition and the use of adjustable tubes to provide time-varying bounds on permissible deviations between upper-level and lower-level planned trajectories; (3) the closed-loop analysis of the hierarchical controller to prove controller feasibility and guarantee constraint satisfaction; and (4) a numerical demonstration of the capabilities of the proposed approach. Note that the proposed work extends the tube-based robust MPC with uncertainty set optimization from [18] to a hierarchical MPC framework with optimal allocation of uncertainty quantified as the differences in control decisions between controller levels and between subsystems. Similar to [18], RPI, tightened output, and tightened terminal sets corresponding to the optimized uncertainty are computed online while solving the control optimization problem.

2 Notation

For a system comprised of multiple subsystems, system-level vectors are denoted in bold, e.g. state $\mathbf{x}$ and input $\mathbf{u}$, while vectors of the $i^{th}$ subsystem have the subscript $i$, e.g. state $x_i$ and input $u_i$. The system state vector is formed by the concatenation of subsystem state vectors as $\mathbf{x} = [x_i]$. Alternatively, the states of subsystem $i$ can be extracted from the system state vector as $x_i = \Pi_i \mathbf{x}$. For a discrete-time system, $\mathbf{x}(k)$ denotes the state $\mathbf{x}$ at time step $k$. With $[k, k+N-1]$, the input trajectory over these time steps is denoted $\{u(j)\}_{j=k}^{k+N-1}$. For MPC, the double index notation $\mathbf{x}(k+l|k)$ denotes the predicted state at future time $k+l$ determined at time step $k$. The block-diagonal matrix $K$ with blocks $K_i$ is denoted $K = \text{diag}(K_i)$. The $p$-norm of a vector is denoted $||\cdot||_p$ and the weighted norm is $||\mathbf{x}||_A = \mathbf{x}^T \Lambda \mathbf{x}$, where $\Lambda$ is a positive-definite diagonal matrix. All sets are shown in caligraphic font. For sets $\mathcal{X}, \mathcal{Y} \in \mathbb{R}^n$, $\mathcal{X} \oplus \mathcal{Y}$ denotes the Minkowski sum and $\mathcal{X} \odot \mathcal{Y}$ denotes the Minkowski/Pontryagin difference of $\mathcal{Y}$ from $\mathcal{X}$. The Cartesian product of sets is denoted as $\mathcal{X} \times \mathcal{Y}$. The projection of $\mathcal{X}$ on the $n_i$ dimensions of subsystem $i$ is denoted as $\mathcal{X}_i = \Pi_i \mathcal{X}$.

3 Problem Formulation

Consider a linear discrete time-invariant system composed of $M$ dynamically-coupled subsystems, $\mathbf{S}_i$, where $i \in \mathcal{N} \equiv [1, M]$. The dynamics of subsystem $\mathbf{S}_i$ are

\begin{align}
    x_i(k+1) &= A_i x_i(k) + B_i u_i(k) + w_i(k), \quad (1a) \\
    y_i(k) &= C_i x_i(k) + D_i u_i(k), \quad (1b)
\end{align}
where $x_i \in \mathbb{R}^{n_i}$ are the states, $u_i \in \mathbb{R}^{m_i}$ are the inputs, and $y_i \in \mathbb{R}^{r_i+m_i}$ are the outputs. The coupling between subsystems is captured by the disturbance vector

$$w_i(k) = \sum_{j \in \mathcal{N}_i} (A_{ij}x_j(k) + B_{ij}u_j(k)), \quad (2)$$

where $\mathcal{N}_i$ is the set of neighboring subsystems such that

$$\mathcal{N}_i \triangleq \{ j \in \mathcal{N} \setminus \{i\} : [A_{ij} B_{ij}] \neq 0 \}. \quad (3)$$

The outputs are defined to include all states and inputs such that $y_i(k) \triangleq [x_i(k)^T u_i(k)^T]^T$ and $[C_i D_i] \triangleq I_{n_i+m_i}$. The subsystem states, inputs, and outputs are constrained such that

$$x_i(k) \in \mathcal{X}_i, \quad u_i(k) \in \mathcal{U}_i, \quad y_i(k) \in \mathcal{Y}_i \triangleq \mathcal{X}_i \times \mathcal{U}_i. \quad (4)$$

Based on (1) and (2), the full system dynamics are

$$x(k+1) = Ax(k) + Bu(k), \quad (5a)$$
$$y(k) = Cx(k) + Du(k), \quad (5b)$$

where $x = [x_i] \in \mathbb{R}^n$, $u = [u_i] \in \mathbb{R}^m$, and $y = [y_i] \in \mathbb{R}^{n+m}$, such that $n = \sum_{i=1}^{M} n_i$ and $m = \sum_{i=1}^{M} m_i$. The system constraints are

$$x(k) \in \mathcal{X} \triangleq \mathcal{X}_1 \times \ldots \times \mathcal{X}_M, \quad (6a)$$
$$u(k) \in \mathcal{U} \triangleq \mathcal{U}_1 \times \ldots \times \mathcal{U}_M, \quad (6b)$$
$$y(k) \in \mathcal{Y} \triangleq \mathcal{Y}_1 \times \ldots \times \mathcal{Y}_M. \quad (6c)$$

Let $A_D \triangleq \text{diag}(A_{ii})$ and $B_D \triangleq \text{diag}(B_{ii})$ be block diagonal matrices while $A_C \triangleq A - A_D$ and $B_C \triangleq B - B_D$ are off-diagonal matrices that capture the coupling between subsystems.

**Assumption 1.** There exists a static feedback control gain $K_i \in \mathbb{R}^{m_i \times n_i}$ for each subsystem $S_i$, $i \in \mathcal{N}$, such that $A_{ii} + B_{ii}K_i$ is Schur stable and $A + BK$ is Schur stable, where $K = \text{diag}(K_i)$ is a block-diagonal matrix.

**Assumption 2.** With a fixed time step $\Delta t$, the system operates for a finite length of time starting from $t = 0$ and ending at $t = t_F = k_F \Delta t$ with time steps indexed by $k \in [0, k_F]$.

Starting from an initial condition $x(0)$, the goal is to plan and execute an input trajectory and corresponding state and output trajectories satisfying the system dynamics from (5), the constraints from (6) for all $k \in [0, k_F - 1]$, and the terminal constraint

$$x(k_F) \in \mathcal{T} \triangleq \mathcal{T}_1 \times \ldots \times \mathcal{T}_M \subseteq \mathcal{X}. \quad (7)$$

**Assumption 3.** The sets $\mathcal{X}_i$, $\mathcal{U}_i$, and $\mathcal{T}_i$, $i \in \mathcal{N}$, are zonotopes.

The generic cost function

$$J(x(0)) = \sum_{j=0}^{k_F-1} (\ell(j) + \ell_F(k_F)), \quad (8)$$

defines the cost of system operation using a pre-determined reference trajectory $\{r(k)\}_{k=0}^{k_F}$ with stage costs $\ell(j) = \ell(x(j), u(j), r(j))$ and terminal cost $\ell_F(k_F) = \ell_F(x(k_F), r(k_F))$.

Considering the full system (5), operational constraints (6), terminal constraint (7), and cost function (8), this paper develops a two-level hierarchical control approach with $M$ controllers at the lower-level that guarantees constraint satisfaction and provides computational efficiency in the case of a large number of subsystems $M$, small time step size $\Delta t$, and large operating duration $t_F$. 

---

4
4 Hierarchical Control

The proposed hierarchical control formulation consists of a single controller $C_0$ in the upper-level and $M$ controllers $C_i, i \in \mathcal{N}$, in the lower-level, where $C_i$ controls subsystem $S_i$.

**Assumption 4.** The controller $C_0$ has a time step size $\Delta t_0$ and maximum prediction horizon $\bar{N}_0$ such that $\Delta t_0 \bar{N}_0 = t_F$. Each controller $C_i, i \in \mathcal{N}$, has a time step size $\Delta t$ and maximum prediction horizon $\bar{N}$ such that $\Delta t \bar{N} = \Delta t_0$.

Let $\nu_0 \triangleq \frac{\Delta t_0}{\Delta t} = \bar{N} \in \mathbb{Z}_+$ be defined as a time scaling factor for $C_0$. The time steps for $C_0$ are indexed by $k_0$, with $k_0 \triangleq \frac{k}{\nu_0}$, and let $k_{0,F} \triangleq \frac{k_F}{\nu_0} = \bar{N}_0$ denote the terminal step of $C_0$ such that $k_0 \in [0, k_{0,F}]$. Thus, the upper-level controller $C_0$ has a shrinking horizon, with time-varying horizon length $N_0(k_0) \triangleq \bar{N}_0 - k_0$. Each lower-level controller $C_i$ has a shrinking and resetting horizon, with horizon length $N(k) \triangleq \bar{N} - (k \mod \bar{N})$. This allows $C_i$ to predict between updates of $C_0$, at which point $(k \mod \bar{N} = 0)$ and the prediction horizon resets back to $N(k) = \bar{N}$.

Similar to $[8,19]$, $C_0$ predicts coarse state and input trajectories at time indices $k_0$ with a large time step size $\Delta t_0$. Lower-level controllers $C_i$ are permitted bounded deviations from the trajectories planned by $C_0$ to further improve control performance using a smaller time step size $\Delta t$. Unlike $[8,19]$, this work addresses the coupling between subsystems. If the lower-level controller $C_i$ chooses to deviate from the state and input trajectories planned by $C_0$, these deviations create unknown disturbances that could lead to constraint violations in neighboring subsystems. Therefore, instead of using waysets as in $[8,19]$, a tube-based coordination mechanism is used to bound the permissible deviations between the trajectories planned by $C_0$ and those planned by $C_i$. Moreover, the size of these permissible deviations is optimized online by $C_0$ to balance the flexibility provided to lower-level controllers with the potentially time-varying need for close coordination among subsystems.

Specifically, for each subsystem, the sets $\Delta Z_i(\delta^z_i(k_0))$ and $\Delta Y_i(\delta^v_i(k_0))$ denote scaled zonotopes that bound the permissible state and input deviations between the trajectories planned by $C_0$ and those planned by $C_i$. The scaling vectors can be collected to form the output deviation vector $\delta_i(k_0) = [\delta^z_i(k_0) \top \delta^v_i(k_0) \top] \top$ and the permissible output deviation set

$$
\Delta Y_i(\delta_i(k_0)) = \Delta Z_i(\delta^z_i(k_0)) \times \Delta Y_i(\delta^v_i(k_0)). \tag{9}
$$

To reduce notational complexity, the shorthand $\Delta Y_i(k_0) = \Delta Y_i(\delta_i(k_0))$ is used when explicitly stating the dependency on $\delta_i(k_0)$ is unnecessary. The system state, input, and output deviation vectors are $\delta^z(k_0) = [\delta^z_i(k_0)]$, $\delta^v(k_0) = [\delta^v_i(k_0)]$, and $\delta(k_0) = [\delta^z(k_0) \top \delta^v(k_0) \top] \top$ and the scaled subsystem deviation sets combine to form the scaled system deviation sets

$$
\Delta Z(\delta(k_0)) = \Delta Z_1(k_0) \times \ldots \times \Delta Z_M(k_0), \tag{10a}
$$

$$
\Delta Y(\delta(k_0)) = \Delta Y_1(k_0) \times \ldots \times \Delta Y_M(k_0), \tag{10b}
$$

$$
\Delta Y(\delta(k_0)) = \Delta Z(\delta(k_0)) \times \Delta Y(\delta(k_0)). \tag{10c}
$$

The controller $C_0$ updates only when $k = \nu_0 k_0$ (i.e. when $k \mod \nu_0 = 0$), by solving the
constrained optimization problem $P_0(x(k))$ defined as

$$J_0^*(x(k)) = \min_{\delta_{k0}, \delta_{k0}, \ldots, \delta_{k0,F-1}} \sum_{j=0}^{k_0,F-1} \ell(j | k_0) + \ell_F(k_0,F), \quad (11a)$$

s.t. $\forall j \in [k_0, k_0,F - 1]$,

$$\dot{x}(j + 1 | k_0) = A_0 \dot{x}(j | k_0) + B_0 \dot{u}(j | k_0), \quad (11b)$$

$$\dot{y}(j | k_0) = C \dot{x}(j | k_0) + D \dot{u}(j | k_0) \in \mathcal{Y}_0(\delta(k_0)), \quad (11c)$$

$$\dot{x}(k_0,F | k_0) \in \mathcal{T}_0(\delta(k_0)), \quad (11d)$$

$$\dot{x}(k) - \dot{x}(k_0 | k_0) \in \Delta Z(\delta(k_0)) \oplus \mathcal{E}_0(\delta(k_0)), \quad (11e)$$

$$\Delta Z(\delta(k_0)) \subseteq \text{Pre}(\Delta Z(\delta(k_0))). \quad (11f)$$

The shrinking horizon of $P_0(x(k))$ is reflected in the summation limits in $(11a)$. The stage costs are defined as $\ell(j | k_0) = \ell(x(k), x(j | k_0), u(j | k_0), r_0(j))$ to be a function of the measured state, nominal state, nominal input, permissible deviations for lower-level controllers, and the reference trajectory. The terminal cost $\ell_F(k_F)$ is the same as in $[5]$. Note that the system performance can be balanced with the maximization of $\delta$ through the addition of the term $\Lambda || \delta - \delta_0 ||_p$, where $\Lambda$ is a scalar weighting term and $\delta_0$ is a user-specified upper-bound on $\delta$. The nominal input trajectory is defined as $\ddot{U}(k_0) = \{u(j | k_0)\}_{j=k_0}^{k_0,F-1}$. The permissible output deviation scaling vector $\delta(k_0)$ affects the sizes of the tightened output constraint set $\mathcal{Y}_0(\delta(k_0))$, the tightened terminal constraint set $\mathcal{T}_0(\delta(k_0))$, the state deviation constraint set $\Delta Z(\delta(k_0))$, and the RPI set $\mathcal{E}_0(\delta(k_0))$, which are time-varying. In $(11b)$, the model used by $C_0$ assumes a piecewise constant control input over the time step size $\Delta t_0$. Thus $A_0 = A^{\nu_0}$ and $B_0 = \sum_{j=0}^{\nu_0-1} A^j B$ (as in $[20]$). In $(11c)$ and $(11d)$, the outputs and terminal state are constrained to the time-varying tightened output and terminal constraint sets. Similar to tube-based MPC $[21]$, $(11e)$ allows $C_0$ flexibility in the choice of initial condition $x'(k_0 | k_0)$, which is used to prove recursive feasibility of $P_0(x(k))$ in Section 6. Finally, $(11f)$ constrains the time-varying permissible state deviation set to be a subset of its own precursor set. Based on the definition from $[22]$, the precursor set is defined specifically as

$$\text{Pre}(\Delta Z(\delta(k_0))) = \{z \mid \exists v \in \Delta V(\delta(k_0)) \text{ s.t. } A_D z + B_D v \in \Delta Z(\delta(k_0))\}, \quad (12)$$

and is used to establish feasibility of the lower-level controllers in Section 6. The details of how to formulate the sets and set containment conditions used in $(11)$ as linear constraints are provided in the Appendix. The reference trajectory $r_0(j)$ can be obtained by downsampling the predetermined reference trajectory $r(j)$ either using averaging or zero order hold $[19]$. Note that the RPI set $\mathcal{E}_0(\delta(k_0))$ is assumed to be a structured RPI set such that

$$\mathcal{E}_0(\delta(k_0)) = \mathcal{E}_1(\delta_1(k_0)) \times \ldots \times \mathcal{E}_M(\delta_M(k_0)), \quad (13)$$

and is formulated in more detail in Section 5.

The lower-level controllers $C_i$, $i \in \mathcal{N}$, update at each time index $k$ by each solving, in parallel, the constrained optimization problems $P_i(x_i(k))$, defined as

$$J_i^*(x_i(k)) = \min_{\delta_i \in \mathcal{N}} \sum_{j=k}^{k+N(k)-1} \ell_i(j | k) + \ell_i,F(k + N(k)) \quad (14a)$$

s.t. $\forall j \in [k, k + N(k) - 1]$,

$$z_i(j + 1 | k) = A_{ii} z_i(j | k) + B_{ii} v_i(j | k) + \hat{w}_i(j), \quad (14b)$$

$$y_i(j | k) = C_{ii} z_i(j | k) + D_{ii} v_i(j | k), \quad (14c)$$

$$y_i(j | k) - \hat{y}_i(j) \in \Delta \mathcal{Y}_i(\delta_i^*(k_0)), \quad (14d)$$

$$z_i(k + N(k) | k) - \hat{z}_i(k + N(k)) \in \Delta \mathcal{Z}_i(\delta_i^*(k_0)), \quad (14e)$$

$$x_i(k) - z_i(k | k) \in \mathcal{E}_i(\delta_i^*(k_0)). \quad (14f)$$
Figure 1: Two-level hierarchical MPC where $C_0$ is formulated based on (11) and $C_i$, $i \in \mathcal{N}$, based on (14). The $C_0$-optimal trajectories $\hat{y}_i^*(j)$ and $\hat{w}_i^*(j)$ are computed using (16) and (17). The optimal output deviations $\delta_i^*(k_0)$ are used to coordinate controllers $C_0$ and $C_i$, $i \in \mathcal{N}$, and the static feedback control law (15) computes the inputs to each subsystem $S_i$.

The shrinking and resetting horizon of $P_i(x_i(k))$ is reflected in the summation limits in (14a). The stage costs are defined as $\ell_i(j|k) = \ell_i(x_i(k), z_i(j|k), v_i(j|k), r_i(j))$ to be a function of the measured subsystem state, nominal subsystem state, nominal subsystem input, and subsystem reference trajectory. The terminal cost is defined as $\ell_i,F(k+k(N(k)) = \ell_i,F(z_i(k+k(N(k)|k), r_i(k+k(N(k)))).$ The subsystem dynamics from (1a) are used with a time-varying $C_0$-optimal disturbance $\hat{w}_i^*(j)$ that is communicated from $C_0$ (details in Section 5). Nominal subsystem outputs are defined in (14c) and the differences between these outputs and the $C_0$-optimal outputs $\hat{y}_i^*(j)$ are constrained in (14d) to the time-varying permissible output deviation set $\Delta Y_i(\delta_i^*(k_0))$ (details in Section 5). Similarly, the difference between the nominal terminal state and the $C_0$-optimal terminal state is constrained to the time-varying permissible state deviation set $\Delta Z_i(\delta_i^*(k_0))$ in (14e). Finally, (14f) provides flexibility in initial condition $z_i(k|k)$ based on the RPI set computed by $C_0$.

As shown in Fig. 1, coordination between the upper-level controller $C_0$ and lower-level controllers $C_i$, $i \in \mathcal{N}$, is achieved through the communication of the $C_0$-optimal trajectories $\hat{y}_i^*(j)$ and $\hat{w}_i^*(j)$, $j \in [k, k+N(k) - 1]$, terminal state $\hat{x}_i^*(k+N(k))$, and the time-varying permissible deviation vectors $\delta_i^*(k_0)$. In this hierarchical control architecture, only the lower-level controllers $C_i$ directly affect the system through inputs to the subsystems $S_i$. Once each $C_i$ has solved for the optimal nominal input trajectories $V_i^*(k)$ and optimal nominal initial condition $z_i^*(k|k)$, the input to the system is $u(k) = [u_i(k)]$ where

$$u_i(k) = v_i^*(k|k) + K_i(x_i(k) - z_i^*(k|k)).$$

The two-level hierarchical controller is implemented based on Algorithm 1. The specific formulation of the sets in (11) and (14) are presented in Section 5 and the corresponding constraints are used to guarantee satisfaction of the state, input, and terminal constraints from (6) and (7) in Section 6.
Algorithm 1: Two-level Hierarchical MPC with subsystem coupling.

1. Initialize $k, k_0 \leftarrow 0$
2. if $k \mod v_0 = 0$ then
3. solve $P_0(x(k))$;
4. communicate $\{\hat{y}_i^*(j)\}_{j=k}^{k+N(k)}$, $\{\hat{u}_i^*(j)\}_{j=k}^{k+N(k)}$, $\hat{x}_i^*(k + N(k))$, and $\delta_i^*(k_0)$ to $P_i(x_i(k)), \forall i \in \mathcal{N}$;
5. $k_0 \leftarrow k_0 + 1$;
6. end
7. solve $P_i(x_i(k)), \forall i \in \mathcal{N}$, and apply the input $u(k) = [u_i(k)]$ to the system based on (15);
8. $k \leftarrow k + 1$;

5 Nominal Trajectories and Error Propagation

Following the tube-based MPC framework in [21], the goal of this section is to explicitly bound the differences between the nominal state and input trajectories planned by the controllers $C_0$ and $C_i$, $i \in \mathcal{N}$, and the true system trajectories.

First, since $C_0$ has a larger time step size than $C_i$ and system dynamics (i.e. $\Delta t_0 > \Delta t$), the input and state trajectories determined by $C_0$ must be upsampled. Let $\hat{u}^*(k)$ and $\hat{x}^*(k)$ be the upsamled input and state trajectories corresponding to the optimal trajectories determined by $C_0$. Since the model (11b) assumed a piecewise constant input, the upsamled trajectories are computed as the forward simulation of (5a) such that

$$\hat{u}^*(k) = \hat{u}^*(k_0 | k_0),$$

(16a)

$$\hat{x}^*(k) = A^{k-k_0k_0} \hat{x}^*(k_0 | k_0) + \sum_{j=0}^{k-\nu_0k_0-1} A^j B \hat{u}^*(k_0 | k_0),$$

(16b)

for $k \in [\nu_0k_0, \nu_0(k_0 + 1) - 1]$. These trajectories create the $C_0$-optimal output and disturbance trajectories $\hat{y}_i^*(k)$ and $\hat{w}_i^*(k)$ used in (14d) and (14b), where

$$\hat{u}_i^*(k) = \Pi_i \hat{u}^*(k), \quad \hat{x}_i^*(k) = \Pi_i \hat{x}^*(k),$$

and

$$\hat{y}_i^*(k) = [\hat{x}_i^*(k) \equiv \hat{y}_i^*(k)]^T,$$

(17a)

$$\hat{w}_i^*(k) = \sum_{j \in \mathcal{N}_i} (A_{ij} \hat{x}_j^*(k) + B_{ij} \hat{u}_j^*(k)).$$

(17b)

Having defined the upsamled nominal trajectories for $C_0$, let $\Delta x(k) = [\Delta x_i(k)]$, $\Delta u(k) = [\Delta u_i(k)]$, and $\Delta y(k) = [\Delta y_i(k)]$ denote the state, input, and output prediction errors for $C_0$, where

$$\Delta x_i(k) \equiv x_i(k) - \hat{x}_i^*(k),$$

(18a)

$$\Delta u_i(k) \equiv u_i(k) - \hat{u}_i^*(k),$$

(18b)

$$\Delta y_i(k) \equiv y_i(k) - \hat{y}_i^*(k) = [\Delta x_i(k) \equiv \Delta x_i(k) \equiv \Delta u_i(k)]^T.$$

(18c)

As shown in Fig. [2] these upper-level prediction errors consist of two parts, corresponding to the planned deviations by lower-level controllers $C_i$ and the resulting lower-level prediction errors due to the coupling between subsystems. Specifically,

$$\Delta x_i(k) = \Delta z_i(k) + e_i(k),$$

(19a)

$$\Delta u_i(k) = \Delta y_i(k) + K_i e_i(k),$$

(19b)
Figure 2: Schematic showing the true states and inputs for the system compared to those planned by controllers $C_0$ and $C_i$, $i \in \mathcal{N}$, and the corresponding planned and unplanned prediction errors.

where

$$\Delta z_i(k) \triangleq z_i(k) - \hat{x}_i^*(k), \quad \Delta v_i(k) \triangleq v_i(k) - \hat{u}_i^*(k),$$

are the planned deviations and

$$e_i(k) \triangleq x_i(k) - z_i(k),$$

are lower-level prediction errors due to the coupling between subsystems. Note that $K_i e_i(k) = u_i(k) - v_i(k)$ based on the control law from (15). With the planned deviations bounded as $\Delta z_i(k) \in \Delta Z_i(k_0)$ and $\Delta v_i(k) \in \Delta V_i(k_0)$, which are simultaneously imposed as bounded output deviations in (14d), the following two lemmas establish prediction error bounds for the lower- and upper-level controllers.

**Lemma 1.** Let the disturbance error set be defined as

$$\Delta W = A_C \Delta Z \oplus B_C \Delta V.$$  \hspace{2cm} (21)

Then the lower-level prediction errors $e(k) = [e_i(k)]$ are bounded to the RPI set $\mathcal{E}_0 \subset \mathbb{R}^n$, where $\mathcal{E}_0$ satisfies

$$(A + BK)\mathcal{E}_0 \oplus \Delta W \subseteq \mathcal{E}_0.$$  \hspace{2cm} (22)

**Proof.** Using the true subsystem dynamics from (1a) and the nominal subsystem model from (14b), the lower-level prediction error dynamics for each subsystem are

$$e_i(k + 1) = (A_{ii} + B_{ii} K_i) e_i(k) + \Delta w_i(k),$$  \hspace{2cm} (23)

where $\Delta w_i(k) = w_i(k) - \hat{w}_i^*(k)$. Using the definitions of $w_i(k)$ and $\hat{w}_i^*(k)$ from (2) and (17b), this disturbance error is

$$\Delta w_i(k) = \sum_{j \in \mathcal{N}_i} (A_{ij} (x_j(k) - \hat{x}_j^*(k)) + B_{ij} (u_j(k) - \hat{u}_j^*(k))).$$

Based on (18) and (19), this disturbance error can be rewritten as

$$\Delta w_i(k) = \sum_{j \in \mathcal{N}_i} \left[ (A_{ij} + B_{ij} K_j) e_j(k) + A_{ij} \Delta z_j(k) + B_{ij} \Delta v_j(k) \right].$$  \hspace{2cm} (24)

Combining (23) and (24) for all subsystems $i \in \mathcal{N}$ results in the system error dynamics

$$e(k + 1) = (A + BK) e(k) + A_C \Delta z(k) + B_C \Delta v(k).$$

Since $\Delta z_i(k) \in \Delta Z_i$ and $\Delta v_i(k) \in \Delta V_i$, $\forall i \in \mathcal{N}$, $A_C \Delta z(k) + B_C \Delta v(k) \in \Delta W$, as defined in (21). Thus, if $e(k) \in \mathcal{E}_0$ and $\mathcal{E}_0$ satisfies (22), then $e(k + 1) \in \mathcal{E}_0$. \hfill \Box
Lemma 2. The upper-level prediction errors $\Delta x(k) = [\Delta x_i(k)]$ and $\Delta u(k) = [\Delta u_i(k)]$ are bounded such that

$$\Delta x(k) \in \Delta Z \oplus \mathcal{E}_0, \quad \Delta u(k) \in \Delta V \oplus K \mathcal{E}_0.$$  \hfill (25)

Proof. The proof follows directly from the definitions of $\Delta x_i(k)$ and $\Delta u_i(k)$ from \cite{19} and the result of Lemma 1.

Based on the results of Lemmas 1 and 2, the nominal outputs determined by the upper-level controller in (11c) are constrained to the time-varying tightened output constraint set $\hat{\mathcal{Y}}_0(\delta(k_0))$. For notational simplicity, let $\Delta Z = \Delta Z(\delta(k_0))$, $\Delta V = \Delta V(\delta(k_0))$, and $\mathcal{E}_0 = \mathcal{E}_0(\delta(k_0))$. Then the time-varying tightened output constraint set is defined as

$$\hat{\mathcal{Y}}_0(\delta(k_0)) \triangleq \hat{\mathcal{Y}}_0 \oplus [(\Delta Z \oplus \mathcal{E}_0) \times (\Delta V \oplus K \mathcal{E}_0)],$$  \hfill (26)

where $\hat{\mathcal{Y}}_0 \subseteq \mathcal{Y}_0$ is a tightened output constraint set used to prevent inter-sample constraint violations (see Appendix A.1 for details). Similarly, in (11d), the nominal terminal state is constrained to the time-varying tightened terminal constraint set $\hat{T}_0(\delta(k_0))$ defined as

$$\hat{T}_0(\delta(k_0)) \triangleq \mathcal{T} \oplus (\Delta Z \oplus \mathcal{E}_0).$$  \hfill (27)

Note that $\hat{\mathcal{Y}}_0(\delta(k_0))$ and $\hat{T}_0(\delta(k_0))$ are functions of the time-varying permissible output deviations $\delta(k_0)$ due to the definition of $\Delta Z(\delta(k_0))$ and $\Delta V(\delta(k_0))$ and their direct impact on $\mathcal{E}_0(\delta(k_0))$, as established in Lemma 1.

6 Hierarchical Control Feasibility

The following establishes recursive feasibility of each controller in the hierarchy and guarantees constraint satisfaction for the closed-loop system.

Assumption 5. There exists a feasible solution to $\mathbf{P}_0(x(0))$ at time step $k = k_0 = 0$ for the initial condition $x(0)$.

Lemma 3. If $\mathbf{P}_0(x(k))$ is feasible time step $k = v_0k_0$, then $\mathbf{P}_i(x_i(k)), i \in \mathcal{N}$, is feasible at this time step.

Proof. Let $\{\hat{x}^*(j)\}_{j=k}^{k+N(k)}$, $\{\hat{u}^*(j)\}_{j=k}^{k+N(k)-1}$, and $\delta^*(k_0)$ denote upsampled $\mathbf{C}_0$-optimal state and input trajectories and the optimal output deviation determined by $\mathbf{C}_0$ at time step $k = v_0k_0$. First, for all $i \in \mathcal{N}$, it is to be shown that there exists an initial condition $\mathbf{z}(k|k) = [z_i(k|k)]$ that simultaneously satisfies the output deviation constraints from (14c) and the initial condition constraint from (14a). Specifically, the initial condition constraint (11c) ensures that $x(k) - \hat{x}^*(k_0|k_0) \in \Delta Z(\delta^*(k_0)) \oplus \mathcal{E}_0(\delta^*(k_0))$. Based on the structure of these sets from (10a) and (13), $x_i(k) - \hat{x}_i^*(k_0|k_0) \in \Delta Z_i(\delta^*_i(k_0)) \oplus \mathcal{E}_i(\delta^*_i(k_0))$ for all $i \in \mathcal{N}$. Since these sets are all zonotopes, let $\Delta Z_i(\delta^*_i(k_0)) = \{G_i,0\}$ and $\mathcal{E}_i(\delta^*_i(k_0)) = \{G_i,0\}$. Therefore, $x_i(k) - \hat{x}_i^*(k_0|k_0) \in \Delta Z_i(\delta^*_i(k_0)) \oplus \mathcal{E}_i(\delta^*_i(k_0))$ guarantees the existence of $\xi^*_i$ and $\xi^*_i$ such that $||\xi^*_i|| \leq 1, ||\xi^*_i|| \leq 1$, and

$$x_i(k) - \hat{x}_i^*(k_0|k_0) = G_i^*\xi^*_i + G_i^*\xi^*_i.$$  \hfill (28)

Choosing $z_i(k|k) = \hat{x}_i^*(k_0|k_0) + G_i^*\xi^*_i$, ensures that this initial condition satisfies the output deviation constraint from (14d). Solving for $\hat{x}_i^*(k_0|k_0)$ and plugging into (28) results in $x_i(k) - z_i(k|k) = G_i^*\xi^*_i$ and thus this choice of initial condition also satisfies the initial condition constraint from (14a).

It remains to show the existence of a candidate solution starting from this initial condition $z_i(k|k)$, denoted by the nominal input sequence $\{v_i(j|k)\}_{j=k}^{k+N(k)-1}$ and corresponding nominal state sequence $\{z_i(j|k)\}_{j=k}^{k+N(k)}$, that satisfies the model (14b) and the constraints (14c)-(14e).
Comparing the candidate solution satisfying the nominal subsystem dynamics from (14b) and the upsampled C₀-optimal trajectories satisfying the true subsystem dynamics from (1a) results in $z_i(j + 1|k) - \hat{x}_i^*(j + 1) = A_k(z_i(j|k) - \hat{x}_i^*(j)) + B_u(v_i(j|k) - u_i^*(j))$ for all $j \in [k, k + N(k) - 1]$. Since $z_i(k|k)$ has already been shown to satisfy the output deviation constraint from (14d), $z_i(k|k) - \hat{x}_i^*(k) \in \Delta Z_i(\delta_i^*(k_0))$. From (11f) and the definition of the precursor set from (12), this guarantees the existence of $v_i(k|k)$ such that $v_i(k|k) - \hat{u}_i^*(k) \in \Delta V_i(\delta_i^*(k_0))$ and $z_i(k + 1|k) - \hat{x}_i^*(k + 1) \in \Delta Z_i(\delta_i^*(k_0))$. This process is repeated to show that the output deviation constraints from (14d) can be satisfied for all time steps and that the terminal constraint from (14e) is satisfied at the final time step.

Lemma 4. For all $i \in \mathcal{N}$, if $P_i(x_i(k))$ is feasible at time step $k$, where $k \mod \nu_0 = 0$ (i.e. at the time of $C_0$ update), then $P_i(x_i(k))$ is feasible at each time step $k + 1$ through $k + N(k) - 1$.

Proof. Let the feasible solution for $P_i(x_i(k))$ at time step $k$ be defined by the optimal nominal input sequence $\{v_i^*(j|k)\}_{j=k}^{k+N(k)-1}$ and corresponding optimal nominal state sequence $\{z_i^*(j|k)\}_{j=k}^{k+N(k)}$ satisfying (14b). While (14d) guarantees $x_i(k) - z_i^*(k|k) \in \mathcal{E}_i(\delta_i^*(k_0))$, the feasibility of $P_j(x_j(k))$, $j \in \mathcal{N}\{i\}$, ensures that the disturbances $\Delta u_j$ from (24) due to subsystem coupling are bounded to $\Delta W$ used to define $\mathcal{E}_0 = \mathcal{E}_1 \times \cdots \times \mathcal{E}_M$ in Lemma 1. Thus $x_i(k + 1) - z_i^*(k + 1|k) \in \mathcal{E}_i(\delta_i^*(k_0))$, and $\{z_i^*(j|k)\}_{j=k+1}^{k+N(k)}$ and $\{v_i^*(j|k)\}_{j=k+1}^{k+N(k)-1}$ are feasible nominal state and input sequences, which are the tails of sequences determined at previous time step $k$. Thus, $P_i(x_i(k + 1))$ is feasible and by induction, $P_i(x_i(j))$, $\forall j \in [k + 1, k + N(k) - 1)$ is recursively feasible.

Lemma 5. If $P_i(x_i(k - 1)) \forall i \in \mathcal{N}$ had feasible solutions at the previous time step $k - 1$, then $P_0(x(k))$ has a feasible solution at current time step $k = \nu_0 k_0$.

Proof. Let the candidate solution to $P_0(x(k))$ be the optimal nominal state and input sequences $\{\hat{x}_0^*(j|k_0 - 1)\}_{j=k_0}^{k_0+N_0(k_0)}$, $\{\hat{u}_0^*(j|k_0 - 1)\}_{j=k_0}^{k_0+N_0(k_0)-1}$, corresponding to the tails of the optimal solution determined at previous time step $k_0 - 1$, and the previously planned deviation bound $\delta^*(k_0-1)$. Since (11b), (14c), (11e), and (11f) are time-invariant, the candidate solution satisfies these constraints. To show that $\hat{x}_0^*(k_0|k_0-1)$ is a feasible initial condition, consider the following. Since, $P_i(x_i(k - 1))$ is feasible at time step $k - 1$, the terminal state $z_i^*(k|k - 1) = z_i^*(k_0)$ satisfies $z_i^*(k_0) - \hat{x}_i^*(k_0|k_0 - 1) \in \Delta Z_i(\delta_i^*(k_0 - 1))$ for every $S_i$. Additionally, using the invariance of $\mathcal{E}_i(\delta_i^*(k_0 - 1))$ under control law (15), $x_i(k) - z_i^*(k_0) \in \mathcal{E}_i(\delta_i^*(k_0 - 1))$. Thus, by combining these statements for all subsystems, $\hat{x}_0^*(k_0|k_0-1)$ satisfies (11e).

Theorem 1. Following Algorithm 1 for a two-level hierarchical controller with $M$ controllers in the lower-level, all control problems, $P_0(x(k))$ and $P_i(x_i(k))$, $\forall i \in \mathcal{N}$, are feasible, resulting
in system state and input trajectories satisfying state, input, and output constraints from $[6]$ and terminal constraint from $[7]$.

**Proof.** Using **Assumption 5** and **Lemmas 3-5**, Fig. 3 shows how feasibility is established for $C_0$ and $C_i$, $\forall i \in N$. For notational convenience, let

$$\Omega \triangleq \Delta Y(\delta^*(k_0)) \oplus (E_0(\delta^*(k_0)) \times K E_0(\delta^*(k_0))).$$

Since, $P_0(x(k))$ is feasible, the output trajectory $\hat{y}^*(j)$ satisfies

$$\hat{y}^*(j) \in \hat{Y}_0^*(k_0) \triangleq \hat{Y}_0 \ominus \Omega \subseteq Y \ominus \Omega,$$

based on $[11c]$, $[26]$, and the fact that $\hat{Y}_0 \subseteq Y$. Similarly, the feasibility of $P_i(x_i(k))$ guarantees that the output trajectory $y_i^*(k|k)$ generated by $C_i$ satisfies

$$y_i^*(k|k) - \hat{y}_i^*(k) \in \Delta Y_i(\delta_i^*(k_0)),$$

$$y_i(k) - y_i^*(k|k) \in E_i^*(\delta_i^*(k_0)) \times K E_i(\delta_i^*(k_0)),$$

based on $[14d]$ and $[14i]$. Thus, adding $[29a]$ and $[29b]$ for each system results in $y(k) \in \hat{y}^*(k) \ominus \Omega$. Since $\hat{y}^*(k) \in Y \ominus \Omega$, $y(k) \in (Y \ominus \Omega) \oplus \Omega$. Finally, using the anti-extensive property of the set opening operation, $y(k) \in Y$. Note that satisfaction of the terminal constraint from $[7]$ can be proven similarly. $\square$

7 Conclusions

A two-level hierarchical MPC formulation was presented for linear systems of dynamically-coupled subsystems. Adjustable tubes are used to bound permissible deviations between the system trajectories planned by the upper- and lower-level controllers. A tube-based robust MPC formulation with simultaneous uncertainty set optimization and constraint tightening guarantees constraint satisfaction to bounded disturbances between subsystem controllers. This new approach extends the applicability of hierarchical control algorithms to system operation focusing on the notion of completion, where constraint feasible equilibrium might not exist. This document presented proofs that were omitted in the original publication due to space constraints.

APPENDIX

This appendix provides the details necessary to implement the proposed hierarchical MPC controller.

A.1 Inter-sample tightened output constraint set computation for $C_0$

The tightened output constraint set $\hat{Y}_0 \subseteq Y_0$ introduced in $[26]$ is used to ensure that the coarse trajectories planned by $C_0$ produce upsampled trajectories in $[16]$ and $[17a]$ that satisfy the original output constraints. While there are many ways to achieve this, the approach used in this paper is based on bounding the difference between the upsampled trajectories and the linear interpolations of the coarse trajectories planned by $C_0$. Specifically, let $\hat{x}(k_0)$ and $\hat{x}(k_0 + 1)$ denote the first two states predicted by $C_0$ corresponding to the first input $\hat{u}(k_0)$, where $\hat{x}(k_0 + 1) = A_{h_0} \hat{x}(k_0) + \sum_{j=0}^{\nu_0-1} A_j B u(k_0)$. The linearly interpolated trajectories $x_i(k + i)$, $\forall i \in [1, \nu_0 - 1]$, can be computed between $\hat{x}(k_0)$ and $\hat{x}(k_0 + 1)$ as

$$x_i(k + i) = \hat{x}(k_0) + \frac{i}{\nu_0}(\hat{x}(k_0 + 1) - \hat{x}(k_0)).$$
By the convexity of $\mathcal{X}$, $\bar{x}(k_0), \bar{x}(k_0+1) \in \mathcal{X}$ implies $x_i(k+i) \in \mathcal{X}$. However, it is not guaranteed that the upsampled trajectory satisfies $\hat{x}(k+i) \in \mathcal{X}, \forall i \in [1, \nu_0 - 1]$. Defined as $e(k+i) = \hat{x}(k+i) - x_i(k+i)$, the difference between these trajectories can be computed as

$$e(k+i) = A_e(i) \bar{x}(k) + B_e(i) \hat{u}(k),$$

(30)

where $A_e(i) = A^i - \frac{1}{\nu_0} A^{\nu_0} - \frac{1}{\nu_0} \nu A + \frac{1}{\nu_0} \nu A + A^i e$. Since $\bar{x}(k) \in \mathcal{X}$ and $\hat{u}(k) \in \mathcal{U}$, these differences are bounded such that $e(k+i) \in \mathcal{E}_i = \mathcal{X} \cup B_e(i) \mathcal{U}$. Thus, defining $\mathcal{F} = \mathcal{F}_1 \cup \cdots \cup \mathcal{F}_{\nu_0-1}$ ensures $e(k+i) \in \mathcal{F}, \forall i \in [1, \nu_0 - 1]$. Finally, computing $\hat{\mathcal{Y}} = \hat{\mathcal{X}} \times \mathcal{U}$ where $\hat{\mathcal{X}} = \mathcal{X} \cap \mathcal{F}$ guarantees that the upsampled trajectories satisfy the original constraints.

For ease of implementation, outer-approximating bounding boxes of $\mathcal{F}$ are computed and thus, $\mathcal{F}$ is also a bounding box.

### A.2 RPI set computation

This section presents how the RPI set $\mathcal{E}_0(\mathbf{d}(k_0))$, used in (11c), is computed through the addition of linear constraints and cost function terms in the formulation of $\mathbf{P}_0(\mathbf{x}(k))$ using the approach from [18]. Before proceeding with the computation of $\mathcal{E}_0(\mathbf{d}(k_0))$, consider the following definition.

**Definition 1.** [18] The zonotope $Z(\Phi) = \{G\Phi, c\} \subset \mathbb{R}^n$ is a scaled version of the nominal zonotope $Z = \{G, c\}$ with the generator matrix $G$ scaled by a diagonal matrix $\Phi \in \mathbb{R}^{n_g \times n_s}$, $\Phi = \text{diag}(\phi_i), \phi_i \geq 0, \forall i \in [1, n_g]$.

Let the RPI set be a scaled zonotope such that $\mathcal{E}_0(\mathbf{d}) = \{G_\varepsilon \Phi_\varepsilon, 0\}$ with an *a priori* chosen nominal generator matrix $G_\varepsilon \in \mathbb{R}^{n_x \times n_w}$. The permissible state and input deviation sets are defined as scaled zonotopes with centers at the origin such that $\Delta Z = \{G_\varepsilon \text{diag} (\mathbf{d}^z), 0\}$ and $\Delta V = \{G_\varepsilon \text{diag} (\mathbf{d}^v), 0\}$, where $G_\varepsilon \in \mathbb{R}^{n_x \times n_w}$ and $G_\varepsilon \in \mathbb{R}^{n_x \times n_w}$. From (21), the resulting disturbance error set is a scaled zonotope such that $\Delta \mathcal{V} = \{G_w \text{diag} (\mathbf{d}), 0\}$, where $G_w = [A_G G_\varepsilon B_G G_\varepsilon] \in \mathbb{R}^{n_x \times n_w}$ and $n_w = n_z + n_u$.

Following the approach from [18], based on the one-step RPI computation from [17] and the zonotope containment conditions from [23], the decision variables $\Phi_\varepsilon \in \mathbb{R}^{n_x \times n_w}$, $\Gamma_{\varepsilon,1} \in \mathbb{R}^{n_n \times n_w}$, and $\Gamma_{\varepsilon,2} \in \mathbb{R}^{n_n \times n_w}$ are added to $\mathbf{P}_0(\mathbf{x}(k))$ with linear constraints

$$
(A + BK) G_\varepsilon \Phi_\varepsilon = G_\varepsilon \Gamma_{\varepsilon,1},
$$

(31a)

$$
G_w \text{diag} (\mathbf{d}) = G_\varepsilon \Gamma_{\varepsilon,2},
$$

(31b)

$$
|\Gamma_{\varepsilon,1}| 1 + |\Gamma_{\varepsilon,2}| 1 \leq \Phi_\varepsilon 1.
$$

(31c)

The cost function for $\mathbf{P}_0(\mathbf{x}(k))$ is modified to balance system performance with the maximization of $\mathbf{d}$ through the addition of the term $\Lambda \|\mathbf{d} - \bar{d}\|_p$, where $\Lambda$ is a scalar weighting term and $\bar{d}$ is a user-specified upper-bound on $\mathbf{d}$. For the linear constraints (31) to admit a feasible solution, the generator matrix $G_\varepsilon$ needs to be chosen carefully. As in [17][18], an intuitive choice of generators is based on $G_w$ and $A_K = \text{diag}(A_i + B_i K_i)$ such that

$$G_\varepsilon = [G_w A_K G_w \cdots A_K^n] G_w, $$

(32)

where $n_s \in \mathbb{Z}_+$ is a parameter that can be increased to promote the feasibility of (31) at the cost of set complexity and the number of decision variables.

Note that the block-diagonal structure of $A_K$ and structure of $G_w$ ensures that $G_\varepsilon$ from (32) is separable and thus, $\mathcal{E}_0$ is a structured RPI set satisfying [13]. Once $\mathcal{E}_0(\mathbf{d}^*(k_0)) = \{G_\varepsilon \Phi_\varepsilon, c\}$ is computed, the subsystem-level RPI sets $\mathcal{E}_i = \{G_\varepsilon, c\}, i \in \mathcal{N}$, satisfying [13] can be obtained by projection.
A.3 Output Constraint Tightening for $C_0$

This section presents how the output constraint tightening used to compute $\hat{Y}_0(\delta(k_0))$, based on (26) and used in (11c), is integrated into the formulation of $P_0(x(k))$ through the addition of linear constraints and cost function terms. In (26), let the inter-sample tightened output constraint set be a zonotope such that $\hat{Y}_0 = \{\hat{G}_y, \hat{c}_y\}$ with known generator matrix $\hat{G}_y \in \mathbb{R}^{(n+m) \times n_y}$ and center $\hat{c}_y \in \mathbb{R}^{n+m}$. Let the tightened output constraint set be a scaled zonotope such that $\tilde{Y}_0(\delta(k_0)) = \{\tilde{G}_y \Phi_y, \tilde{c}_y\}$, where $\tilde{G}_y \in \mathbb{R}^{(n+m) \times n_y}$ is an a priori chosen nominal generator matrix and $\Phi_y$ is a diagonal scaling matrix satisfying $\Phi_y = diag(\phi_y)$ with $\phi_{yi} \geq 0, \forall i \in [1, n_y]$.

Following the approach from [18], based on the one-step Pontryagin difference computation from [17] and the zonotope containment conditions from [23], the decision variables $\tilde{c}_y \in \mathbb{R}^{n+m}$, $\Phi_y \in \mathbb{R}^{n_y \times n_y}$, $\Gamma_y \in \mathbb{R}^{n_y \times (n_y + n_w + 2n_v)}$, and $\beta_y \in \mathbb{R}^{n_y}$ are added to $P_0(x(k))$ with linear constraints

$$
\begin{bmatrix}
\hat{G}_y \Phi_y & [G_z diag(\delta^*) G_z \Phi_z] \\
0 & [G_z diag(\delta^*) KG_z \Phi_z]
\end{bmatrix} = \tilde{G}_y \Gamma_y,
$$

(33a)

$$
\begin{aligned}
\tilde{c}_y - \hat{c}_y &= \tilde{G}_y \beta_y, \\
|\Gamma_y| \mathbf{1} \leq |\beta_y| \leq 1.
\end{aligned}
$$

(33b)

The cost function for $P_0(x(k))$ is modified to balance system performance with maximizing the size of $\hat{Y}_0(\delta(k_0))$ through the addition of the term $-||\phi_y||_p$, where $\phi_y$ is the vector of scaling variables along the diagonal of $\Phi_y$. Note that the choice of $\tilde{G}_y$ affects the quality of the inner-approximation of the Pontryagin difference from (26).

A.4 Terminal Constraint Tightening for $C_0$

The terminal constraint tightening used to compute $\hat{T}_0(\delta^*(k_0))$, based on (27) and used in (11d), is integrated into the formulation of $P_0(x(k))$ through the addition of linear constraints and cost function terms following the same approach used in the previous section for output constraint tightening, and thus is not repeated here for brevity.

A.5 Set containment condition ($\Delta Z \subseteq Pre(\Delta Z)$)

This section presents how the set containment $\Delta Z \subseteq Pre(\Delta Z)$ in (11) can be enforced using linear constraints. Assuming the invertibility of $A_D$, the precursor set is a zonotope defined as $Pre(\Delta Z) = \{G_p diag(\delta), 0\}$, with $G_p = [A_D^{-1}G_z - A_D^{-1}B_DG_z]$. Using the zonotope containment conditions from [23], the decision variable $\Gamma_p \in \mathbb{R}^{n_w \times n_z}$ is added to $P_0(x(k))$ with linear constraints

$$
G_z diag(\delta^*) = G_p \Gamma_p,
$$

(34a)

$$
|\Gamma_p| \mathbf{1} \leq diag(\delta) \mathbf{1}.
$$

(34b)

References

[1] Ye Wang, Vicenç Puig, and Gabriela Cembrano. Non-linear economic model predictive control of water distribution networks. *Journal of Process Control*, 56:23–34, 2017.

[2] Jinwoo Seok, Ilya Kolmanovsky, and Anouck Girard. Coordinated model predictive control of aircraft gas turbine engine and power system. *Journal of Guidance, Control, and Dynamics*, 40(10):2538–2555, 2017.

[3] Irfan Khan, Zhicheng Li, Yinliang Xu, and Wei Gu. Distributed control algorithm for optimal reactive power control in power grids. *International Journal of Electrical Power and Energy Systems*, 83:505–513, 2016.
[4] Muhammad Irfan, Jamshed Iqbal, Adeel Iqbal, Zahid Iqbal, Raja Ali Riaz, and Adeel Mehmood. Opportunities and challenges in control of smart grids – Pakistani perspective. *Renewable and Sustainable Energy Reviews*, 71 (2017):652–674, 2017.

[5] Wisdom Enang and Chris Bannister. Modelling and control of hybrid electric vehicles (A comprehensive review). *Renewable and Sustainable Energy Reviews*, 74:1210–1239, 2017.

[6] Wenqing Wang and Justin P Koeln. Hierarchical Multi-Timescale Energy Management for Hybrid-Electric Aircraft. *ASME Dynamic Systems and Control Conference*, 2020.

[7] Riccardo Scattolini. Architectures for distributed and hierarchical Model Predictive Control - A review. *Journal of Process Control*, 19:723–731, 2009.

[8] Justin P. Koeln, Vignesh Raghuraman, and Brandon M. Hencey. Vertical hierarchical MPC for constrained linear systems. *Automatica*, 113:108817, 2020.

[9] Davide Barcelli, Alberto Bemporad, and Giulio Ripaccioli. Hierarchical multi-rate control design for constrained linear systems. *Proceedings of the IEEE Conference on Decision and Control*, pages 5216–5221, 2010.

[10] M. Farina, X. Zhang, and R. Scattolini. A hierarchical MPC scheme for interconnected systems. *IFAC-PapersOnLine*, 50(1):12021–12026, 2017.

[11] Marcello Farina, X. Zhang, and Riccardo Scattolini. A hierarchical multi-rate MPC scheme for inter-connected systems. *Automatica*, 90:38–46, 2018.

[12] Chris Vermillion, Amor Menezes, and Ilya Kolmanovsky. Stable hierarchical model predictive control using an inner loop reference model and $\lambda$-contractive terminal constraint sets. *Automatica*, 50(1), 2014.

[13] Balaji Sampathnarayanan, Simona Onori, and Stephen Yurkovich. An optimal regulation strategy with disturbance rejection for energy management of hybrid electric vehicles. *Automatica*, 50:128–140, 2014.

[14] David B Doman. Rapid mission planning for aircraft thermal management. *AIAA Guidance, Navigation, and Control Conference*, page 1076, 2015.

[15] Arthur Richards and Jonathan P. How. Model Predictive Control of Vehicle Maneuvers with Guaranteed Completion Time and Robust Feasibility. *American Control Conference*, pages 4034–4040, 2003.

[16] Vignesh Raghuraman, Venkatraman Renganathan, Tyler H. Summers, and Justin P. Koeln. Hierarchical MPC with Coordinating Terminal Costs. *American Control Conference*, pages 4126–4133, 2020.

[17] Vignesh Raghuraman and Justin P. Koeln. Set operations and order reductions for constrained zonotopes. *arXiv:2009.06039v1*, 2020.

[18] Vignesh Raghuraman and Justin P. Koeln. Tube-based robust MPC with adjustable uncertainty sets using zonotopes. *American Control Conference*, pages 462–469, 2021.

[19] Justin P. Koeln and Brandon M. Hencey. Constrained Hierarchical MPC via Zonotopic Waysets. *American Control Conference*, pages 4237–4244, 2019.

[20] Riccardo Scattolini and Patrizio Colaneri. Hierarchical model predictive control. *Proceedings of the IEEE Conference on Decision and Control*, pages 4803–4808, 2007.
[21] D.Q. Mayne, M.M. Seron, and S.V. Raković. Robust Model Predictive Control of Constrained Linear Systems with Bounded Disturbances. *Automatica*, 41:219–224, 2005.

[22] F. Borrelli, A. Bemporad, and M. Morari. Predictive Control for Linear and Hybrid Systems. *Cambridge University Press*, 2011.

[23] Sadra Sadraddini and Russ Tedrake. Linear Encodings for Polytope Containment Problems. *Proceedings of the IEEE Conference on Decision and Control*, pages 4367–4372, 2019.