Dynamic Incentive Selection for Hierarchical Convex Model Predictive Control

Akshay Thirugnanam and Koushil Sreenath

***Abstract*— In this paper, we discuss incentive design for hierarchical model predictive control (MPC) systems viewed as Stackelberg games. We consider a hierarchical MPC formulation where, given lower-level convex MPCs (LoMPCs), the upper-level system solves a bilevel MPC (BiMPC) subject to the constraint that the lower-level systems act optimally. Such problems are challenging due to optimality constraints in the BiMPC formulation. We analyze an incentive Stackelberg game variation, where the BiMPC provides additional incentives for the LoMPC cost, which grants the BiMPC influence over the LoMPC inputs. We show that for such problems, the BiMPC can be reformulated as a simpler MPC problem, and the optimal incentives can be iteratively computed without knowing the LoMPC costs. For the case of multiple LoMPCs, we propose an algorithm that finds bounded suboptimal solutions for the BiMPC. We demonstrate our method for a dynamic pricing example, where an independent system operator (ISO) sets the electricity prices for electric vehicle (EV) charging with the goal of minimizing a social cost and satisfying electricity generation constraints. Our method scales well to large EV population sizes.**

***Index Terms*— Convex model predictive control, bilevel optimization, incentive Stackelberg games, convex analysis, dynamic pricing, electric vehicle charging.**

I. Introduction

Complex control subsystems that interact with each other can be broadly classified into two categories, depending on the type of information flow: distributed and hierarchical. In distributed systems, information is passed between local regulators, while in hierarchical systems, a high-level coordinating algorithm passes information to local regulators; see [1] for classifications. Stackelberg games are leader-follower games where the leader announces its strategy to the followers, with the goal of minimizing the leader cost function [2]. We consider a hierarchical MPC problem where each lower-level system solves a convex MPC problem (LoMPC) to determine its control input. The upper-level system solves a bilevel MPC problem (BiMPC), which selects the upper-level system inputs subject to the constraint that the lower-level system inputs are optimal for the LoMPC. Such bilevel optimization problems can be viewed as Stackelberg games and are challenging due to the implicit optimality constraints. In this paper, we analyze an incentive Stackelberg game variation, where the BiMPC provides additional incentives to the LoMPC that allow the BiMPC to steer the LoMPC inputs.

Submitted to the editors February 6, 2025. This work was supported in part through funding from the Tsinghua-Berkeley Shenzhen Institute (TBSI) program.

The authors are with the Department of Mechanical Engineering, UC Berkeley, CA 94720, USA. {akshay_t, koushils}@berkeley.edu

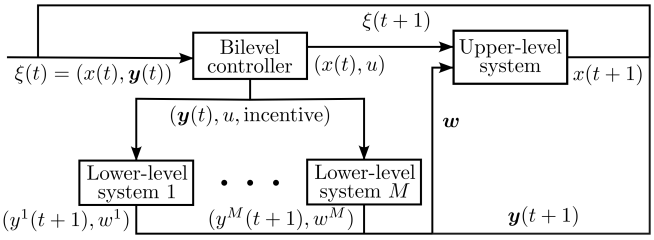


Fig. 1: A flowchart depicting the incentive hierarchical MPC formulation. Given the current state $\xi(t) = (x(t), \boldsymbol{y}(t))$ (x and $\boldsymbol{y} = (y^1, ..., y^M)$ are the upper and lower-level system states), the bilevel controller (the BiMPC) provides upper-level input u and an incentive to the lower-level systems, which compute the lower-level input $\boldsymbol{w}$ using the LoMPCs. We assume that the BiMPC can query the lower-level system inputs for different incentives. The controller output $(u, \boldsymbol{w})$ is then used to update the upper-level system state. The BiMPC and the LoMPCs together comprise the hierarchical MPC problem. In the EV charging example considered in Secs. II-B and VI, the BiMPC is solved by an ISO to determine the amount of electricity generation, while the LoMPCs are solved by EVs to determine charging rates. The incentive provided by the ISO is the unit price of electricity.

A. Related Work

1) Distributed MPC: In distributed MPC, the information shared between the local regulators consists of predicted system states or control inputs [1], [3]. The local regulators can be cooperative [4], [5], or non-cooperative [6]. Existing works also consider sparse communication topologies [7]. Some standard solution methods for distributed MPCs are robust MPC techniques [8] (for non-cooperative settings) and operator splitting methods like ADMM [9] (for cooperative settings). Control theoretical properties such as stability have also been studied for distributed MPC [10].

2) Hierarchical MPC: In hierarchical MPC, the high-level coordinator shares its predicted control signal with the local regulators, and the goal is to optimize a high-level cost function [11], [12]. Solution techniques for hierarchical MPC include ADMM [13], [14], multiparametric programming [15], waypoint tracking [16], [17], tube-MPC [18]–[20], and aggregation methods [21]. Price coordination is another class of methods relevant to our work, which uses Lagrange multipliers to link constraints across various subsystems. The prices are iteratively computed to ensure global consensus among all the participating subsystems [4], [7], [12], [22]–[24].

3) Incentive Stackelberg Games: Stackelberg games can be formulated as bilevel optimization problems, and optimal leader strategies can be obtained by reformulating the bilevel problem to a single-level problem [25]–[27]. The control theoretical properties of hierarchical MPC as a Stackelberg game are studied in [28]. A subclass of Stackelberg games is incentive Stackelberg games, where, in addition to the leader strategy, an incentive is provided to the follower by the leader [29]–[31]. The game is said to be *incentive controllable* if the leader can find incentives such that, given the leader strategy, the optimal follower strategy coincides with the optimal team strategy [30] (also see [29, Sec. 2]). In this paper, we extend the discussion in [28] to incentive Stackelberg games. We define and discuss sufficient conditions for incentive controllability from the perspective of MPC.

4) Dynamic Pricing for EV Charging: We consider a dynamic price control example for electric vehicle (EV) charging to demonstrate the algorithm proposed in this paper. Dynamic price control is one approach to reconcile the mismatch between electricity supply and demand at different times of the day and can be formulated as a Stackelberg game [25], [28], [32], [33]. Dynamic price control aims to reduce the spikes in electricity generation by shifting the electricity demand to different times of the day. Ideally, the electricity consumption pattern obtained using dynamic pricing achieves demand valley-filling [32], [33].

B. Contributions

The contributions of our paper are as follows: (a) We extend the discussion in [28] on hierarchical MPC as Stackelberg games to incentive Stackelberg games. For the hierarchical MPC problem, we define incentive controllability and team-optimal solutions. An incentive that achieves the team-optimal solution is defined as an optimal incentive. We show that for the case of a single LoMPC, the optimal linear-convex incentives can be found iteratively without knowing the LoMPC cost function. Our analysis relies on tools from convex analysis and duality theory. (b) For the case of multiple LoMPCs, we compute linear-convex incentives that achieve the team-optimal solution up to an error bound. We provide a single-level robust MPC reformulation of the BiMPC, whose optimal solution has bounded error compared to the team-optimal solution. (c) We demonstrate our algorithm for dynamic price control for EV charging. Our algorithm scales well with the EV population size and does not require the EV cost function.

C. Paper Structure

The paper is structured as follows: Sec. II provides background on convex analysis and motivates and states the hierarchical MPC problem. Sec. III defines linear incentives and incentive controllability for the hierarchical MPC and provides an iterative algorithm to calculate the optimal incentives. Sec. IV extends the analysis in Sec. III to linear-convex incentives. Sec. V considers the case where the hierarchical MPC problem with linear-convex incentives consists of multiple LoMPCs and provides a robust MPC formulation. A dynamic price control example for EV charging is considered in Sec. VI and conclusions are presented in Sec. VII.

D. Notations

The following notations are used throughout the paper: $\bar{\mathbb{R}} = \mathbb{R} \cup \{-\infty, +\infty\}$ is the set of extended real numbers. The Minkowski sum of two sets $\mathcal{A}, \mathcal{B} \subset \mathbb{R}^n$ is $\mathcal{A} + \mathcal{B} = \{a + b : a \in \mathcal{A}, b \in \mathcal{B}\}$. $\langle \cdot, \cdot \rangle$ represents an arbitrary inner product on the vector space considered, and $\|\cdot\|$ is the corresponding norm. Note that the transpose of a linear map $A : \mathbb{R}^n \to \mathbb{R}^m$, $A^\top$, depends on the inner products defined on the domain $\mathbb{R}^n$ and co-domain $\mathbb{R}^m$. $\operatorname{dom} f$ is the domain of the function f. For a set $\mathcal{C}$, $\operatorname{relint} \mathcal{C}$ is the relative interior of $\mathcal{C}$ [34, Sec. 2.1.3]. $\mathcal{I}_\mathcal{C} : \mathbb{R}^n \to \bar{\mathbb{R}}$ is the indicator function of a set $\mathcal{C} \subset \mathbb{R}^n$, where $\mathcal{I}_\mathcal{C}(x) = 0 \ \forall x \in \mathcal{C}$ and $\mathcal{I}_\mathcal{C}(x) = \infty \ \forall x \notin \mathcal{C}$. We also define $\langle N \rangle := \{0, ..., N-1\}$ and $[N] := \{0, ..., N\}$. For a function $\phi : \mathbb{R}^n \to \mathbb{R}^r$ and $x \in \operatorname{dom} \phi$, $D\phi(x) : \mathbb{R}^n \to \mathbb{R}^r$ is the derivative of ϕ. For a real-valued function $f : \mathbb{R}^n \to \mathbb{R}$, ∇f is the gradient of f (when f is differentiable) and ∂f is the subdifferential of f (see Sec. II-A.1). Subscripts such as $D_x \phi, \partial_x f$ denote the differentiation variable.

II. Background and Problem Formulation

A. Convex Analysis

For a function $f : \mathbb{R}^n \to \bar{\mathbb{R}}$, the epigraph of f is defined as $\operatorname{epi} f = \{(x, t) \in \mathbb{R}^{n+1} : f(x) \leq t, x \in \operatorname{dom} f\}$. Note that $\operatorname{epi} f \subset \mathbb{R}^{n+1}$. A function $f : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is a closed proper convex function if $\operatorname{epi} f$ is a closed, nonempty, and convex set respectively [35, Sec. 4, Sec. 7].

1) Subdifferential of a Convex Function: The subdifferential of a convex function $f : \mathbb{R}^n \to \bar{\mathbb{R}}$ at x is [35, Sec. 23]

$$\partial f(x) = \{v \in \mathbb{R}^n : f(z) \geq f(x) + \langle v, z - x \rangle, \ \forall z\}. \quad (1)$$

∂f is called the subdifferential of f. For all x, $\partial f(x)$ is a closed, convex set [35, Sec. 23]. A proper convex function f is subdifferentiable on $\operatorname{relint}(\operatorname{dom} f)$, and $\partial f(x)$ is non-empty and bounded if and only if $x \in \operatorname{int}(\operatorname{dom} f)$ [35, Th. 23.4].

2) Conjugate of a Convex Function: For a convex function f, its conjugate $f^* : \mathbb{R}^n \to \bar{\mathbb{R}}$ is defined as [35, Sec. 12]

$$f^*(v) = \sup_x \ (\langle x, v \rangle - f(x)). \quad (2)$$

For a proper convex function f, f^* is a closed proper convex function [35, Thm. 12.2]. For a closed proper convex function f, $f^{**} = f$ [35, Cor. 12.2.1], and the subdifferential of f^* is given by [35, Thm. 23.5]

$$\partial f^*(v) = \operatorname{argmax}_x \ (\langle x, v \rangle - f(x)). \quad (3)$$

3) Strong Convexity and Lipschitz Smoothness: A proper function $f : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is strongly convex with modulus $m > 0$ (abbreviated as m-strongly convex), if $\operatorname{dom} f$ is convex and for all $x \in \operatorname{dom} f$ and $v \in \partial f(x)$,

$$f(z) \geq f(x) + \langle v, z - x \rangle + (m/2)\|z - x\|^2, \quad \forall z. \quad (4)$$

A function f is m-strongly convex if and only if $f - (m/2)\|\cdot\|^2$ is convex [36, Prop. IV.1.1.2]. Strongly convex functions with closed domains have unique minima. So by (3), the conjugate of a strongly convex function is differentiable (see Prop. 1 below). A differentiable function $f : \mathbb{R}^n \to \bar{\mathbb{R}}$ is Lipschitz

smooth with constant $L > 0$ (abbreviated as L-smooth) if ∇f is Lipschitz continuous with constant L on $\operatorname{dom} f$, i.e.,

$$\|\nabla f(x^1) - \nabla f(x^2)\| \leq L\|x^1 - x^2\|, \ \forall x^1, x^2 \in \operatorname{dom} f. \quad (5)$$

There is a duality between strongly convex functions and Lipschitz smooth convex functions as follows.

Proposition 1. *[37, Thm. X.4.2.1, Thm. X.4.2.2] If a proper function $f : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ is m-strongly convex, then $\operatorname{dom} f^* = \mathbb{R}^n$ and f^* is $(1/m)$-smooth.*

4) Dual Cones: Let $\mathcal{K} \subset \mathbb{R}^n$ be a closed convex cone. Then, $\mathcal{K}$ defines a relation $\preceq_{\mathcal{K}}$ on $\mathbb{R}^n$ as follows: $x^1 \preceq_{\mathcal{K}} x^2$ if and only if $x^2 - x^1 \in \mathcal{K}$ [34, Sec. 2.4]. The dual cone $\mathcal{K}^*$ of a cone $\mathcal{K}$ is defined as [34, Sec. 2.6]

$$\mathcal{K}^* = \{v : \langle v, x\rangle \geq 0, \forall x \in \mathcal{K}\}. \quad (6)$$

The dual cone $\mathcal{K}^*$ depends on the inner product chosen on $\mathbb{R}^n$. For any cone $\mathcal{K}$, $\mathcal{K}^*$ is a closed convex cone.

5) Convex Vector Functions: Let $\mathcal{K} \subset \mathbb{R}^m$ be a closed convex cone. A function $f : \mathbb{R}^n \to \mathbb{R}^m$ is called $\mathcal{K}$-convex if $\operatorname{dom} f$ is convex and for all $x^1, x^2 \in \operatorname{dom} f$ and $\alpha \in [0, 1]$,

$$f(\alpha x^1 + (1-\alpha)x^2) \preceq_{\mathcal{K}} \alpha f(x^1) + (1-\alpha) f(x^2). \quad (7)$$

For any $\mathcal{K}$-convex function f and $\lambda \succeq_{\mathcal{K}^*} 0$, $\langle \lambda, f(\cdot)\rangle$ is a convex function [34, Sec. 3.6.2].

In the subsequent sections, we will use convex analysis to compute optimal incentives for the hierarchical MPC problem. Next, we motivate and describe the hierarchical MPC problem.

B. Motivating Example

As a motivating example for the hierarchical MPC formulation in the next subsection, we consider the charging of a population of EVs. Given the unit price of electricity, each EV determines its electricity consumption by considering the total price of electricity, battery degradation, and current state of charge (SoC). For the same electricity price, each EV can have different consumption depending on its SoC. An independent system operator (ISO) determines the amount of electricity generated, the amount stored in a storage battery, and the electricity price. The ISO's goal is to minimize a social cost, which comprises the cost of electricity generation and the cost of not meeting the EV consumption demand. In particular, the ISO's cost only considers the average EV consumption demand (computed using current EV SoCs); for the same average consumption, individual consumption may depend on the EV SoC. Similarly, the ISO determines electricity generation based only on the average consumption.

Such a hierarchical problem is an incentive Stackelberg game (see Sec. I-A.3), where the leader (the ISO) provides incentives (electricity prices) to the followers (the EV population) to minimize the social cost. The following subsection provides a general hierarchical MPC formulation applicable to the above example. In Secs. III-V, we develop an algorithm to solve the hierarchical MPC problem, and in Sec. VI, we apply our method to the EV charging example. We will occasionally refer to the EV charging example to provide intuition.

C. Hierarchical MPC Formulation

We consider a hierarchical control system consisting of multiple lower-level systems and an upper-level system as a generalization of the dynamics in [28]. Let $\mathcal{M}$ be an index set denoting the set of lower-level systems, with $M := |\mathcal{M}|$. For $i \in \mathcal{M}$, the i-th lower-level system has state $y^i \in \mathbb{R}^\rho$ and input $(u, w^i) \in \mathbb{R}^q \times \mathbb{R}^\gamma$. We define $\boldsymbol{w} := (w^1, ..., w^M)$. The upper-level system has state $x \in \mathbb{R}^p$ and input $\nu := (u, \boldsymbol{w})$. The hierarchical system has coupled linear dynamics given by,

$$\begin{aligned} x_{k+1} &= A^u x_k + B_1^u u_k + B_2^u w_k, \\ y_{k+1}^i &= A^l y_k^i + B_1^l u_k + B_2^l w_k^i, \quad i \in \mathcal{M}, \\ w_k &:= \frac{1}{M}\sum_{i\in\mathcal{M}} w_k^i, \end{aligned} \quad (8)$$

where the superscripts 'u' and 'l' refer to the upper and lower-level systems, respectively. Variable subscripts represent time, and superscripts are used for the index set $\mathcal{M}$. The superscript $(\cdot)^*$ denotes an optimal value, while the superscript $(\cdot)^{(k)}$ denotes the k-th iterate of a sequence. We define $\boldsymbol{y} := (y^1, ..., y^M)$ as the joint state of all the lower-level systems, and $\xi := (x, \boldsymbol{y})$ as the state of the entire system. Note that the upper-level system dynamics depend on the average w_k, similar to the EV charging example in Sec. II-B.

1) Lower-level MPC Problem: The i-th lower-level MPC (LoMPC) problem with horizon N optimizes the input sequence $w_{0:N-1}^i = (w_0^i, ..., w_{N-1}^i) \in \mathbb{R}^{\gamma N}$, given the initial state ξ_0 and the input sequence $u_{0:N-1} = (u_0, ..., u_{N-1}) \in \mathbb{R}^{qN}$. The symbols ξ, x, $\boldsymbol{y}$, ν, u, and $\boldsymbol{w}$ are overloaded to also represent $\xi_{0:N}$, $x_{0:N}$, $\boldsymbol{y}_{0:N}$, $\nu_{0:N-1}$, $u_{0:N-1}$, and $\boldsymbol{w}_{0:N-1}$ respectively. The variables used in the hierarchical MPC are tabulated in Tab. I. The i-th LoMPC problem is given by,

$$\min_{y^i, w^i} \ g^i(y^i, w^i; \xi_0, u), \quad (9a)$$

$$\text{s.t} \quad y_{k+1}^i = A^l y_k^i + B_1^l u_k + B_2^l w_k^i, \quad k \in \langle N\rangle, \quad (9b)$$

$$y_k^i \in \mathcal{Y}, \ w_k^i \in \mathcal{W}, \qquad k \in \langle N\rangle, \quad (9c)$$

$$y_N^i \in \mathcal{Y}_\Omega, \quad (9d)$$

where $\mathcal{Y}$, $\mathcal{Y}_\Omega \subset \mathbb{R}^\rho$ and $\mathcal{W} \subset \mathbb{R}^\gamma$ are closed convex sets, and $\langle N\rangle := \{0, ..., N-1\}$. The function g^i is parametrized by the input u from the bilevel MPC and ξ_0. The constraints in (9) can be added to the cost using the indicator function (see Sec. I-D) to obtain a reduced LoMPC problem as follows:

$$w^{i*}(\xi_0, u) := \operatorname{argmin}_{w^i} \ g^i(w^i; \xi_0, u), \quad (10)$$

where the symbol g^i is overloaded to represent the new cost function, where the constraints from (9) are added to the cost using the indicator function. We will work with the simple formulation of the cost function, (10), for the theory and use the full formulation, (9), for the implementation. The assumptions on g^i are as follows:

Assumption 1. *(Properties of g^i) $g^i(\cdot\,; \xi_0, u)$ is a closed proper function and m-strongly convex, for all (ξ_0, u). Further, $g^i(\cdot\,; \xi_0, u)$ is subdifferentiable on its domain for each (ξ_0, u).*

Since $g^i(\cdot\,; \xi_0, u)$ is closed and strongly convex, a unique optimum for w^i exists, given (9) is feasible (see Sec. II-A.3).

For the EV charging example in Secs. II-B and VI, y_k^i is the SoC and w_k^i is the charging rate of EV i at time k. The LoMPC cost function g^i comprises tracking costs for maximum SoC and costs for high charging rates.

2) Bilevel MPC Problem: Given the initial state ξ_0, the bilevel MPC (BiMPC) problem optimizes the input sequence $\nu = (u, \boldsymbol{w})$, subject to the constraint that w^i is obtained as the optimal solution of the LoMPC problem (10). In other words, the BiMPC accounts for the fact that each LoMPC solves an optimization problem to determine its input. The BiMPC problem is given by,

$$\min_{x,\nu=(u,\boldsymbol{w})} \; f(x,u,w;\xi_0), \tag{11a}$$

$$\text{s.t} \quad w^i = \operatorname{argmin}_{w^i} \; g^i(w^i;\xi_0,u), \qquad i \in \mathcal{M}, \tag{11b}$$

$$x_{k+1} = A^u x_k + B_1^u u_k + B_2^u w_k, \quad k \in \langle N \rangle, \tag{11c}$$

$$w_k = \frac{1}{M}\sum_{i\in\mathcal{M}} w_k^i, \qquad k \in \langle N \rangle, \tag{11d}$$

$$x_k \in \mathcal{X}, \; u_k \in \mathcal{U}, \qquad k \in \langle N \rangle, \tag{11e}$$

$$x_N \in \mathcal{X}_\Omega, \tag{11f}$$

where $\mathcal{X}$, $\mathcal{X}_\Omega \subset \mathbb{R}^p$ and $\mathcal{U} \subset \mathbb{R}^q$ are closed convex sets, and f is jointly continuous in (x,u,w). Note that the cost function f depends on the average $w = (w_1, ..., w_{N-1})$. Similar to the LoMPC problem, the constraints (11c), (11e), and (11f) can be added to the cost (11a) using the indicator function to obtain the reduced BiMPC problem as follows:

$$\nu^*(\xi_0) \in \operatorname*{argmin}_{\nu=(u,\boldsymbol{w})} \; f(u,w;\xi_0), \tag{12a}$$

$$\text{s.t} \quad w^i = \operatorname{argmin}_{w^i} \; g^i(w^i;\xi_0,u), i \in \mathcal{M}, \tag{12b}$$

$$w_k = \frac{1}{M}\sum_{i\in\mathcal{M}} w_k^i, \qquad k \in \langle N \rangle. \tag{12c}$$

The optimization problems (10) and (12) define the hierarchical MPC problem. At each time step, the current state $\xi(t)$ is provided to the BiMPC (12), and the first input ν_0^* from the optimal input sequence ν^* is applied to get the next state. Throughout the paper, we assume that (12) is feasible and has an optimal solution for all ξ_0.

For the EV charging example in Secs. II-B and VI, x_k is the SoC of the storage battery controlled by the ISO at time k and u_k is the amount of electricity generated. The dynamics (11c) corresponds to energy balance, and the cost function f is a social cost comprising electricity generation cost and EV electricity demand cost.

Dividing the problem of control selection for the hierarchical system into a multi-level problem such as (12) can result in a loss of controllability or stabilizability properties of the overall hierarchical system if the cost functions g^i are selected inappropriately [28].

The hierarchical MPC problem given by (10) and (12) is a Stackelberg game, where the BiMPC (12) is the leader, and the LoMPCs (10) are the followers. As discussed in Sec. I-A.3, we consider the incentive Stackelberg game variation, where the BiMPC provides an additional incentive to the LoMPCs. Incentive variables impose penalties (or rewards) on the LoMPCs in addition to the cost function g^i and thus grant the BiMPC influence over the LoMPC output. Fig. 1 depicts a flowchart of the incentive hierarchical MPC problem. We make the following assumption about the incentive hierarchical MPC problem, formally defined in Sec. III-A.

TABLE I: Hierarchical MPC variable definitions

Variable	Domain	Definition
(y_k^i, w_k^i, u_k)	$\mathbb{R}^{\rho+\gamma+q}$	State and input of i-th lower-level system at time k
(y^i, w^i)	$\mathbb{R}^{\rho(N+1)+\gamma N}$	$((y_0^i, ..., y_N^i), (w_0^i, ..., w_{N-1}^i))$
$(\boldsymbol{y}, \boldsymbol{w})$	$\mathbb{R}^{\rho(N+1)M+\gamma NM}$	$((y^1, ..., y^M), (w^1, ..., w^M))$
(w, w_k)	$\mathbb{R}^{\gamma N+\gamma}$	Averages: $(1/M\,\Sigma_i w^i, 1/M\,\Sigma_i w_k^i)$
$(x_k, u_k, \boldsymbol{w}_k)$	$\mathbb{R}^{p+q+\gamma M}$	State and input of upper-level system at time k
(x, u)	$\mathbb{R}^{p(N+1)+qN}$	$((x_0, ..., x_N), (u_0, ..., u_{N-1}))$
ξ	$\mathbb{R}^{(p+\rho M)(N+1)}$	$(x, \boldsymbol{y})$, state of the full system
ν	$\mathbb{R}^{(q+\gamma M)N}$	$(u, \boldsymbol{w})$, input of the full system

Assumption 2. *(Privacy of LoMPCs) The cost functions g^i are unknown to the BiMPC, apart from $\operatorname{dom} g^i$. The BiMPC can query the optimal solution w^{i*} from the i-th LoMPC for any given (ξ_0, u) and incentive.*

In the following two sections, we first consider the case of a single LoMPC (i.e., when $M = 1$). In Sec. III, we define a linear incentive structure and the notion of incentive controllability. We prove that an optimal incentive for the incentive hierarchical MPC can be computed iteratively, even when the LoMPC cost function is unknown. In Sec. IV, we extend the formulation in Sec. III to linear-convex incentives. In Sec. V, we consider the case of multiple LoMPCs (i.e., when $M > 1$) with linear-convex incentives and propose a robust formulation for the incentive hierarchical MPC.

III. Linear Incentives for Hierarchical MPC

In this section, we propose an incentive hierarchical MPC that adds a linear incentive term to the LoMPC cost function. For the EV charging example, the electricity price is a linear incentive for the EVs, as explained in the following subsection. We consider the case of a single LoMPC (i.e., when $M = 1$) and denote the state, input, and cost function of the LoMPC by y, w, and g, respectively. We define the notion of incentive controllability and show that the incentive hierarchical MPC satisfies this property. We also provide an iterative method to compute an optimal incentive without knowing the LoMPC cost function g.

A. Linear Incentives and Incentive Controllability

Let $g(w;\xi_0,u,\lambda)$ be the incentive LoMPC cost function given by,

$$g(w;\xi_0,u,\lambda) := g(w;\xi_0,u) + \langle \lambda, w \rangle, \tag{13}$$

where $\lambda \in \mathbb{R}^{\gamma N}$ is the incentive to the LoMPC and $\langle \lambda, w \rangle$ is the linear incentive term. For the EV charging example in Secs. II-B and VI, λ is the unit price of electricity, and $\langle \lambda, w \rangle$

is the total price paid by the EV. The LoMPC problem (10) with an incentive is as follows:

$$w^*(\xi_0, u, \lambda) := \operatorname{argmin}_w \ (g(w; \xi_0, u) + \langle \lambda, w \rangle), \tag{14}$$

where the BiMPC problem provides both u and λ as inputs to the LoMPC. As discussed in Sec. II-C.2, incentives impose additional penalties or rewards on the LoMPC and can be used to influence w^*. By Assumption 1, the cost function of the LoMPC problem (14) is a closed proper function, m-strongly convex, and subdifferentiable for all values of λ. So, there is a unique solution to (14) for all values of λ. The BiMPC problem (12) with an incentive is as follows:

$$(\nu^*, \lambda^*)(\xi_0) \in \underset{\nu=(u,w),\lambda}{\operatorname{argmin}} \ f(\nu; \xi_0), \tag{15a}$$

$$\text{s.t} \quad w = \operatorname{argmin}_w \ g(w; \xi_0, u, \lambda). \tag{15b}$$

The BiMPC dynamically chooses the incentive λ at each time step. The optimization problems (14) and (15) define the incentive hierarchical MPC problem. The incentive BiMPC problem (15) is nontrivial to solve due to the optimality constraint (15b). In the rest of the section, we discuss a solution method for (15). Following [38, Sec. 7.4] and [29, Def. 1], we first define a team-optimal solution for the incentive hierarchical MPC problem as follows.

Definition 1. *(Team-optimal solution) For a given ξ_0, a team-optimal solution of the incentive hierarchical MPC is an optimal solution $\nu^* = (u^*, w^*)$ of the following problem:*

$$\nu^*(\xi_0) \in \underset{\nu=(u,w)}{\operatorname{argmin}} \ f(\nu; \xi_0), \tag{16a}$$

$$\textit{s.t} \quad w \in \operatorname{dom} g(\cdot\,; \xi_0, u). \tag{16b}$$

Recall from Sec. II-C.1 that the constraint (16b) is equivalent to enforcing the state and input constraints (9b)-(9d). We assume that a team-optimal solution exists for all initial states ξ_0. Next, we define incentive controllability (this definition is similar to the one in [38, Sec. 7.4] and [30], adapted to the problem considered in this paper).

Definition 2. *(Incentive controllability) The incentive hierarchical MPC is called incentive controllable if, for all ξ_0 and corresponding team-optimal solutions $\nu^* = (u^*, w^*)$, there exists an incentive λ^* such that w^* is optimal for* (14) *with $(u, \lambda) = (u^*, \lambda^*)$, i.e., $w^* = w^*(\xi_0, u^*, \lambda^*)$.*

The following result states that the incentive hierarchical MPC is incentive controllable and shows that (15) can be solved using a two-step approach.

Theorem 1. *(Linear incentive controllability) Under Assumption 1, the incentive hierarchical MPC problem,* (14) *and* (15)*, is incentive controllable. Further, the incentive BiMPC* (15) *is equivalent to* (16)*. For any team-optimal solution $\nu^* = (u^*, w^*)$ and $-\lambda^* \in \partial_w g(w^*; \xi_0, u^*)$, (ν^*, λ^*) is an optimal solution of* (15)*.*

Proof. Under Assumption 1, $g(\cdot\,; \xi_0, u, \lambda)$ is a closed proper function, m-strongly convex, and subdifferentiable. Let $\nu^* = (u^*, w^*)$ be a team-optimal solution for some ξ_0. Then, w^* (uniquely) minimizes $g(w; \xi_0, u^*, \lambda)$ if and only if $-\lambda \in \partial_w g(w^*; \xi_0, u^*) \neq \emptyset$ [35, Thm. 23.5]. This means that if $-\lambda^* \in \partial_w g(w^*; \xi_0, u^*)$ then w^* is optimal for (14) with $(u, \lambda) = (u^*, \lambda^*)$. Thus, the incentive hierarchical MPC is incentive controllable.

Next, we show the equivalence of (15) and (16). For any feasible solution (ν, λ) of (15), ν is also feasible for (16) and has the same cost. Similarly, if $\nu = (u, w)$ is feasible for (16) and $-\lambda \in \partial_w g(w; \xi_0, u)$, then (ν, λ) is feasible for (15) and has the same cost. Thus, (15) and (16) are equivalent. □

Thm. 1 shows that (15) can be solved by first finding a team-optimal solution $\nu^* = (u^*, w^*)$ using (16) and then choosing $\lambda^* \in -\partial_w g(w^*; \xi_0, u^*)$ as an optimal incentive for (15). However, computing such an optimal incentive λ^* for a given w^* requires the explicit knowledge of g. In the following subsection, we show that optimal incentives can be computed iteratively without knowing g or its subdifferential.

B. Iterative Method for Optimal Incentives

To simplify notation, we assume that the team-optimal solution ν^* is available after solving the optimization problem (16) and we suppress the variables (ξ_0, u^*) in the cost function g, and write it as $g(w; \lambda) = g(w) + \langle \lambda, w \rangle$. Given w^*, the goal is to find λ^* such that $w^* = \operatorname{argmin}_w g(w; \lambda^*)$. Such an optimal incentive λ^* exists from Thm. 1, which shows that ∂g can be used to find λ^*. However, ∂g is unknown to the BiMPC (see Assumption 2). Instead, we can query the optimal solution of the LoMPC (14) for any incentive λ, which is computed as $w^*(\lambda) = \operatorname{argmin}_w g(w; \lambda)$. The following result shows that access to the LoMPC as a black-box solver is sufficient to iteratively compute incentives $\lambda^{(k)}$, such that $\operatorname{argmin}_w g(w; \lambda^{(k)}) \to w^*$ as $k \to \infty$.

Theorem 2. *(Iterative method for linear incentives) Let Assumption 1 hold, and $w^* \in \operatorname{dom} g$. Let $\lambda^{(k)}$, $w^{(k)}$ be iterates satisfying*

$$w^{(k)} = w^*(\lambda^{(k)}) = \operatorname{argmin}_w \ (g(w) + \langle \lambda^{(k)}, w \rangle), \tag{17a}$$

$$\lambda^{(k+1)} = \lambda^{(k)} + m(w^{(k)} - w^*), \tag{17b}$$

where m is the modulus of strong convexity of g. Then, $w^{(k)} \to w^*$ with the rate given by $\|w^{(k)} - w^*\| = O(1/\sqrt{mk})$.

Proof. We define a dual function $\bar{g}^*$ corresponding to the closed proper and m-strongly convex function g as follows:

$$\bar{g}^*(\lambda) = \min_w \ (g(w) + \langle \lambda, w \rangle). \tag{18}$$

Note that $\bar{g}^*(\lambda) = -g^*(-\lambda)$, where g^* is the conjugate of g (see Sec. II-A.2). So, by Prop. 1, $\bar{g}^*$ is a $(1/m)$-smooth concave function and $\operatorname{dom}(\bar{g}^*) = \mathbb{R}^{\gamma N}$. Now consider the function $\tilde{g}^*(\lambda) := \bar{g}^*(\lambda) - \langle w^*, \lambda \rangle$. $\tilde{g}^*$ is concave and $(1/m)$-smooth. The gradient of $\tilde{g}^*$ is given by (3) as

$$\nabla \tilde{g}^*(\lambda) = \nabla \bar{g}^*(\lambda) - w^* = \operatorname{argmin}_w \ (g(w) + \langle \lambda, w \rangle) - w^*.$$

So, if λ is a critical point of $\tilde{g}^*$, then it must satisfy

$$\nabla \tilde{g}^*(\lambda) = 0 \Rightarrow w^* = \operatorname{argmin}_w \ (g(w) + \langle \lambda, w \rangle).$$

In other words, $w^* = \operatorname{argmin}_w \ g(w; \lambda)$, meaning λ is an optimal incentive to achieve the solution w^*. At least one critical

point of $\tilde{g}^*$ exists because g is assumed to be subdifferentiable (Assumption 1) and $w^* \in \operatorname{dom} g$ (it is obtained from (16)).

Gradient ascent on $\tilde{g}^*$ with the constant step size of m converges because $\tilde{g}^*$ is $(1/m)$-smooth [39, Sec. 1.2.3]. Gradient ascent for $\tilde{g}^*$ is of the form:

$$\lambda^{(k+1)} = \lambda^{(k)} + m\nabla\tilde{g}^*(\lambda^{(k)}) = \lambda^{(k)} + m(w^*(\lambda^{(k)}) - w^*).$$

Then, replacing $w^*(\lambda^{(k)})$ with $w^{(k)}$ in the above equation using (17a), we obtain (17). Since $(w^{(k)} - w^*)$ is the gradient of $\tilde{g}^*$, it converges to zero with the rate given by $\|w^{(k)} - w^*\| = O(1/\sqrt{mk})$ [39, Eq. 1.2.22]. □

To summarize the results in this section, we have shown that adding a linear incentive to the LoMPC cost results in incentive controllability and allows the BiMPC to select an optimal incentive to achieve the team-optimal solution. Moreover, optimal incentives can be calculated iteratively without knowing the LoMPC cost. In the next section, we build upon the definitions and results in this section and consider a linear-convex incentive structure, which is a generalization of the linear incentive. We show that an analog of the iterative method, Thm. 2, can be used for linear-convex incentives.

IV. Linear-Convex Incentives

In this section, we consider linear-convex incentives of the form $\langle \lambda, \phi(w) \rangle$, which is linear in λ and convex in w. We consider the case of a single LoMPC (i.e., when $M = 1$) and assume that ϕ is known to the BiMPC. The results in this section follow the general outline for the linear incentive case in Sec. III. First, we state assumptions on the function ϕ and then define a dual function $\bar{g}^*$ in λ, similar to the conjugate function g^*. We then prove a duality result similar to Prop. 1. Finally, assuming incentive controllability, we derive an iterative method for linear-convex incentives. The results in this section are a generalization of the results in Sec. III, see Tab. II for a comparison.

A linear-convex incentive $\langle \lambda, \phi(w) \rangle$ is a generalization of the linear incentive $\langle \lambda, w \rangle$. For the EV charging example, linear-convex incentives can be used for nonlinear electricity pricing. Linear-convex incentives can also be used to encode the fact that unit electricity prices are nonnegative, as we will show in Sec. VI. The LoMPC problem (10) with a linear-convex incentive structure is as follows:

$$w^*(\xi_0, u, \lambda) := \operatorname{argmin}_w \; g(w; \xi_0, u) + \langle \lambda, \phi(w) \rangle. \tag{19}$$

Similar to Sec. III-B, we assume that the team-optimal solution ν^* is available after solving the optimization problem (16) and drop the dependence of g on (ξ_0, u^*) for the rest of this section. We define a linear-convex incentive as follows.

Definition 3. *(Linear-convex incentive) Let $\mathcal{K} \subset \mathbb{R}^r$ be a closed convex cone and $\phi : \mathbb{R}^{\gamma N} \to \mathbb{R}^r$ be a continuously differentiable $\mathcal{K}$-convex function (see Sec. II-A.5). Then the function $\bar{\phi} : \mathcal{K}^* \times \mathbb{R}^{\gamma N} \to \mathbb{R}$, with $\bar{\phi}(\lambda, w) := \langle \lambda, \phi(w) \rangle$ is called a linear-convex incentive.*

Remark 1. *If ϕ is a vector function that is convex in each of its components, then ϕ is $\mathcal{K}$-convex for the cone of nonnegative vectors, $\mathcal{K} = \mathbb{R}^r_+$. $\mathcal{K}$ is self-dual, i.e., $\mathcal{K}^* = \mathcal{K} = \mathbb{R}^r_+$.*

Since ϕ is a $\mathcal{K}$-convex function, $\langle \lambda, \phi(\cdot) \rangle$ is a convex function for any given $\lambda \succeq_{\mathcal{K}^*} 0$ (see Sec. II-A.5). So, for each $\lambda \succeq_{\mathcal{K}^*} 0$, $\langle \lambda, \phi(\cdot) \rangle$ is a closed proper convex function and differentiable at each $w \in \mathbb{R}^{\gamma N}$. For the EV charging example in Sec. II-B, we choose $\mathcal{K}$ as the set of nonnegative vectors $\mathbb{R}^r_+$. This corresponds to the fact that the unit price of electricity cannot be negative. This nonnegativity constraint motivates the linear-convex incentives used in this section.

Following the outline of Sec. III, we define a dual function $\bar{g}^* : \mathcal{K}^* \to \bar{\mathbb{R}}$, similar to (2), as follows:

$$\bar{g}^*(\lambda) := \min_w \; (g(w) + \langle \lambda, \phi(w) \rangle). \tag{20}$$

Note that $\bar{g}^*$ is similar to the dual function in (18). Tab. III tabulates the symbols in the paper, which are used consistently through Secs. III-V, and can be used as a quick reference. The minimum in w exists for each λ because $\langle \lambda, \phi(\cdot) \rangle + g$ is m-strongly convex. Now we prove a result on the differentiability of $\bar{g}^*$, similar to (3).

Lemma 1. *(Convexity and gradient of $-\bar{g}^*$) Under Assumption 1, Def. 3, and (20), $-\bar{g}^*$ is a proper convex function with $\operatorname{dom} \bar{g}^* = \mathcal{K}^*$. $-\bar{g}^*$ is differentiable for each $\lambda \in \mathcal{K}^*$, with*

$$\nabla(-\bar{g}^*)(\lambda) = -\phi(w^*(\lambda)), \tag{21}$$

where $w^*(\lambda)$ is the unique minimum of (20), i.e.,

$$w^*(\lambda) = \operatorname{argmin}_w \; (g(w) + \langle \lambda, \phi(w) \rangle). \tag{22}$$

Further, $w^*(\cdot)$ is a continuous function on $\mathcal{K}^*$.

Proof. The proof is provided in Appendix I-A. □

Lem. 1 shows that $-\bar{g}^*$ is a continuously differentiable convex function, with gradient given by $\nabla\bar{g}^*(\lambda) = \phi(w^*(\lambda))$. Similar to Prop. 1, the following result shows that $-\bar{g}^*$ is a Lipschitz smooth function. We use this result in Thm. 3 to show the convergence of an iterative method for calculating optimal incentives, similar to Thm. 2 (see Tab. II).

Lemma 2. *(Lipschitz smoothness of $-\bar{g}^*$) Under Assumption 1, Def. 3, and (20), $-\bar{g}^*$ satisfies the following inequality for any $\lambda, \bar{\lambda} \in \mathcal{K}^*$:*

$$\begin{aligned} -\bar{g}^*(\lambda) \leq -\bar{g}^*(\lambda; \bar{\lambda}) := &- \bar{g}^*(\bar{\lambda}) + \langle \lambda - \bar{\lambda}, -\phi(\bar{w}) \rangle \\ &+ 1/(2m)\|D\phi(\bar{w})^\top(\lambda - \bar{\lambda})\|^2, \end{aligned} \tag{23}$$

where $\bar{w} = w^*(\bar{\lambda}) = \operatorname{argmin}_w\{g(w) + \langle \bar{\lambda}, \phi(w) \rangle\}$, m is the modulus of strong convexity of g, and $D\phi$ is the Jacobian of the vector function ϕ. In particular, when restricted to any compact domain, $-\bar{g}^*$ is Lipschitz smooth.

Proof. The proof is provided in Appendix I-B. □

Lem. 2 shows that the dual function $-\bar{g}^*$ can be majorized, i.e., bounded from above, by a quadratic function $-\bar{g}^*(\cdot\,; \bar{\lambda})$ anchored at $\bar{\lambda}$. Lems. 1 and 2 discuss the properties of the dual function $\bar{g}^*$. In the rest of the section, we consider the incentive controllability property of the incentive hierarchical MPC and show that an iterative method can be used to compute an optimal incentive. Unlike for linear incentives, incentive controllability does not necessarily hold for linear-convex incentives. From Def. 2, incentive controllability holds

TABLE II: Properties of the dual functions and main results

Property	Single LoMPC ($M = 1$)		Multiple LoMPCs ($M > 1$)
	Linear incentives (Sec. III)	Linear-convex incentives (Sec. IV)	Linear-convex incentives (Sec. V)
Gradient of the dual function $\bar{g}^*$	Eq. (3)	Lem. 1	–
Lipschitz smoothness of $\bar{g}^*$	Prop. 1	Lem. 2	–
Incentive controllability (Def. 2)	Thm. 1	Assumption 3	Lem. 3
Iterative method for optimal incentives	Thm. 2	Thm. 3	Thm. 3
Incentive BiMPC reformulation	Def. 1	Def. 1	Thm. 4

TABLE III: Definitions for symbols used in the paper

Symbol	Definition	Meaning
$g^i(w^i; \xi_0, u)$	Sec. III: (10)	LoMPC cost (also $g^i(w^i)$)
$g^i(w^i; \xi_0, u, \lambda)$ $w^{i*}(\xi_0, u, \lambda)$	Sec. III: (13), (14) Sec. IV: (19) Sec. V: (31)	Incentive LoMPC cost and optimal solution (also $g^i(w^i; \lambda)$ and $w^{i*}(\lambda)$)
$(u^*, \hat{w}^*)(\xi_0)$	Sec. V: (38)	Team-optimal solution for robust BiMPC
$\bar{g}^*$, $\tilde{g}^*$	Sec. III: (18) Sec. IV: (20), (25) Sec. V: (35)	Concave dual functions (related to the conjugate function g^*, (2))
$\bar{g}^*(\cdot\,; \bar{\lambda})$ $(\tilde{g}^*(\cdot\,; \bar{\lambda}))$	Sec. IV: (23), (26)	Concave quadratic under-approximation of $\bar{g}^*$ ($\tilde{g}^*$) at $\bar{\lambda}$

if for all $w^* \in \operatorname{dom} g$, $\exists \lambda^* \in \mathcal{K}^*$ such that $w^* = \operatorname{argmin}_w\{g(w) + \langle \lambda^*, \phi(w)\rangle\}$. Using the optimality condition for $g(w) + \langle \lambda^*, \phi(w)\rangle$ [35, Thm. 23.8], w^* must satisfy

$$-D\phi(w^*)^\top \lambda^* \in \partial g(w^*), \tag{24}$$

for some $\lambda^* \in \mathcal{K}^*$. Using the above condition, we make the following assumption on the incentive structure.

Assumption 3. *(Linear-convex incentive controllability) ϕ satisfies the following properties:*
1) $\phi(w^1) \preceq_{\mathcal{K}} \phi(w^2) \Rightarrow w^1 = w^2,\ \forall w^1, w^2$.
2) $\forall w^* \in \operatorname{dom} g$, $\exists \lambda^* \in \mathcal{K}^*$, such that (24) *holds.*

Following the discussion above, incentive controllability holds for linear-convex incentives when Assumption 3 holds. In the following remark, we identify a family of functions ϕ that satisfy Assumption 3.

Remark 2. *One class of linear-convex incentives that always satisfies Assumption 3 is of the form $\langle \lambda_l, w\rangle + \langle \lambda_c, \phi_c(w)\rangle$, where ϕ_c is continuously differentiable and $\mathcal{K}_c$-convex for some closed convex cone $\mathcal{K}_c$, $\lambda_l \in \mathbb{R}^{\gamma N}$, $\lambda_c \in (\mathcal{K}_c)^*$, $\lambda = (\lambda_l, \lambda_c)$, and $\phi(w) = (w, \phi_c(w))$. Here, $\mathcal{K} = \{\mathbb{0}_{\gamma N}\} \times \mathcal{K}_c$, and $\mathcal{K}^* = \mathbb{R}^{\gamma N} \times (\mathcal{K}_c)^*$. Consequently, adding a linear-convex incentive to a linear incentive retains the incentive controllability property, Def. 2.*

We make the following observation about the dual function $-\bar{g}^*$. Given $w^* \in \operatorname{dom} g$, we define the function $\tilde{g}^*$ as follows:

$$\tilde{g}^*(\lambda) := \bar{g}^*(\lambda) - \langle \lambda, \phi(w^*)\rangle. \tag{25}$$

If λ is a critical point of $\tilde{g}^*$, then by Lem. 1,

$$\nabla \tilde{g}^*(\lambda) = 0 \ \Rightarrow\ \phi(w^*(\lambda)) - \phi(w^*) = 0.$$

By Assumption 3, this implies that $w^*(\lambda) = w^*$, i.e., λ is an optimal incentive for w^*. Thus, we claim that minimizing $-\tilde{g}^*$ results in an optimal incentive. By Lem. 2, for any $\lambda, \bar{\lambda} \in \mathcal{K}^*$,

$$-\tilde{g}^*(\lambda) \le -\tilde{g}^*(\lambda\,; \bar{\lambda}) := -\bar{g}^*(\lambda\,; \bar{\lambda}) + \langle \lambda, \phi(w^*)\rangle, \tag{26}$$

where $\bar{g}^*(\cdot\,; \bar{\lambda})$ is defined in (23) (see Tab. III for a summary). So, $-\tilde{g}^*$ is majorized by the quadratic function $-\tilde{g}^*(\cdot\,; \bar{\lambda})$. Thus, to minimize $-\tilde{g}^*$, we can follow an iterative method where we minimize the majorization $-\tilde{g}^*(\cdot\,; \lambda^{(k)})$ at the iterate $\lambda^{(k)}$ to obtain the next iterate $\lambda^{(k+1)}$. The following theorem, one of this paper's main results, formalizes the above discussion and guarantees the convergence of the majorization-minimization procedure.

Theorem 3. *(Iterative method for linear-convex incentives) Let Assumptions 1 and 3 hold, and $w^* \in \operatorname{dom} g$. Let $\lambda^{(k)}$ and $w^{(k)}$ be iterates satisfying*

$$w^{(k)} = w^*(\lambda^{(k)}) = \operatorname{argmin}_w \left(g(w) + \langle \lambda^{(k)}, \phi(w)\rangle\right), \tag{27a}$$

$$\lambda^{(k+1)} = \operatorname{argmin}_{\lambda \in \mathcal{K}^*} \left(\epsilon^{(k)} \|\lambda - \lambda^{(k)}\|^2 - \tilde{g}^*(\lambda; \lambda^{(k)})\right), \tag{27b}$$

where $\tilde{g}^*(\lambda; \lambda^{(k)}) := \bar{g}^*(\lambda; \lambda^{(k)}) - \langle \lambda, \phi(w^*)\rangle$, $\bar{g}^*(\cdot\,; \lambda^{(k)})$ is the quadratic function defined in (23)*, and $\epsilon^{(k)} > 0$ is a bounded regularization weight. If $\{\lambda^{(k)}\}$ is a bounded sequence, then $\lim_{k\to\infty} w^{(k)} = w^*$.*

At each iteration, there is a guaranteed dual cost decrease:

$$\tilde{g}^*(\lambda^{(k)}) - \tilde{g}^*(\lambda^{(k+1)}) \le -\tilde{g}^*(\lambda^{(k+1)}; \lambda^{(k)}) + \tilde{g}^*(\lambda^{(k)}; \lambda^{(k)}) + \epsilon^{(k)} \|\lambda^{(k+1)} - \lambda^{(k)}\|^2, \tag{28}$$

where $\tilde{g}^*(\lambda) := \bar{g}^*(\lambda) - \langle \lambda, \phi(w^*)\rangle$.

For any optimal incentive λ^*, and $\bar{\lambda} \in \mathcal{K}^*$ such that $D\phi(w^*)^\top(\bar{\lambda} - \lambda^*) = 0$, $\bar{\lambda}$ is also an optimal incentive.

Proof. The proof is provided in Appendix I-C. □

Similar to the iterative method for linear incentives, Thm. 2, the sequence $\{\lambda^{(k)}\}$ obtained from (27) need not converge. However, given an optimal incentive λ^* (obtained as a limit point of $\{\lambda^{(k)}\}$), Thm. 3 provides us a method to find other optimal incentives. If $\bar{\lambda} \in \mathcal{K}^*$ is such that $D\phi(w^*)^\top(\bar{\lambda} - \lambda^*) = 0$, then $\bar{\lambda}$ is also an optimal incentive. As noted in the following remark, we can use this property to find an optimal incentive that minimizes some cost function.

Remark 3. *(Optimal incentive regularization) Let λ^* be any optimal incentive obtained using* (27) *and $c \in \mathbb{R}^r$. Then any solution $\bar{\lambda}$ to the conic program*

$$\begin{aligned} &\min_{\lambda \in \mathcal{K}^*} \ \langle \lambda, c\rangle, \\ &\textit{s.t} \quad D\phi(w^*(\lambda^*))^\top(\lambda - \lambda^*) = 0, \end{aligned} \tag{29}$$

is also an optimal incentive. Eq. (29) *can be used to regularize along the set of optimal incentives. For instance, in the EV*

charging example considered in Sec. II-B, the incentive λ denotes the unit price of electricity, and we choose $c = \phi(w^*(\lambda^*))$ to minimize the total price paid by the EVs.

Remark 4. *Following Rem. 2, if we choose $\phi(w) = w$ and $\mathcal{K} = \{\mathbb{0}_{\gamma N}\}$, the linear-convex incentive $\langle \lambda, \phi(w)\rangle$ is the same as the linear incentive $\langle \lambda, w\rangle$. In this case, Lem. 1, Lem. 2, and Thm. 3 are equivalent to* (3)*, Prop. 1, and Thm. 2 respectively (see Tab. II for a comparison).*

To summarize the results in this section, we first define a linear-convex incentive structure in Def. 3 and a corresponding dual function in (20). We prove the convexity properties of the dual function in Lems. 1 and 2. Next, we show that the hierarchical MPC problem, with the incentive LoMPC (19), is incentive controllable when Assumption 3 holds. Finally, Thm. 3 shows that for any team-optimal solution $\nu^* = (u^*, w^*)$, an optimal incentive λ^* can be found iteratively, without the knowledge of the LoMPC cost function g.

In the following section, we consider the hierarchical MPC problem for the case of multiple LoMPCs (i.e., when $M > 1$), as defined in Sec. II-C. We use the tools developed in Secs. III and IV to propose an algorithm that solves a particular case of the incentive hierarchical MPC problem.

V. Multiple Lower-level MPCs

A. Robust BiMPC Formulation

In this section, we consider the hierarchical MPC problem (10) and (12) for the case of multiple LoMPCs. Recall that the LoMPC problems are indexed by the set $\mathcal{M}$ (with $M = |\mathcal{M}|$). First, we define the incentive BiMPC problem as an extension to (12) as follows (see Tab. I for variable definitions):

$$(\nu^*, \lambda^*)(\xi_0) \in \underset{\substack{\nu=(u,\boldsymbol{w}),\\ \lambda\in\mathcal{K}^*}}{\operatorname{argmin}} \; f(u, w; \xi_0), \tag{30a}$$

$$\text{s.t} \quad w^i = w^{i*}(\xi_0, u, \lambda), \; i \in \mathcal{M}, \tag{30b}$$

$$w = \frac{1}{M}\sum_{i\in\mathcal{M}} w^i, \tag{30c}$$

where $w^{i*}(\xi_0, u, \lambda) := \operatorname{argmin}_{w^i} g^i(w^i; \xi_0, u, \lambda)$. From (30b), we note that all LoMPCs are provided the same incentive $\lambda \in \mathcal{K}^*$. We assume an affine parametric form for the incentive LoMPC cost functions g^i:

$$\begin{aligned} g^i(w^i; \xi_0, u, \lambda) := \; & g(w^i; \xi_0, u) + \langle \lambda, \phi(w^i)\rangle \\ & + \langle \theta^i(\xi_0, u), w^i\rangle, \end{aligned} \tag{31}$$

where $\theta^i(\xi_0, u)$ is a parameter for the i-th LoMPC satisfying $\|\theta^i(\xi_0, u)\| \le \bar{\theta}(\xi_0) \; \forall i \in \mathcal{M}$, and $\langle \lambda, \phi(w)\rangle$ is a linear-convex incentive (see Def. 3) with a $\mathcal{K}$-convex function ϕ and $\lambda \in \mathcal{K}^*$. It can be shown that all LQR-like LoMPC problems have a parametric cost as given in (31) (see Rem. 6). In particular, the EV charging costs generally considered in literature can be written in the form (31) (see Sec. VI-A). Note that for the single LoMPC case considered in Secs. III and IV, we can choose $\theta^i \equiv 0$.

Assumption 4. *For all $i \in \mathcal{M}$, g^i has the parametric form given by* (31)*, where the cost function g satisfies Assumption 1 and $\|\theta^i(\xi_0, u)\| \le \bar{\theta}(\xi_0) \; \forall i \in \mathcal{M}$.*

Remark 5. *From Sec. II-C, the cost function g^i includes the state and input constraints for the i-th LoMPC. If Assumption 4 holds, then, from* (31)*, $\operatorname{dom} g^i(\cdot\,; \xi_0, u) = \operatorname{dom} g^j(\cdot\,; \xi_0, u)$ for all $i, j \in \mathcal{M}$ and (ξ_0, u). Therefore, to satisfy Assumption 4, the LoMPCs in $\mathcal{M}$ must not have any state constraints.*

Given Rem. 5, the following remark identifies a class of LoMPCs for which Assumption 4 holds.

Remark 6. *One class of LoMPC problems that always satisfy Assumption 4 is input-constrained LQRs with quadratic state cost and strongly convex input cost. Any such LQR problem can be expressed in batch form as [40, Sec. 8.2]*

$$\min_{w^i\in\mathcal{W}^N} \big((w^i)^\top Q w^i + \textstyle\sum_k g_w(w^i_k) + (y^i_0)^\top R w^i\big), \tag{32}$$

where $Q \succeq 0$, $\mathcal{W} \subset \mathbb{R}^\gamma$ is a closed convex set denoting the input constraints, and $g_w : \mathbb{R}^\gamma \to \mathbb{R}$ is a strongly convex function. In this case, $\theta^i(\xi_0, u) = R^\top y^i_0$. The EV LoMPC considered in Sec. VI is one such example.

For the rest of the section, we suppress the dependence of θ and $\bar{\theta}$ on (ξ_0, u). Since all LoMPCs are provided the same incentive $\lambda \in \mathbb{R}^r$ (from (30b)), incentive controllability is not guaranteed for the BiMPC problem (30) with multiple LoMPCs. However, we can show that under Assumption 4, incentive controllability holds with bounded error.

Lemma 3. *(Bounded incentive controllability) Let Assumptions 3 and 4 hold. Given any ξ_0, let $(u^*, \hat{w}^*) \in \operatorname{dom} f$ with $\hat{w}^* \in \operatorname{dom} g$, and $\lambda^* \in \mathcal{K}^*$ be such that $-D\phi(\hat{w}^*)^\top \lambda^* \in \partial_w g(\hat{w}^*; \xi_0, u^*)$ (by Assumption 3). Let, for $i \in \mathcal{M}$, w^{i*} be the optimal solution of the i-th LoMPC with the incentive λ^*:*

$$w^{i*} = \operatorname{argmin}_{w^i}(g(w^i; \xi_0, u^*) + \langle \lambda^*, \phi(w^i)\rangle + \langle \theta^i, w^i\rangle). \tag{33}$$

Then, $w^* = (1/M)\sum_{i\in\mathcal{M}} w^{i*} \in \operatorname{dom} g$ and

$$\big\|\hat{w}^* - w^*\big\| \le \bar{\theta}/m. \tag{34}$$

Proof. Corresponding to the incentive LoMPC cost function g^i in (31), we define the dual function, $\bar{g}^*$, as follows:

$$\bar{g}^*(\theta) := \min_w \; (g(w; \xi_0, u^*) + \langle \lambda^*, \phi(w)\rangle + \langle \theta, w\rangle). \tag{35}$$

Then, $\bar{g}^*(\theta) = -g^*(-\theta)$, where g^* is the conjugate function (see Sec. II-A.2) of the m-strongly convex function $g(\cdot\,; \xi_0, u^*) + \langle \lambda^*, \phi(\cdot)\rangle$. So, by Prop. 1, we have that $\bar{g}^*$ is a $(1/m)$-smooth concave function. By (3),

$$\nabla\bar{g}^*(\theta) = \operatorname{argmin}_w (g(w; \xi_0, u^*) + \langle \lambda^*, \phi(w)\rangle + \langle \theta, w\rangle). \tag{36}$$

Since $-D\phi(\hat{w}^*)^\top \lambda^* \in \partial_w g(\hat{w}^*; \xi_0, u^*)$, we have from the optimality condition for $g(w; \xi_0, u^*) + \langle \lambda^*, \phi(w)\rangle$ that [35, Thm. 23.8]

$$\hat{w}^* = \operatorname{argmin}_w \; (g(w; \xi_0, u^*) + \langle \lambda^*, \phi(w)\rangle). \tag{37}$$

Using (36), (33), (37), and the fact that $\bar{g}^*$ is $(1/m)$-smooth, we get that

$$\|\hat{w}^* - w^{i*}\| = \|\nabla\bar{g}^*(0) - \nabla\bar{g}^*(\theta^i)\| \le \|\theta^i\|/m \le \bar{\theta}/m.$$

Finally, (34) follows from the triangle inequality. Since $w^{i*} \in \operatorname{dom} g \; \forall i$ and $\operatorname{dom} g$ is convex, $w^* \in \operatorname{dom} g$. □

For any $w \in \operatorname{dom} g$, Lem. 3 proves the existence of an incentive λ^* such that the average LoMPC solution w^* with incentive λ^* has bounded error compared to w. Moreover, from (37), λ^* is an optimal incentive corresponding to w, when $\theta = 0$. Therefore, using Thm. 3, we can iteratively compute λ^* given (u^*, w). We also note that the error bound given by (34) is independent of the number of LoMPCs M.

In the case of a single LoMPC, as considered in Sec. III, the incentive controllability property for linear incentives leads to the equivalent formulation (16) of the incentive BiMPC (15). Similarly, we can show that the bounded incentive controllability property, Lem. 3, leads to a robust formulation of the incentive BiMPC (30). By Lem. 3, for any $w \in \operatorname{dom} g$, we can find an incentive such that the average LoMPC solution $w^* \in \operatorname{dom} g$ has bounded error compared to w. We must account for this error to ensure that (u^*, w^*) is also feasible for the incentive BiMPC (30).

Next, we describe a robust formulation of (30) using Lem. 3. The following main result is a generalization of Thm. 1 for the case of multiple LoMPCs (see Tab. II).

Theorem 4. *(Robust BiMPC formulation) Let Assumptions 3 and 4 hold. For any ξ_0, let $(u^*, \hat{w}^*)$ be an optimal solution of the following optimization problem:*

$$(u^*, \hat{w}^*) \in \underset{\nu=(u,\hat{w})}{\operatorname{argmin}} \; f(\nu; \xi_0), \tag{38a}$$

$$\text{s.t} \quad \hat{w} \in \operatorname{dom} g(\cdot\,; \xi_0, u), \tag{38b}$$

$$\{(u, \hat{w})\} + \mathcal{E} \subset \operatorname{dom} f(\cdot\,; \xi_0), \tag{38c}$$

where the addition between sets in (38c) *is the Minkowski sum (see Sec. I-D), and*

$$\mathcal{E} := \{(\mathbb{0}, w) \in \mathbb{R}^{qN} \times \mathbb{R}^{\gamma N} : \|w\| \leq \bar{\theta}/m\}. \tag{39}$$

Then, for $\lambda^* \in \mathcal{K}^*$ such that $-D\phi(\hat{w}^*)^\top \lambda^* \in \partial_w g(\hat{w}^*; \xi_0, u^*)$, $(u^*, \boldsymbol{w}^*, \lambda^*)$ is feasible for the incentive BiMPC (30) *and*

$$\left\|\hat{w}^* - w^*\right\| \leq \bar{\theta}/m,$$

where $\boldsymbol{w}^* = (w^{1*}, ..., w^{M*})$, $w^* = (1/M)\sum_{i\in\mathcal{M}} w^{i*}$, and

$$w^{i*} = \operatorname{argmin}_{w^i}(g(w^i; \xi_0, u^*) + \langle \lambda^*, \phi(w^i)\rangle + \langle \theta^i, w^i\rangle).$$

Proof. Since $\mathbb{0}_{(q+\gamma)N} \in \mathcal{E}$ and $(u^*, \hat{w}^*)$ satisfies the constraint (38c), $(u^*, \hat{w}^*) \in \operatorname{dom} f(\cdot\,; \xi_0)$. The assumptions of Lem. 3 are satisfied, and so (34) holds. Thus, by (34) and (38c),

$$(u^*, w^*) \in (u^*, \hat{w}^*) + \mathcal{E} \subset \operatorname{dom} f(\cdot\,; \xi_0).$$

Therefore, $(u^*, \boldsymbol{w}^*, \lambda^*)$ is feasible for the incentive BiMPC (30). □

The robust BiMPC (38) defines a team-optimal solution $(u^*, \hat{w}^*)$ for the incentive hierarchical problem given by (30) and (19). Since the same incentive is provided to all LoMPCs, it might not be possible to obtain the team-optimal solution $(u^*, \hat{w}^*)$ for any incentive. However, Thm. 4 shows that we can achieve a solution (u^*, w^*) that has bounded error compared to the team-optimal solution. The robust formulation (38) shrinks the domain of the BiMPC problem to account for this error and thus guarantees that the obtained solution (u^*, w^*) satisfies the BiMPC constraints.

Remark 7. *(Explicit robust BiMPC formulation) Since the state and input constraints of the BiMPC are convex (see Sec. II-C), $\operatorname{dom} f$ is convex. Thus, the robustness constraint* (38c) *is a convex constraint. When the state constraints of the BiMPC problem* (11) *are polytopic or quadratic, we can explicitly write the robustness constraint* (38c) *using duality theory. We show this explicit formulation in App. II.*

Remark 8. *(Robustness horizon) The robust BiMPC* (38) *accounts for the error in the optimization variable $\hat{w}$ for the entire horizon N. The robustness horizon $N_r > 0$ can be defined as the number of time steps for which the error in $\hat{w}$ is considered. For $N_r \leq N$, we modify the robust BiMPC* (38) *by redefining $\mathcal{E}$ in* (38c) *as follows:*

$$\mathcal{E} := \{(\mathbb{0}, w_r, \mathbb{0}) \in \mathbb{R}^{qN} \times \mathbb{R}^{\gamma N_r} \times \mathbb{R}^{\gamma(N-N_r)} : \|w_r\| \leq k\bar{\theta}/m\},$$

where $k = \max_w\{\|w_{0:N_r-1}\| : \|w\| \leq 1\}$. Using $N_r < N$ expands the feasible set of (38) *but only guarantees the satisfaction of the BiMPC constraints for N_r time steps. In other words, the state x_k^* obtained from $(u^*, \boldsymbol{w}^*)$ might violate the BiMPC state constraints for $k > N_r$. However, since the hierarchical MPC problem is solved with state feedback at each time step (see Fig. 1), we can choose $N_r < N$, as long as* (38) *is feasible.*

Remark 9. *We can extend the incentive BiMPC* (30) *to account for a partition of the set of LoMPCs $\mathcal{M}$ as $\sqcup_j \mathcal{M}_j$. Here, the BiMPC assigns a different incentive λ^j to each set of LoMPCs $\mathcal{M}_j$. We can formulate a robust BiMPC similar to* (38) *by considering the error bounds corresponding to each partition.*

To summarize the results in this section, we first specify the assumptions on the incentive hierarchical MPC problem in Assumption 4. Then, we show that although incentive controllability does not hold for the case of multiple LoMPCs, we can still guarantee bounded incentive controllability, Lem. 3. Finally, we use the bounded incentive controllability property to formulate a robust BiMPC, Thm. 4, which defines a team-optimal solution. In the next subsection, we outline a procedure for solving the hierarchical MPC problem given by (30) and (19) using the results from Secs. IV and V.

B. Computation of the Hierarchical MPC Solution

We propose a two-step procedure to solve the hierarchical MPC problem given by (30) and (19). The method is depicted in Fig. 2, and Algs. 2 and 1 provide an algorithm for the solution computation.

We first describe the overall method, as shown in Fig. 2 and Alg. 1. At each time step, given the current state $\xi(t)$, we first solve the robust BiMPC problem (38) to obtain a team-optimal solution $(u^*, \hat{w}^*)$ (line 3, Alg. 1). The team-optimal solution is provided to an *incentive solver* that iteratively computes an optimal incentive λ^* corresponding to $\hat{w}^*$ using Thm. 3 (line 4, Alg. 1). We note that, from Thm. 4, the LoMPC solution $\boldsymbol{w}^*$ obtained using the incentive λ^* is feasible for the incentive BiMPC (30). The optimal solution $(u^*, \boldsymbol{w}^*)$ is then used as an input to the dynamical system (8) to obtain the next state $\xi(t+1)$ (see Fig. 1).

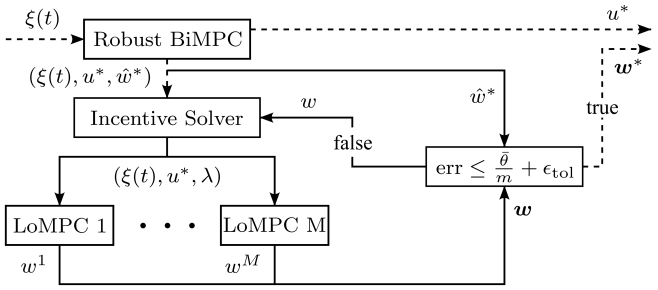


Fig. 2: A framework for solving the hierarchical MPC problem given by (30) and (19) (this figure corresponds to the controller subsystem in Fig. 1). At each time step, given the current state $\xi(t)$, we first solve the robust BiMPC (38) to get the team-optimal solution $(u^*, \hat{w}^*)$. By the bounded incentive controllability property, Lem. 3, we can find an incentive λ^* such that the average LoMPC solution w^* has bounded error compared to $\hat{w}^*$. The *incentive solver* uses the iterative method in Thm. 3 to compute λ^*. The output $(u^*, \boldsymbol{w}^*)$ is guaranteed to be feasible for the incentive BiMPC (30). In the flowchart, the solid lines indicate information flows that happen multiple times per time step, while the dashed lines indicate those that happen once.

Algorithm 1: Incentive BiMPC Solver

Input : $\xi(t)$ // Current state
Output : $(u^*, \boldsymbol{w}^*)$, λ^* // Incentive BiMPC solution
1 **Function** `solveIncentiveBiMPC` $(\xi(t))$
// Solves the incentive BiMPC problem (30) using (38)
2 $\xi_0 \leftarrow \xi(t)$
// Solve robust BiMPC (38)
3 $(u^*, \hat{w}^*) \leftarrow$ `solveRobustBiMPC` (ξ_0)
// Optimal incentive from Alg. 2
4 $\lambda^* \leftarrow$ `optimalIncentive` $(\xi_0, u^*, \hat{w}^*)$
5 **for** $i \leftarrow 1$ **to** M **do**
6 $w^{i*} \leftarrow w^{i*}(\xi_0, u^*, \lambda^*)$ // see (30b)
7 **end**
8 $\boldsymbol{w}^* \leftarrow (w^{1*}, ..., w^{M*})$
9 **return** u^*, $\boldsymbol{w}^*$, λ^*
10 **end**

Next, we describe the *incentive solver*, which is used to compute an optimal incentive λ^* given the team-optimal solution $(u^*, \hat{w}^*)$, as shown in Alg. 2. We follow the iterative method described in Thm. 3. At each iteration, we update the w-iterate using (27a) to the average LoMPC solution obtained from the current incentive λ (line 6, Alg. 2). The λ-iterate is updated using the majorization-minimization step in (27b) (line 5, Alg. 2). We iterate these update steps until the error $\|\hat{w}^* - w\|$ is less than the $\bar{\theta}/m$, which is the error bound guaranteed by Lem. 3, up to a tolerance $\epsilon_{\text{tol}} > 0$. We emphasize that the iterative method described by Alg. 2 does not require the knowledge of the LoMPC functions g^i. Instead, the optimal solutions from the LoMPCs are repeatedly queried to obtain an optimal incentive (see Assumption 2).

C. Advantages and Disadvantages of Proposed Method

Finally, we discuss the advantages and disadvantages of our method compared to an exact bilevel optimization solver, such as the one proposed in [28]. The advantages of our method are as follows: 1) Our method scales well to a large number of LoMPCs. In particular, neither the robust BiMPC computation

Algorithm 2: Iterative Method for Optimal Incentives

Input : ξ_0, $(u^*, \hat{w}^*)$ // Initial state, team-optimal solution
Param : ϵ_{tol}, $\bar{\theta}$, m
Output : λ^* // λ^* is such that (33), (34) hold
1 **Function** `optimalIncentive` $(\xi_0, u^*, \hat{w}^*)$
// Solves for an optimal incentive using Thm. 3
2 $\lambda \leftarrow \lambda_{\text{ws}}$ // Warm-start solution from previous call
3 $(w, \text{err}) \leftarrow$ `LoMPCSolution` $(\xi_0, u^*, \lambda, \hat{w}^*)$
4 **while** $err > \bar{\theta}/m + \epsilon_{tol}$ **do**
// Solve (27b) with $(w^{(k)}, \lambda^{(k)}, w^*) = (w, \lambda, \hat{w}^*)$
5 $\lambda \leftarrow$ `updateIncentive` $(w, \lambda, \hat{w}^*)$
// Update w, see (27a)
6 $(w, \text{err}) \leftarrow$ `LoMPCSolution` $(\xi_0, u^*, \lambda, \hat{w}^*)$
7 **end**
// Regularize incentive using (29), Rem. 3
8 $\lambda \leftarrow$ `regularizeIncentive` (λ, w)
9 **return** λ
10 **end**
11 **Function** `LoMPCSolution` $(\xi_0, u^*, \lambda, \hat{w}^*)$
// Average LoMPC solution, see (19) and (31)
12 $w \leftarrow (1/M) \sum_{i \in \mathcal{M}} w^{i*}(\xi_0, u^*, \lambda)$
// Convergence error, see Lem. 3
13 $\text{err} \leftarrow \|\hat{w}^* - w\|$
14 **return** (w, err)
15 **end**

time nor the optimal incentive computation time grows with the number of LoMPCs (the optimal incentive computation relies on the LoMPC solution for any incentive, which is performed in a distributed manner). Also, the error bound provided by Lem. 3 is independent of the number of LoMPCs. In contrast, the BiMPC reformulation in [28] grows in size with the number of LoMPCs. 2) Our method only requires the knowledge of the LoMPC domain, but not the LoMPC cost function. The disadvantages of our method are as follows: 1) Our method only guarantees the optimality of the incentive BiMPC problem (30) up to an error bound. 2) Our method places constraints on the LoMPC structure (see Assumption 4) and does not provide any guarantees when the incentive λ is constrained. On the other hand, the BiMPC reformulation in [28] allows for constraints on the incentive λ and computes the optimal solution of the BiMPC.

VI. Numerical Example: Dynamic Price Control for EV Charging

In this section, we consider dynamic price control for electric vehicle (EV) charging (see Sec. II-B). We mostly use the dynamical systems and costs in [32], [33], but specify when we deviate. The set of MPC problems solved by the EVs comprises the set of LoMPCs. The optimization problem solved by the ISO comprises the BiMPC problem. Together, the BiMPC and the LoMPC problems form an incentive hierarchical MPC problem, where the incentive provided by the ISO is the unit price of electricity. The Python code for the example can be found in the repository[1]. Next, we describe the LoMPC and BiMPC problems in detail.

[1] The implementation code can be found at https://github.com/AkshayThiru/incentive-design-mpc.

A. LoMPC Formulation

The i-th EV has a state variable $y_k^i \in \mathbb{R}$, which denotes its battery state of charge (SoC) as a fraction of its total capacity Θ^i at time step k. y_k^i is called the normalized SoC of EV i. The fraction of the EV battery capacity charged during time interval k is given by $w_k^i \in \mathbb{R}$. w_k^i is called the normalized charging rate at time step k. In addition to the system considered in [33], we assume that w_k^i is bounded by $w_{\max}^i$. While a more realistic model of battery charging might consider $w_{\max}^i$ as a function of the SoC, we ignore this complication. The normalized SoC of EV i evolves according to the dynamics:

$$\begin{aligned} y_{k+1}^i &= y_k^i + w_k^i, \\ 0 \le w_k^i \le w_{\max}^i \le 1, \quad 0 \le y_k^i &\le y_{\max}^i \le 1. \end{aligned} \tag{40}$$

We assume that the EV population comprises 500 small and large EVs for a total population size of $M = 1000$. Additionally, all the small EVs have the same values for $\Theta, w_{\max}$, and $y_{\max}$, given by $\Theta^s, w_{\max}^s$, and $y_{\max}^s$, respectively. Likewise, the large EVs share the values $\Theta^l, w_{\max}^l$, and $y_{\max}^l$. While this is a simplifying assumption compared to [33], we note that [33] sets different prices for each EV. On the other hand, we set identical electricity prices for large parts of the EV population. In the rest of the subsection, we discuss the small EV LoMPC formulation; a similar formulation is used for the large EVs.

Each small EV i solves a LoMPC problem with horizon N to determine the amount of electricity it draws during a time interval. Following the notation in Tab. I, let $y^i = (y_0^i, ..., y_N^i)$ and $w^i = (w_0^i, ..., w_{N-1}^i)$. The dynamics and constraints of small EV i are given by (40) with $w_{\max}^i = w_{\max}^s$ and $y_{\max}^i = y_{\max}^s$. Without loss of generality, we ignore the state constraints $\mathbb{0} \le y^i \le y_{\max}^s \mathbb{1}$ by assuming that EVs stop charging once their maximum normalized SoC $y_{\max}^s$ is attained. We also assume, for simplicity, that once an EV is fully charged, it is replaced by another EV with a random normalized SoC. We also define the total battery capacity of the EV population as follows:

$$B = (M/2)\Theta^s + (M/2)\Theta^l. \tag{41}$$

The LoMPC of each EV has a cost function with three components, which we describe next.

1) Battery Degradation Cost: The charging behavior over time affects an EV battery's health. We associate a battery degradation cost function g_{bat}^i, a function of the normalized charging rate w_k^i, to the rate of decline of the battery health [33]. The total battery degradation cost is given by

$$(\Theta^i)^2 \textstyle\sum_{k=0}^{N-1} g_{\text{bat}}^i(w_k^i), \tag{42}$$

where the $(\Theta^i)^2$ factor normalizes the battery degradation cost term with respect to the other cost terms. We assume that g_{bat}^i can be any closed convex function (possibly nonsmooth) and that g_{bat}^i is unknown to the ISO. Additionally, all small and large EVs share the battery degradation cost functions g_{bat}^s and g_{bat}^l, respectively (the exact forms can be found in the code[1]).

2) Charging Cost: Each EV i has a quadratic tracking cost at time k given by $\delta(\Theta^i)^2(y_{\max}^i - y_k^i)^2$, where $\delta > 0$ is a fixed parameter. This cost function imposes a penalty for not being at the desired SoC. The total charging cost is given by

$$\delta(\Theta^i)^2 \textstyle\sum_{k=1}^{N} (y_{\max}^i - y_k^i)^2. \tag{43}$$

3) Electricity Cost: While [33] considers an auction mechanism, we assume that the ISO sets identical electricity prices for parts of the EV population. Since electricity prices are provided by the ISO as incentives to the EVs, the electricity cost is the incentive cost term, as defined in (19). The total electricity cost for EV i is given by

$$\Theta^i\big(\lambda_{l_1}^\top w^i + \lambda_{l_2}^\top (w_{\max}^i \mathbb{1} - w^i) + (w^i)^\top \operatorname{diag}(\lambda_q) w^i\big), \tag{44}$$

where $\lambda_{l_1}, \lambda_{l_2}, \lambda_q \in \mathbb{R}_+^N$ and $\lambda = (\lambda_{l_1}, \lambda_{l_2}, \lambda_q) \in \mathbb{R}_+^{3N}$ is the incentive. The first term in (44) penalizes higher electricity consumption, while the third term adds a nonlinear electricity cost. The second term reduces prices for higher consumption. This can be interpreted as the ISO providing compensation to the EVs for charging at a higher rate at the expense of EV battery degradation. The electricity cost for small EV i can be expressed as $\langle \lambda, \phi^s(w^i) \rangle$, where

$$\phi^s(w^i) = \Theta^s\big(w^i, w_{\max}^s \mathbb{1} - w^i, w^i \odot w^i\big), \tag{45}$$

where $\odot$ represents the Hadamard product. From (45) and Rem. 1, ϕ^s is $\mathcal{K}$-convex with $\mathcal{K} = \mathbb{R}_+^{3N}$, and $\lambda \in \mathcal{K}^* = \mathbb{R}_+^{3N}$. Thus, $\langle \lambda, \phi^s(w^i) \rangle$ is a linear-convex incentive (see Def. 3).

The LoMPC problem for small EV i is given by the dynamics (40) (without the state constraints) and the three cost terms discussed above. We can verify that the small EV LoMPC matches the form in (32). Thus, by Rem. 6, the small EV LoMPC satisfies Assumption 4 and can be written in the parametric form (31) with

$$\begin{aligned} \theta^i(\boldsymbol{y}_0) &= 2\delta(\Theta^s)^2\big(y_{0,c}^s - y_0^i\big)\mathbb{1}, \\ y_{0,c}^s &= \frac{1}{2}\left(\max_{i \in \text{small EVs}} \{y_0^i\} + \min_{i \in \text{small EVs}} \{y_0^i\}\right). \end{aligned} \tag{46}$$

Similarly, we can define the LoMPC problem for a large EV.

B. LoMPC Solution Error Bounds

In this subsection, we show that Assumptions 1, 3, and 4 hold for the EV charging example, and thus we can apply all the results derived in the paper. In particular, we compute the error bound provided by Lem. 3.

Since the EV LoMPC costs are convex, Assumption 1 is satisfied. Moreover, the small EV LoMPC cost functions have a strong-convexity modulus of $m^s = 2\delta(\Theta^s)^2$, while those of large EVs have a strong-convexity modulus of $m^l = 2\delta(\Theta^l)^2$.

Next, we show that Assumption 3 holds for both small EV and large EV incentives. If $\phi^s(w^1) \preceq_{\mathcal{K}} \phi^s(w^2)$, then from (45), $w^1 \le w^2$ and $w^2 \le w^1 \Rightarrow w^1 = w^2$. For any $w^* \in \mathbb{R}^N$,

$$D\phi^s(w^*)^\top = \Theta^s[I, \ -I, \ 2\operatorname{diag}(w^*)].$$

Then, for any $v \in \mathbb{R}^N$, $D\phi^s(w^*)^\top \lambda = v$, where

$$\lambda = (\max\{v, \mathbb{0}\}, -\min\{v, \mathbb{0}\}, \mathbb{0})/\Theta^s \succeq_{\mathcal{K}^*} 0.$$

Thus ϕ^s, and similarly ϕ^l, satisfies Assumption 3.

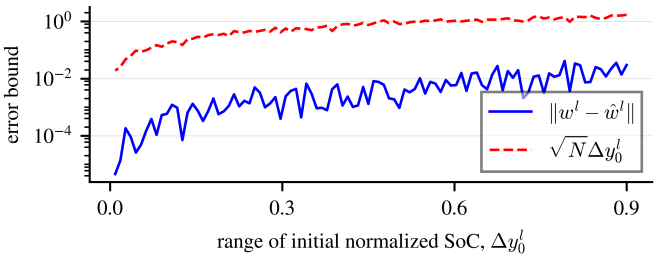


Fig. 3: Verification of the LoMPC solution error bound in (48) for large EVs with $M = 20$. For a given Δy_0^l, we randomly generate an incentive λ^l, compute the team-optimal solution $\hat{w}^l$ as the optimal solution of the EV LoMPC with $\theta^i = 0$ (see (37)), and compute w^l using (48). The plot shows that for any desired average electricity consumption $\hat{w}^l$ (that is feasible), we can set the unit price of electricity λ^l such that the actual average electricity consumption w^l has a bounded error compared to $\hat{w}^l$. The error bound is given by $\sqrt{N}\Delta y_0^l$, where Δy_0^l is the range of normalized SoC of the large EV population.

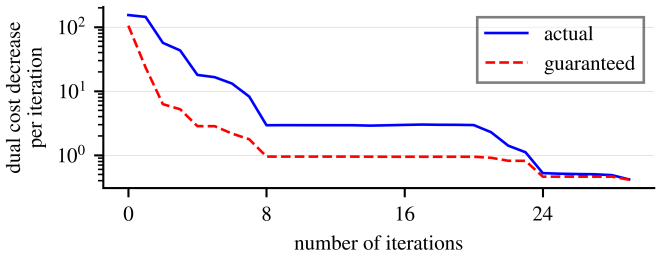


Fig. 4: Verification of the incentive solver in Alg. 2. The incentive solver uses the majorization-minimization method to compute an optimal incentive λ^* iteratively. For each iterate $\lambda^{(k)}$, the dual cost $-\tilde{g}^*$ is majorized by $-\tilde{g}^*(\cdot\,;\lambda^{(k)})$, which is then minimized (see Thm. 3). The solid line shows the actual decrease in the dual cost function $-\tilde{g}^*$ at each iteration, which is guaranteed to be greater than the decrease in $-\tilde{g}^*(\cdot\,;\lambda^{(k)})$ (the dashed line). The plot is shown for large EVs with $M = 200$.

Finally, as discussed in Sec. VI-A, Assumption 4 holds for small EVs (and likewise for large EVs), with

$$\begin{aligned} \|\theta^i(\boldsymbol{y}_0)\| &\le \bar{\theta}^s = 2\delta(\Theta^s)^2\Delta y_0^s\sqrt{N}, \\ \Delta y_0^s &= \frac{1}{2}\Big(\max_{i\in\text{small EVs}}\{y_0^i\} - \min_{i\in\text{small EVs}}\{y_0^i\}\Big). \end{aligned} \tag{47}$$

Since Assumptions 3 and 4 hold, we can use Lem. 3 to provide an error bound on the small EV LoMPC solution. Let $\hat{w}^s$ be such that $\mathbb{0} \le \hat{w}^s \le w^s_{\max}\mathbb{1}$. Then, we can find an incentive $\lambda^s \succeq_{\mathcal{K}^*} 0$ such that

$$\begin{aligned} \|\hat{w}^s - w^s\| &\le \bar{\theta}^s/m^s = \sqrt{N}\Delta y_0^s, \\ w^s &= \frac{1}{M}\sum_{i\in\text{small EVs}} w^{i*}(\xi_0, \lambda^s), \end{aligned} \tag{48}$$

where $w^{i*}(\xi_0, \lambda)$ is the optimal solution of the EV i LoMPC. In particular, we note that the error bound in (48) depends on the horizon N and the range of initial normalized SoCs Δy_0^s, but not on the population size M. We verify the dependence of the error bound on Δy_0^s in Fig. 3.

For a given $\hat{w}^s$ such that $\mathbb{0} \le \hat{w}^s \le w^s_{\max}\mathbb{1}$, an optimal incentive λ^{s*} is computed using the incentive solver given by Alg. 2. In Fig. 4, we verify the incentive solver, derived from Thm. 3, by checking the dual cost decrease condition (28) for each solver iteration.

For the robust BiMPC (38), we use a robustness horizon $N_r = 1$, i.e. we only account for the error in LoMPC solution at time step 0 (see Rem. 8). In Sec. VI-C, we find conditions under which the robust BiMPC (38) with $N_r = 1$ is always feasible. The LoMPC solution error bound at time step 0 can be found from (48) as follows:

$$|\hat{w}_0^s - w_0^s| \le \sqrt{N}\Delta y_0^s, \quad |\hat{w}_0^l - w_0^l| \le \sqrt{N}\Delta y_0^l. \tag{49}$$

C. BiMPC Formulation

The ISO decides the amount of electricity to be generated in each time interval, depending on the SoCs of the EV population and a forecast of the external electricity demand [33]. In addition to the model in [33], we assume that the ISO controls a storage battery that can be charged or depleted depending on the total electricity demand. The ISO has state variable $x_k \in \mathbb{R}$, which denotes the state of charge of its storage battery at the time step k. The amount of electricity generated in time interval k is denoted by $u_k^g \in \mathbb{R}$, and the electricity discharged from the storage battery in time interval k is denoted by $u_k^b \in \mathbb{R}$. The unit prices of electricity for small and large EVs are given by $\lambda^s, \lambda^l \in \mathcal{K}^*$, respectively. The state dynamics are as follows:

$$\begin{aligned} x_{k+1} &= x_k + u_k^b, \\ u_k^g &= (M/2)(\Theta^s w_k^s + \Theta^l w_k^l) + D_k + u_k^b, \\ 0 &\le x_k \le x_{\max}, \\ 0 \le u_k^g \le u^g_{\max}, &\quad -u^b_{\max} \le u_k^b \le u^b_{\max}, \end{aligned} \tag{50}$$

where w^s (and likewise w^l), defined in (48), is the average electricity consumption for small EVs corresponding to unit price λ^s. D_k is the forecast of external electricity demand in time interval k [33], and $x_{\max}$ is the capacity of the storage battery. $u^g_{\max}$ is the maximum electricity generation in a time interval, and $u^b_{\max}$ is the maximum charge/discharge amount for the storage battery in a time interval.

The BiMPC cost comprises electricity generation cost and aggregate charging cost [33]. The electricity generation cost penalizes high electricity generation and is given by

$$\textstyle\sum_{k=0}^{N-1} c^g (u_k^g)^{1.7}, \tag{51}$$

where $c^g > 0$ is a parameter. The aggregate charging cost is similar to the LoMPC charging cost and is given by

$$\begin{aligned} \sum_{k=1}^{N} \gamma^{k-N} \big[&(\textstyle\sum_{k'=1}^{k} w_{k'}^s - (y^s_{\max} - y^s_{0,m}))^2 \\ &+ (\textstyle\sum_{k'=1}^{k} w_{k'}^l - (y^l_{\max} - y^l_{0,m}))^2 \big], \end{aligned} \tag{52}$$

where $y^s_{0,m}$ and $y^l_{0,m}$ are mean initial normalized SoCs and $\gamma > 1$ is a parameter. For the robust BiMPC formulation, we define $\hat{w}^s$ (and likewise $\hat{w}^l$) as the desired average electricity consumption for small EVs. Then, for any $\hat{w}^s$ satisfying $\mathbb{0} \le \hat{w}^s \le w^s_{\max}\mathbb{1}$, we can find an incentive λ^s such that the error bounds in (48) and (49) hold. Using the LoMPC solution error bound (49) for robustness horizon $N_r = 1$, we can write the

constraints of the robust BiMPC (38) as (see Rem. 7) follows:

$$\begin{aligned} \hat{u}^b &= u^g - D - (M/2)(\Theta^s \hat{w}^s + \Theta^l \hat{w}^l), \\ x_{k+1} &= x_k + \hat{u}^b_k, \quad k \in \langle N \rangle, \\ \mathbb{0} &\leq u^g \leq u^g_{\max} \mathbb{1}, \\ -u^b_{\max} \mathbb{1} + \Delta e_1 &\leq \hat{u}^b \leq u^b_{\max} \mathbb{1} - \Delta e_1, \\ \Delta \mathbb{1} &\leq \hat{x}_{1:N} \leq x_{\max} - \Delta \mathbb{1}, \end{aligned} \tag{53}$$

where $e_1 = (1, 0, ..., 0)$ is a unit vector, and

$$\Delta = (M/2)\sqrt{N}(\Theta^s \Delta y^s_0 + \Theta^l \Delta y^l_0). \tag{54}$$

Note that the constraint bounds in (53), as compared to (50), are reduced by an amount proportional to Δ. Moreover, if $u^g_{\max}$ is large enough, the set of constraints (53) is feasible if, $\Delta \leq x_0 \leq x_{\max} - \Delta$ and

$$\Delta \leq u^b_{\max}, \quad 2\Delta \leq x_{\max}. \tag{55}$$

We can find a feasible vector by setting u^g such that $\hat{u}^b = \mathbb{0}$.

The Robust BiMPC problem (38) is as follows:

$$\begin{aligned} (u^{g*}, \hat{w}^{s*}, \hat{w}^{l*})(\xi_0) = \underset{u^g, \hat{w}^s, \hat{w}^l}{\operatorname{argmin}} \; & f(u, \hat{w}^s, \hat{w}^l; \xi_0), \\ \text{s.t} \quad & (u^g, \hat{w}^s, \hat{w}^l) \text{ satisfies (53)}, \end{aligned} \tag{56}$$

where the cost function f is the sum of (51) and (52) with $(w^s, w^l) = (\hat{w}^s, \hat{w}^l)$. In the hierarchical MPC formulation (see Sec. II-C), we don't impose convexity assumptions on the BiMPC cost f. Thus, more complex costs can be incorporated into the BiMPC formulation without affecting the solution.

Lastly, we describe the quantities known to the ISO. The ISO knows the values of N, Θ^s, Θ^l, $w^s_{\max}$, and $w^l_{\max}$. Notably, the battery degradation costs are private to the EVs. Additionally, at each time step, the ISO gets the current normalized SoCs of all EVs and thus knows $\xi_0 = (x_0, \boldsymbol{y}_0)$. These are the same as the assumptions in [33]. We assume that EVs are truthful when reporting their SoCs.

Remark 10. *(Differential pricing based on initial SoC) For smaller values of* Δ*, the feasible set of* (56) *is larger. To reduce the value of* Δ*, we can reduce* Δy^s_0 *and* Δy^l_0 *by differentially pricing EVs depending on their initial normalized SoC* y^i_0*. As an example, the unit price of electricity* λ^s *for a small EV* i *with initial SoC* $y^i_0 \in [0.3, 0.4)$ *might be different from that of a small EV* j *with* $y^j_0 \in [0.4, 0.5)$*. Differential pricing allows for tighter control on* $\Delta y^s_0, \Delta y^l_0$ *for EVs in each interval, and thus the error bound* Δ*. In the extreme case, we set different prices for each EV, allowing us to get an error bound of* 0*. In general, we can partition the set of small and large EVs into* P *sets depending on* y^i_0 *and set electricity prices accordingly (see Rem. 9). For clarity of presentation, we don't consider differential pricing in the BiMPC formulation* (56)*, but we do so in the implementation*[1].

D. Simulation Setup

The external electricity demand vector D, shown in Fig. 6, is obtained for a day in August 2024 in the Midcontinent ISO region of North America and scaled down by a factor of 4000. The external demand is at its peak during the afternoon and at its lowest during the early morning. The simulations are shown for a duration of 48 hrs, with MPC horizon length $N = 12$ hrs and an EV population size of 1000. All the results are normalized with respect to the total EV battery capacity B, as defined in (41).

All the convex optimization problems (the robust BiMPC in (56), the incentive solver λ update step in line 5 Alg. 2, and the EV LoMPCs) are formulated using CVXPY [41] in parametric form and solved using the CLARABEL solver [42]. The simulation parameters and the Python code for all the results can be found in the repository[1].

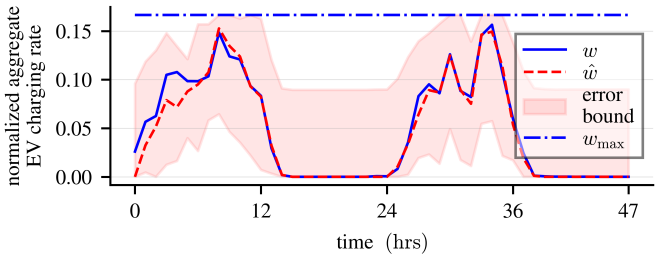


Fig. 5: The total electricity consumption by the EVs over 48 hrs, normalized by B (see (41)). The solid line shows the actual electricity consumption w, while the dotted line shows the predicted electricity consumption $\hat{w}$ (the team-optimal solution from the BiMPC). The colored region shows the error bound; the actual consumption w is guaranteed to be within the error bound range around the predicted consumption $\hat{w}$. The actual consumption reduces during peak external demand and rises when external demand is low.

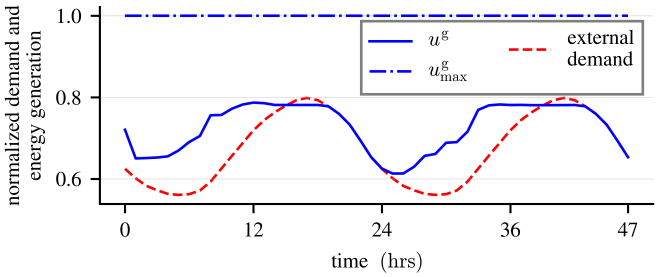


Fig. 6: Electricity generation by the ISO over 48 hrs, normalized by B (see (41)). The solid line shows the amount of electricity generated u^g, while the dotted line shows the external demand. The total electricity supply from the ISO always fulfills the total demand. The unit price of electricity is set by the ISO so that the total EV electricity consumption fills the overnight demand valley [32], [33].

E. Results

The solution trajectories obtained from the simulation are shown in Figs. 5, 6, and 7. Fig. 6 shows that the ISO sets the unit price of electricity to ensure that electricity generation is evenly spread throughout the day. Fig. 5 shows the total electricity consumption by the EV population. We can observe that EV electricity consumption is close to zero during the afternoon and high during the early morning. Fig. 7 shows the state of charge of the storage battery. The ISO charges the storage battery during low external demand and depletes it during peak external demand. Moreover, the ISO always maintains a nonzero storage battery SoC to ensure that unexpected EV electricity demands can be satisfied.

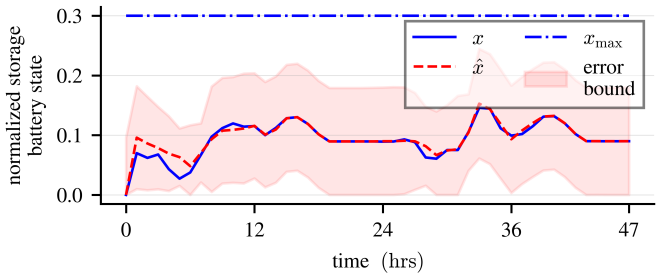


Fig. 7: The SoC of the storage battery over 48 hrs, normalized by B (see (41)). The solid line shows the actual SoC, while the dotted line shows the predicted SoC (from the BiMPC). The colored region shows the error bound; the actual SoC is guaranteed to lie within the error bound range of the predicted SoC. We note that the plot only shows the usable SoC range of the storage battery; most batteries have a minimum SoC, which is used as a baseline for the plot. The ISO stores the excess electricity generated in a storage battery. The storage battery is charged during low external demand and depleted during peak external demand to reduce electricity generation costs. Since the total EV electricity consumption is uncertain, the ISO always maintains a nonzero storage battery SoC.

During the simulation, a total of 3793 small EVs and 1715 large EVs are fully charged, which is 38.7% and 29.2% of the maximum possible value (at the highest charging rate), respectively. On average, the incentive solver requires 14.7 iterations and 22.7 iterations for convergence for small and large EVs, respectively.

Ideally, the total electricity consumption by the EVs is such that there are no spikes in electricity generation. This type of electricity consumption is called valley-filling [32], [33]. From Figs. 5 and 6, we can see that spikes in electricity generation are avoided by shifting EV consumption loads towards periods of low external demand. In our formulation, we allow EVs to set their optimal charging rates while also ensuring that the goals of the ISO are also met.

VII. Conclusions

In this paper, we presented a control framework for incentive hierarchical MPCs with multiple LoMPCs. We defined team-optimal solutions and incentive controllability for the incentive hierarchical MPC. We defined the dual of the incentive LoMPC cost function and, using tools from convex analysis, established a connection between the minima of the dual function and the set of optimal incentives. We showed that, under some assumptions, the hierarchical MPC problem with a linear-convex incentive structure is incentive controllable, and the optimal incentives can be computed iteratively. We also proved that, for the case of multiple LoMPCs, the team-optimal solution can be attained up to an error bound and provided an algorithm for computing optimal incentives. Notably, the proposed algorithm scales well with the number of LoMPCs and does not require knowing the LoMPC cost function. Finally, we considered dynamic electricity price control for large-scale EV charging. We used our algorithm to show that an ISO can set the electricity prices to approximately minimize a social cost while also guaranteeing constraint satisfaction. Our BiMPC formulation resulted in an electricity consumption pattern that fills the overnight demand valley.

Appendix I
Proofs

A. Proof of Lem. 1

Since $\langle \lambda, \phi(w) \rangle$ is linear in λ for each w and $\bar{g}^*$ is the pointwise minimum of a family of affine functions in λ, $-\bar{g}^*$ is convex [34, Sec. 3.2.3]. Since, for each $\lambda \succeq_{\mathcal{K}^*} 0$, $g + \langle \lambda, \phi(\cdot) \rangle$ is m-strongly convex (by Assumption 1) with a closed convex domain, (20) has a unique minimum and $-\bar{g}^*(\lambda) < \infty$ for each $\lambda \in \mathcal{K}^*$. So, $\operatorname{dom} \bar{g}^* = \mathcal{K}^*$.

By Danskin's Theorem [43, Prop. B.25], $-\bar{g}^*$ is subdifferentiable on $\operatorname{int}(\mathcal{K}^*)$ with subdifferential given by

$$\partial(-\bar{g}^*)(\lambda) = \left\{ -\phi(\bar{w}) : \bar{w} \in \operatorname{argmin}_w \{ g(w) + \langle \lambda, \phi(w) \rangle \} \right\}.$$

Using the fact that (20) has a unique minimum for all $\lambda \in \mathcal{K}^*$, we get that $\bar{g}^*$ is differentiable on $\operatorname{int}(\mathcal{K}^*)$ and that its gradient is given by (21). If $\lambda \in \operatorname{bd}(\mathcal{K}^*)$ (the boundary of $\mathcal{K}^*$), $-\phi(w^*(\lambda))$ is a subgradient of $-\bar{g}^*$. For ease of notation, we assign $\nabla \bar{g}^*(\lambda) = \phi(w^*(\lambda))$ when $\lambda \in \operatorname{bd}(\mathcal{K}^*)$.

Next, we prove the continuity of $w^*(\cdot)$. Let $\{\lambda^{(k)}\} \to \bar{\lambda}$, $\lambda^{(k)} \in \mathcal{K}^*$, $\|\lambda^{(k)}\| \le M$, and $w^{(k)} = w^*(\lambda^{(k)})$. First, we show that $\{w^{(k)}\}$ is bounded. Choose any $w' \in \operatorname{dom} g$ and $v \in \partial g(w')$. Then, by m-strong convexity of g and $\mathcal{K}$-convexity of ϕ (see Sec. II-A.3),

$$\begin{aligned} g(w) + \langle \lambda^{(k)}, \phi(w) \rangle &\ge g(w') + \langle v, w - w' \rangle + \frac{m}{2}\|w - w'\|^2 \\ &\quad + \langle \lambda^{(k)}, \phi(w') + D\phi(w')[w - w'] \rangle, \\ &\ge g(w') - M\|\phi(w')\| + \frac{m}{2}\|w - w'\|^2 \\ &\quad - (\|v\| + M\|D\phi(w')\|)\|w - w'\|. \end{aligned}$$

Since $g(w') + \langle \lambda^{(k)}, \phi(w') \rangle \le g(w') + M\|\phi(w')\|$, we have

$$\|w^{(k)} - w'\| \le \frac{\|v\| + M\|D\phi(w')\| + \sqrt{mM\|\phi(w')\|}}{(m/2)}.$$

Thus $\{w^{(k)}\}$ is bounded. Let $\bar{w}$ be a limit point of $\{w^{(k)}\}$ and $\{w^{(l)}\}$ be the corresponding subsequence. Then,

$$\begin{aligned} \bar{g}^*(\bar{\lambda}) &\le g(\bar{w}) + \langle \bar{\lambda}, \phi(\bar{w}) \rangle = \lim_{l \to \infty} \left(g(w^{(l)}) + \langle \lambda^{(l)}, \phi(w^{(l)}) \rangle \right), \\ &= \lim_{l \to \infty} \bar{g}^*(\lambda^{(l)}) \le \lim_{l \to \infty} \left(g(w^*(\bar{\lambda})) + \langle \lambda^{(l)}, \phi(w^*(\bar{\lambda})) \rangle \right), \\ &= g(w^*(\bar{\lambda})) + \langle \bar{\lambda}, \phi(w^*(\bar{\lambda})) \rangle = \bar{g}^*(\bar{\lambda}). \end{aligned}$$

So, any limit point $\bar{w}$ of $\{w^{(k)}\}$ minimizes $g(w) + \langle \bar{\lambda}, \phi(w) \rangle$. Therefore, there is a unique limit point of $\{w^{(k)}\}$ and $\{w^{(k)}\} \to w^*(\bar{\lambda})$, i.e. $w^*(\cdot)$ is continuous.

B. Proof of Lem. 2

From Lem. 1, $-\bar{g}^*$ is a convex function. Since $\bar{w} = w^*(\bar{\lambda})$, the following optimality condition must hold [35, Thm. 23.8]:

$$0 \in \partial g(\bar{w}) + \{ D\phi(\bar{w})^\top \bar{\lambda} \}.$$

So, $\exists v \in \partial g(\bar{w})$ such that $v = -D\phi(\bar{w})^\top \bar{\lambda}$. g is m-strongly convex under Assumption 1, and so for all $w \in \operatorname{dom} g$ we have that (see Sec. II-A.3)

$$g(w) \ge g(\bar{w}) + \langle v, w - \bar{w} \rangle + (m/2)\|w - \bar{w}\|^2.$$

Under Def. 3, for any $\lambda \in \mathcal{K}^*$, $\langle \lambda, \phi(\cdot)\rangle$ is convex and differentiable. By (1), we have that for all $w \in \operatorname{dom} g$,

$$\begin{aligned}\langle \lambda, \phi(w)\rangle &\geq \langle \lambda, \phi(\bar{w})\rangle + \langle \lambda, D\phi(\bar{w})[w-\bar{w}]\rangle,\\ &= \langle \lambda, \phi(\bar{w})\rangle + \langle D\phi(\bar{w})^\top \lambda, w-\bar{w}\rangle.\end{aligned}$$

Adding the above two inequalities and using the fact that $v = -D\phi(\bar{w})^\top \bar{\lambda}$, we get that for all $w \in \operatorname{dom} g$ and $\lambda \in \mathcal{K}^*$,

$$\begin{aligned}g(w) + \langle \lambda, \phi(w)\rangle \geq g(\bar{w}) + \langle \lambda, \phi(\bar{w})\rangle + (m/2)\|w-\bar{w}\|^2\\ + \langle D\phi(\bar{w})^\top(\lambda-\bar{\lambda}), w-\bar{w}\rangle.\end{aligned}$$

Note that for a fixed λ, the RHS of the above inequality is quadratic in w. Minimizing both the LHS and the RHS over $w \in \mathbb{R}^{\gamma N}$ and using (20), we get that for all $\lambda \in \mathcal{K}^*$,

$$\begin{aligned}\bar{g}^*(\lambda) &\geq g(\bar{w}) + \langle \lambda, \phi(\bar{w})\rangle - \|D\phi(\bar{w})^\top(\lambda-\bar{\lambda})\|^2/(2m),\\ &= \bar{g}^*(\bar{\lambda}) + \langle \lambda-\bar{\lambda}, \phi(\bar{w})\rangle - \|D\phi(\bar{w})^\top(\lambda-\bar{\lambda})\|^2/(2m),\end{aligned}$$

which is the same as (23).

Let $\Lambda \subset \mathcal{K}^*$ be any compact set. We show that the restriction $-\bar{g}^*|_\Lambda$ is Lipschitz smooth. By continuity of the function $w^*(\cdot)$ defined in (22) (Lem. 1), $w^*(\Lambda)$ is a compact set. By continuous differentiability of ϕ (Def. 3), $\|D\phi(w)\|$ is bounded on $w^*(\Lambda)$. Then, Lipschitz smoothness of $-\bar{g}^*|_\Lambda$ follows from (23) and [39, Thm. 2.1.5].

C. Proof of Thm. 3

From Lem. 1, $-\tilde{g}^*$ is a differentiable convex function on $\mathcal{K}^*$. The gradient of $\tilde{g}^*$ can be computed using (21) as follows:

$$\nabla(-\tilde{g}^*)(\lambda) = -\phi(w^*(\lambda)) + \phi(w^*).$$

By the incentive controllability assumption, Assum. 3, $\exists \lambda^* \in \mathcal{K}^*$ such that $w^*(\lambda^*) = w^*$. So, $\nabla(-\tilde{g}^*)(\lambda^*) = 0$, i.e., λ^* is a minimizer of $-\tilde{g}^*$, and $-\tilde{g}^*$ is lower bounded. Next, using (23) with $\bar{\lambda} = \lambda^{(k)}$ and $\bar{w} = w^{(k)}$, we get that

$$\begin{aligned}-\tilde{g}^*(\lambda) &\leq -\tilde{g}^*(\lambda; \lambda^{(k)}), \qquad (57)\\ &\leq -\hat{g}^*(\lambda; \lambda^{(k)}) := -\tilde{g}^*(\lambda; \lambda^{(k)}) + \epsilon^{(k)}\|\lambda - \lambda^{(k)}\|^2,\end{aligned}$$

Thus, $-\hat{g}^*(\cdot\,; \lambda^{(k)})$ is a differentiable convex function that majorizes $-\tilde{g}^*(\lambda)$, with $-\hat{g}^*(\lambda^{(k)}; \lambda^{(k)}) = -\tilde{g}^*(\lambda^{(k)})$. Using Lem. 2, and the assumption that $\{\epsilon^{(k)}\}$ and $\{\lambda^{(k)}\}$ are bounded, $-\hat{g}^*(\cdot\,; \lambda^{(k)})$ is Lipschitz smooth, with a constant independent of k. Thus, by the convergence of gradient descent for Lipschitz smooth functions, any limit point λ^* of $\{\lambda^{(k)}\}$ is a stationary point of $-\tilde{g}^*$ [43, Prop. 2.3.2]. Because $\operatorname{dom}(-\tilde{g}^*) = \mathcal{K}^*$, by the necessary and sufficient condition for optimality of convex problems [43, Prop. 2.1.2(a)],

$$\begin{aligned}&\nabla(-\tilde{g}^*)(\lambda^*) = \phi(w^*) - \phi(w^*(\lambda^*)) \in \mathcal{K},\\ &\Rightarrow \phi(w^*) \succeq_{\mathcal{K}} \phi(w^*(\lambda^*)).\end{aligned}$$

By Assumption 3, this implies that $w^* = w^*(\lambda^*)$ for all limit points λ^* of $\{\lambda^{(k)}\}$. By continuity of $w^*(\cdot)$ (Lem. 1), $\lim_{k\to\infty} w^{(k)} = \lim_{k\to\infty} w^*(\lambda^{(k)}) = w^*(\lambda^*) = w^*$.

The dual cost decrease guarantee (28) follows from (57) applied to $\lambda^{(k+1)}$ and the equality $-\tilde{g}^*(\lambda^{(k)}) = -\hat{g}^*(\lambda^{(k)}; \lambda^{(k)})$.

Let $\lambda^* \in \mathcal{K}^*$ be an optimal incentive for w^*. Then, by Lem. 1, $\nabla(-\tilde{g}^*)(\lambda^*) = 0$. Using (1), Lem. 1, and Lem. 2, we have that for any $\lambda \in \mathcal{K}^*$,

$$-\tilde{g}^*(\lambda^*) \leq -\tilde{g}^*(\lambda) \leq -\tilde{g}^*(\lambda^*) + \frac{1}{2m}\|D\phi(w^*)^\top(\lambda-\lambda^*)\|^2.$$

Thus, if $D\phi(w^*)^\top(\bar{\lambda}-\lambda^*) = 0$ for some $\bar{\lambda} \in \mathcal{K}^*$, then $-\tilde{g}^*(\lambda^*) = -\tilde{g}^*(\bar{\lambda})$, i.e., $\bar{\lambda}$ minimizes $-\tilde{g}^*(\lambda)$. Thus, $\bar{\lambda}$ is an optimal incentive.

APPENDIX II
ROBUST BIMPC FORMULATION

To write the robust BiMPC formulation (38) for the problem structure defined in (11), we can explicitly write the constraints (38b) and (38c). We assume, for now, that the state constraint sets $\mathcal{X}$ and $\mathcal{X}_\Omega$ in (11) are polytopic. In other words, we assume that the state constraints $x_k \in \mathcal{X}$ for all $k \in \langle N\rangle$ and $x_N \in \mathcal{X}_\Omega$ of the BiMPC (11) are encoded by $\{x \in \mathbb{R}^{p(N+1)} : Cx \leq d\}$. Given $(\xi_0, u, \hat{w})$, the constraint (38b) can be expressed using the LoMPC problem (9) as $\hat{w}_k \in \mathcal{W},\ k \in \langle N\rangle$ (recall from Rem. 5 that the LoMPCs cannot have any state constraints). The domain of the BiMPC cost function, $\operatorname{dom} f$, in the BiMPC constraint (38c) consists of dynamics constraints (11c) and state and input constraints (11e), (11f). The input constraint in (11e) can be directly written as $u_k \in \mathcal{U},\ k \in \langle N\rangle$.

Given $(x_0, u, \hat{w})$, the dynamics constraint (11c) can be expressed in batch form as follows [40, Sec. 8.2]:

$$x = \bar{A}^u x_0 + \bar{B}_1^u u + \bar{B}_2^u \hat{w},$$

where $\bar{A}^u \in \mathbb{R}^{p(N+1)\times p}$, $\bar{B}_1^u \in \mathbb{R}^{p(N+1)\times qN}$, and $\bar{B}_2^u \in \mathbb{R}^{p(N+1)\times \gamma N}$ are the batch matrices constructed from A^u, B_1^u, and B_2^u. The robustness constraint (38c) requires that $Cx \leq d$ is satisfied for all x of the form

$$x = \bar{A}^u x_0 + \bar{B}_1^u u + \bar{B}_2^u \hat{w} + \bar{B}_2^u \tilde{w},$$

where $\|\tilde{w}\| \leq \bar{\theta}/m$. This is equivalent to

$$\sup_{\tilde{w}:\|\tilde{w}\|\leq\bar{\theta}/m} C_j(\bar{A}^u x_0 + \bar{B}_1^u u + \bar{B}_2^u \hat{w} + \bar{B}_2^u \tilde{w}) \leq d_j, \quad \forall j$$

where C_j, d_j are the rows of C and d (see [34, Sec. 6.4]). Thus, the robustness constraint can be written as

$$C_j(\bar{A}^u x_0 + \bar{B}_1^u u + \bar{B}_2^u \hat{w}) + (\bar{\theta}/m)\|C_j \bar{B}_2^u\|_* \leq d_j, \quad \forall j,$$

where $\|\cdot\|_*$ is the dual norm [34, App. A.1.6]. Combining all the above constraints, the constraints of the robust MPC (38) can be written as

$$\begin{aligned}&\hat{w}_k \in \mathcal{W},\ u_k \in \mathcal{U}, \quad k \in \langle N\rangle, && (58a)\\ &\hat{x} = \bar{A}^u x_0 + \bar{B}_1^u u + \bar{B}_2^u \hat{w}, && (58b)\\ &C_j\hat{x} + (\bar{\theta}/m)\|C_j\bar{B}_2^u\|_* \leq d_j, \quad \forall j. && (58c)\end{aligned}$$

If some state constraint sets $\mathcal{X}$ and $\mathcal{X}_\Omega$ in (11) are given as quadratic constraints, the S-procedure can be used to reformulate the robustness constraint (38c) [34, App. B].

REFERENCES

[1] R. Scattolini, "Architectures for distributed and hierarchical model predictive control–a review," *Journal of process control*, vol. 19, no. 5, pp. 723–731, 2009.

[2] M. Simaan and J. B. Cruz Jr, "On the stackelberg strategy in nonzero-sum games," *Journal of Optimization Theory and Applications*, vol. 11, no. 5, pp. 533–555, 1973.

[3] R. R. Negenborn and J. M. Maestre, "Distributed model predictive control: An overview and roadmap of future research opportunities," *IEEE Control Systems Magazine*, vol. 34, no. 4, pp. 87–97, 2014.

[4] F. Amigoni and N. Gatti, "A formal framework for connective stability of highly decentralized cooperative negotiations," *Autonomous Agents and Multi-Agent Systems*, vol. 15, pp. 253–279, 2007.

[5] A. N. Venkat, I. A. Hiskens, J. B. Rawlings, and S. J. Wright, "Distributed mpc strategies with application to power system automatic generation control," *IEEE transactions on control systems technology*, vol. 16, no. 6, pp. 1192–1206, 2008.

[6] G. Betti, M. Farina, and R. Scattolini, "Distributed mpc: A noncooperative approach based on robustness concepts," in *Distributed model predictive control made easy*. Springer, 2013, pp. 421–435.

[7] R. R. Negenborn, B. De Schutter, and J. Hellendoorn, "Efficient implementation of serial multi-agent model predictive control by parallelization," in *2007 IEEE International Conference on Networking, Sensing and Control*. IEEE, 2007, pp. 175–180.

[8] D. Q. Mayne, M. M. Seron, and S. V. Raković, "Robust model predictive control of constrained linear systems with bounded disturbances," *Automatica*, vol. 41, no. 2, pp. 219–224, 2005.

[9] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[10] A. N. Venkat, J. B. Rawlings, and S. J. Wright, "Stability and optimality of distributed model predictive control," in *Proceedings of the 44th IEEE Conference on Decision and Control*. IEEE, 2005, pp. 6680–6685.

[11] R. Scattolini and P. Colaneri, "Hierarchical model predictive control," in *2007 46th IEEE conference on decision and control*. IEEE, 2007, pp. 4803–4808.

[12] W. Findeisen, F. N. Bailey, M. Brdys, K. Malinowski, P. Tatjewski, and A. Wozniak, *Control and coordination in hierarchical systems*. John Wiley & Sons, 1980.

[13] P. Braun, T. Faulwasser, L. Grüne, C. M. Kellett, S. R. Weller, and K. Worthmann, "Hierarchical distributed admm for predictive control with applications in power networks," *IFAC Journal of Systems and Control*, vol. 3, pp. 10–22, 2018.

[14] X. Zhang, J. Ma, Z. Cheng, S. Huang, C. W. de Silva, and T. H. Lee, "Improved hierarchical admm for nonconvex cooperative distributed model predictive control," *arXiv preprint arXiv:2011.00463*, 2020.

[15] S. Gupta, R. S. Saini, and H. S. Ganesh, "Hierarchical mpc for a dynamic process system employing parametric global optimization strategy," *Digital Chemical Engineering*, vol. 9, p. 100120, 2023.

[16] J. P. Koeln and A. G. Alleyne, "Two-level hierarchical mission-based model predictive control," in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 2332–2337.

[17] J. Koeln, V. Raghuraman, and B. Hencey, "Vertical hierarchical mpc for constrained linear systems," *Automatica*, vol. 113, p. 108817, 2020.

[18] M. Kögel, M. Ibrahim, C. Kallies, and R. Findeisen, "Safe hierarchical model predictive control and planning for autonomous systems," *International Journal of Robust and Nonlinear Control*, 2023.

[19] V. Raghuraman and J. P. Koeln, "Hierarchical mpc for coupled subsystems using adjustable tubes," *Automatica*, vol. 143, p. 110435, 2022.

[20] B. Picasso, D. De Vito, R. Scattolini, and P. Colaneri, "An mpc approach to the design of two-layer hierarchical control systems," *Automatica*, vol. 46, no. 5, pp. 823–831, 2010.

[21] R. Johansson and A. Rantzer, *Distributed decision making and control*. Springer, 2012.

[22] M. D. Mesarovic, D. Macko, and Y. Takahara, *Theory of hierarchical, multilevel, systems*. Elsevier, 2000.

[23] E. Tatara, A. Çınar, and F. Teymour, "Control of complex distributed systems with distributed intelligent agents," *Journal of process control*, vol. 17, no. 5, pp. 415–427, 2007.

[24] R. R. Negenborn, "Multi-agent model predictive control," *Delft Center for Systems and Control, Delft University of Technology, PhD thesis*, 2007.

[25] M. Zugno, J. M. Morales, P. Pinson, and H. Madsen, "A bilevel model for electricity retailers' participation in a demand response market environment," *Energy Economics*, vol. 36, pp. 182–197, 2013.

[26] M. Z. Jamaludin and C. L. Swartz, "A bilevel programming formulation for dynamic real-time optimization," *IFAC-PapersOnLine*, vol. 48, no. 8, pp. 906–911, 2015.

[27] A. Ouattara and A. Aswani, "Duality approach to bilevel programs with a convex lower level," in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 1388–1395.

[28] Y. Mintz, J. A. Cabrera, J. R. Pedrasa, and A. Aswani, "Control synthesis for bilevel linear model predictive control," in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 2338–2343.

[29] H. Mukaidani, *Incentive Stackelberg Games for Stochastic Systems*. Cham: Springer International Publishing, 2020, pp. 41–118.

[30] Y.-C. Ho, P. B. Luh, and G. J. Olsder, "A control-theoretic view on incentives," *Automatica*, vol. 18, no. 2, pp. 167–179, 1982.

[31] M. Li, J. Cruz, and M. A. Simaan, "An approach to discrete-time incentive feedback stackelberg games," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 32, no. 4, pp. 472–481, 2002.

[32] Z. Ma, D. S. Callaway, and I. A. Hiskens, "Decentralized charging control of large populations of plug-in electric vehicles," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 1, pp. 67–78, 2013.

[33] S. Zou, Z. Ma, X. Liu, and I. Hiskens, "An efficient game for coordinating electric vehicle charging," *IEEE Transactions on Automatic Control*, vol. 62, no. 5, pp. 2374–2389, 2017.

[34] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[35] R. T. Rockafellar, *Convex analysis*. Princeton university press, 1997, vol. 11.

[36] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex analysis and minimization algorithms I: Fundamentals*. Springer science & business media, 1996, vol. 305.

[37] ——, *Convex analysis and minimization algorithms II: Advanced Theory and Bundle Methods*. Springer science & business media, 1993, vol. 305.

[38] T. Başar and G. J. Olsder, *Dynamic noncooperative game theory*. SIAM, 1998.

[39] Y. Nesterov *et al.*, *Lectures on convex optimization*. Springer, 2018, vol. 137.

[40] F. Borrelli, A. Bemporad, and M. Morari, *Predictive control for linear and hybrid systems*. Cambridge University Press, 2017.

[41] S. Diamond and S. Boyd, "CVXPY: A Python-embedded modeling language for convex optimization," *Journal of Machine Learning Research*, vol. 17, no. 83, pp. 1–5, 2016.

[42] P. J. Goulart and Y. Chen, "Clarabel: An interior-point solver for conic programs with quadratic objectives," 2024.

[43] D. P. Bertsekas, "Nonlinear programming," *Journal of the Operational Research Society*, vol. 48, no. 3, pp. 334–334, 1997.



Akshay Thirugnanam received his M.S. degree in Mechanical Engineering at the University of California, Berkeley, USA, in 2022, and B.Tech. degree in Mechanical Engineering from the Indian Institute of Technology Bombay, Mumbai, India, in 2020.

He is currently a Ph.D. candidate in Controls, Mechanical Engineering at the University of California, Berkeley. His research interests include optimization, nonlinear control, safety-critical control, and robotics.



Koushil Sreenath received the M.S. degree in applied mathematics and the Ph.D. degree in electrical engineering and systems from the University of Michigan, Ann Arbor, MI, USA, in 2011.

He is an Associate Professor of mechanical engineering with the University of California, Berkeley, CA, USA. He was an Assistant Professor with Carnegie Mellon University, Pittsburgh, PA, USA, between 2013 and 2017. His research interests include the intersection of highly dynamic robotics and applied nonlinear control.