Abstract—This article presents a novel methodology for tractably solving optimal control and offline reinforcement learning (RL) problems for high-dimensional systems. This work is motivated by the ongoing challenges of safety, computation, and optimality in high-dimensional optimal control. We address these key questions with the following approach. First, we identify a sequence-modeling surrogate methodology that takes as input the initial state and a time series of control inputs and outputs an approximation of the objective function and trajectories of constraint functions. Importantly, this approach entirely absorbs the individual state transition dynamics. The sole dependence on the initial state means we can apply dimensionality reduction to compress the model input while retaining most of its information. Uncertainty in the surrogate objective will affect the resulting optimality. Critically, however, uncertainty in the surrogate constraint functions will lead to infeasibility, i.e., unsafe actions. When considering offline RL, the most significant modeling errors will be encountered on out-of-distribution (OOD) data. Therefore, we apply Wasserstein ambiguity sets to “robustify” our surrogate modeling approach subject to worst case out-of-sample modeling errors based on the distribution of test data residuals. We demonstrate the efficacy of this combined approach through a case study of safe optimal fast charging of a high-dimensional lithium-ion battery model at low temperatures.

Index Terms—High-dimensional control, lithium-ion battery, nonlinear control, optimal control, reinforcement learning (RL), robust optimization.

I. INTRODUCTION

This article presents a novel model-based data-driven method for robust optimal control and offline reinforcement learning (RL) of high-dimensional dynamical systems.

Optimal control faces unique challenges related to guaranteeing optimality and computational efficiency [1]. These challenges are generally exacerbated when the dynamical system in question is a high-dimensional system, a classification based on the cardinality of state variables $n(x \in \mathbb{R}^n)$ being high (i.e., $n > 10^2$ or $10^3$). Learning-based methods can also struggle to guarantee feasible solutions.

In this work, we introduce a simple algorithmic framework that utilizes: 1) neural function approximation; 2) dimensionality reduction; and 3) distributionally robust optimization (DRO) to obtain computationally tractable optimal control for high-dimensional nonlinear optimal control problems. This contribution is important, considering that the majority of real-life dynamical systems (i.e., heat transfer, fluid dynamics, etc.) are inherently high-dimensional. This is partially a result of their representation with partial differential equations (PDEs), which when solved numerically are frequently represented with numerous state variables [2]. Often, the model-order reduction is applied to generate a “control-oriented” dynamical model when the true underlying system is complex and high-dimensional [3]. However, reductions can refute our ability to observe fundamental insights from our optimal control solution [4]. Reductions can also compromise the capability of maximizing the performance of the control policy.

Recent literature presents a host of methods for high-dimensional optimal control. Besides use of specialized and case-specific heuristics, these generally include: 1) control vector parameterization (CVP); 2) RL and approximate dynamic programming (ADP); 3) pseudospectral optimal control (POC); and 4) variational calculus and Pontryagin methods (PM) [5], [6].

CVP is a powerful tool due to its simplicity (see [7]). In CVP, the control input is represented and manipulated in reduced form. For instance, the control input can be defined using a zero-order hold over long timesteps, or as a polynomial whose coefficients we optimize. The advantage of CVP is it reduces the number of decision variables in the optimization program. For instance, CVP has been used to reduce the complexity of highly non-convex but relatively small-scale problems [8]. Nonetheless, for high-dimensional control CVP has been shown to yield useful results [7], [9]. CVP simplifies the problem, which compromises optimality. Furthermore, CVP only addresses computational cost from the cardinality of the control input. Other sources of computational expense (i.e., simulation, numerical optimization) can still prohibit tractable solutions to the control problem.

ADP leverages black-box function approximations to enable policy learning beyond the spatial/memory limitations of tabular dynamic programming methods [10], [11]. The three biggest shortcomings of ADP relate to safety, optimality, and computation. In this context, safety refers to the ability of a learned control policy to satisfy relevant constraints. ADP and other model-free RL methods often require constraints to be encoded as auxiliary penalties to the objective/reward function [12]. Weighting these penalties requires
tuning the objective function carefully. More importantly, however, model-free and model-based RL algorithms must learn behavior through exploration. For constrained problems, this can implicitly require violating constraints throughout online learning [13]. Moreover, RL can lose guarantees of converging to an optimal policy when the problem is complex (i.e., not linear-quadratic). Furthermore, for high-dimensional nonlinear problems, ADP and model-free RL methods can require a large number of iterations to converge to a high-performing control policy [11]. At a high level, many of these challenges are just as relevant for model-based RL methods. These challenges are exacerbated when learning policies from fixed, offline datasets. Recent research in offline RL literature has provided modified algorithms that address these challenging questions while also proving to be amenable to high-dimensional control [14], [15]. In particular, offline RL methods address distributional shifts between the training data and data encountered from novel experience. For high-dimensional systems, these shifts become more likely, and can hamper optimality and feasibility.

Surrogate optimization models typically map decision variables to an approximation of the true objective function. Historically, surrogate optimization has been popular in aerospace applications, where complex high-dimensional physics-based models form the basis for design and analysis [16], [17]. The surrogate functions are fit using samples from the original objective, which is typically expensive to evaluate. The most popular approach is efficient global optimization (EGO). EGO is an adaptive sampling regime that is guaranteed to yield a surrogate optimization model with bounded modeling errors under certain conditions [18]. EGO can work for simple control problems [19], however for high-dimensional problems the parameterization of the surrogate model and the required sampling depth can become intractable. Surrogate models have also been used to approximate state-transition dynamics for control [20], [21]. This application underpins modern research activity on model-based RL [22], [23]. For high-dimensional systems, such models are ostensibly impractical again due to the expansive parameterizations which would be required to represent state-transition dynamics. The use of embeddings, latent spaces, and dimensionality reduction can ease computational demands, but add additional approximations that have yet to be addressed in a certified way [24].

Table I shows a summary of these algorithms. Existing methods possess unique strengths in solving high-dimensional optimal control problems, but there is room for further development. The objective of this article is to present a general, data-driven algorithmic framework applicable to high-dimensional systems which address the critical, unanswered question of safety and feasibility. First, we define neural network surrogates which map a reduced state representation and a finite time series of control inputs to an approximation of the objective function. Instead of constraint penalties, we develop auxiliary surrogate models which predict the time series of the constraint functions using the same reduced input data. Our method is then, by definition, a model-based RL approach. For optimal control problems with a short time horizon, we obtain approximate solutions by optimizing around the models a single time. However, for optimal control problems on longer time horizons, we apply these surrogates within a receding horizon control framework. Via a sequence-modeling method, we absorb the dynamics of the state transitions into the prediction of the surrogate models, eliminating modeling drift.

By leveraging surrogate models, we introduce modeling errors. While objective uncertainty may affect optimality, uncertainty in the constraint functions can mean the difference between safe control and critically unsafe behavior. Therefore, this work accommodates uncertainty in the constraints via distributionally robust chance constraints (DRCC). These chance constraints encode distributions of modeling errors computed from testing data. We apply Wasserstein ambiguity sets to strengthen robustness by optimizing with respect to worst case modeling errors sourced from a family of distributions within some Wasserstein distance of the empirical distribution. The Wasserstein measure is distinguished from other probabilistic distances (i.e., moment-based methods of \( \phi \)-divergence [25]) in that it is symmetric between two distributions, makes no assumptions on the shape of the distributions, and importantly provides an “out-of-sample” safety guarantee [26]. When used for DRCCs, we can probabilistically guarantee adherence to constraints even when our surrogate models experience distributional shifts relative to the training data.

To evaluate the efficacy of the algorithm, we solve the safe-fast charging problem for a high-dimensional lithium-ion battery model at low temperatures. Lithium-ion battery fast charging is currently an active research area in the energy systems and control literature. Significant challenges can arise in this problem from using reduced-order models [27]. If we leverage full-order electrochemical battery models, then we benefit from more granular electrochemical information to safe operate the cell farther toward the boundary of its safe operating conditions [28]. This increases the performance of the resulting charge/discharge cycle but requires that we strictly adhere to safety constraints. Violation of some electrochemical constraints leads to rapid aging and potential catastrophic cell failure. Consequently, the fast charging problem presents a relevant safety-critical challenge to our proposed algorithm. Historically, fast charging has been explored with reduced order models due to the nonlinearity and computational complexity of simulating the full-order dynamics [29], [30], [31]. By demonstrating that our surrogate optimal control algorithm can yield fast and feasible charge cycles based on the full-order electrochemical model in real-time, we validate its use for high-dimensional nonlinear optimal control problems.

The results in this article comprise a significant extension of our previous work in [32]. These extensions include:

| Algorithm | Challenges |
|-----------|------------|
| CVP       | optimality, computation, requires model knowledge |
| RL*       | safety, optimality, computation |
| POC       | requires model knowledge, proprietary software |
| PM        | numerical instability, computation, requires model knowledge |
1) a comprehensive novel case study using a full-order electrochemical battery model, including a computational comparison to control using a reduced order model and 2) the use of Wasserstein ambiguity sets instead of more limited φ-divergence.

II. PROBLEM FORMULATION

A. Optimal Control Problem Formulation

This article considers the following optimal control problem statement, cast in discrete time:

\[
\begin{align*}
\text{min} \quad & \sum_{k=0}^{N} J(x_k, u_k) \\
\text{s.t.} \quad & x_{k+1} = f(x_k, u_k) \\
& g(x_k, u_k) \leq 0 \\
& h(x_k, u_k) = 0 \\
& x_0 = x(0)
\end{align*}
\]

where \( k \) is the current time and \( N \) is the final time; \( x_k \in \mathbb{R}^n \) is the state vector at time \( k \); \( u_k \in \mathbb{R}^p \) is the control input vector; \( J(x_k, u_k) : \mathbb{R}^n \times \mathbb{R}^p \rightarrow \mathbb{R} \) is the stage cost function at time \( k \); \( f(x_k, u_k) : \mathbb{R}^n \times \mathbb{R}^p \rightarrow \mathbb{R}^n \) represents the system dynamics; \( g(x_k, u_k) : \mathbb{R}^n \times \mathbb{R}^p \rightarrow \mathbb{R}^m \) represents inequality constraints; and \( h(x_k, u_k) : \mathbb{R}^n \times \mathbb{R}^p \rightarrow \mathbb{R}^f \) represents equality constraints. In this article, we are particularly interested in problems where the cardinality of \( x \) is high, i.e., \( n > 10^2, 10^3 \), or more.

Our objective is to simplify the computation required to solve (1a)–(1e) when the model is high-dimensional. Fig. 1 shows a block diagram of our method. In Sections II-B and II-C, we discuss the components represented in this diagram.

B. Offline Dataset

Our method leverages a fixed, offline dataset composed of state trajectories matched with control input sequences. Typically, training data for surrogate optimization models are generated via a host of methods. For instance, one popular method in the literature is Latin hypercube sampling (LHS) [17]. In another method, EGO, sampling from the original objective function is organized and adaptive to the real-time evolution of modeling errors [18]. In this article, we train our surrogate models using data obtained from random, offline, parallelizable simulations of the original high-dimensional dynamical model. However, any dataset could be used to learn these surrogate models. For example, such data could come from physical experiments, an existing suboptimal controller, etc. In considering how such a dataset can be generated, the distributional shift problem becomes highly relevant. We want to minimize the degree to which real-time control data will deviate from the distribution of training data. How this question is answered is highly dependent on the specific application. Importantly, our framework is data-driven and does not require explicit model knowledge. This is differentiated from many existing methods (including CVP, pseudospectral optimal control).

C. Model Formulation and Training

Within the context of optimal control, surrogate models have been applied to represent state transition dynamics directly [20], [21]. Direct approximation of state transition dynamics is not ideal for high-dimensional dynamical systems, where the large cardinality of state variables would require function approximators with intractable parameterizations. This article proposes using a modified finite-time surrogate modeling approach which takes the following form:

\[
\begin{align*}
\text{min} \quad & \mathcal{J}(x_0, U) \\
\text{s.t.} \quad & \mathcal{G}_i(x_0, U) \leq 0 \quad \forall i = 1, \ldots, m.
\end{align*}
\]

The surrogate model \( \mathcal{J} \) absorbs the state transition dynamics by mapping the initial state \( x_0 \) and time series of control inputs \( U = [u(0), \ldots, u(N)] \) directly to an approximation of the objective function given in (1a). In set notation \( \mathcal{J}(\cdot, \cdot) : \mathbb{R}^n \times \mathbb{R}^{p \times (N+1)} \rightarrow \mathbb{R} \). Likewise, the surrogate constraint functions \( \mathcal{G}_i : \mathbb{R}^n \times \mathbb{R}^{p \times (N+1)} \rightarrow \mathbb{R}^{(N+1)} \) take the same inputs and predict as output a time series of the relevant constraint function values for each of \( i = 1, \ldots, m \) inequality constraints. Importantly, the constraint surrogates only model the most relevant information in time series format. State variables that do not pertain to constraints in the optimization
problem are disregarded by the surrogate models. Furthermore, by outputting an entire time series, we avoid the possibility of modeling drift inherent to a surrogate that predicts individual state transitions across a single time step [22].

For a model predictive control application, the optimal control problem in (2a) and (2b) becomes

\[
\begin{align*}
\min & \quad J(x_k, U_{k:k+N}) \\
\text{s.t.} & \quad G_i(x_k, U_{k:k+N}) \leq 0 \quad \forall i = 1, \ldots, m.
\end{align*}
\]  

At \( k = 0 \), the initial state becomes the current state, and the control input time series \( U_{k:k+N} = [u_k, \ldots, u(k+N)] \) starts at the current state and evolves over a horizon of \( N \) time steps into the future. Note we are re-using \( N \) here to indicate the control horizon length relative to the current time step, as opposed to the global time horizon length in (2a) and (2b). After solving this reduced optimization program, we apply the first control input to the plant, simulating one step forward and then repeating the overall process.

The most important transformation we make relates to reducing the state with dimensionality reduction techniques. This article specifically uses principal component analysis (PCA) to project the state on a reduced basis. So in fact, the optimization program becomes

\[
\begin{align*}
\min & \quad \tilde{J}(\tilde{x}_k, \tilde{U}_{k:k+N}) \\
\text{s.t.} & \quad \tilde{G}_i(\tilde{x}_k, \tilde{U}_{k:k+N}) \leq 0 \quad \forall i = 1, \ldots, m
\end{align*}
\]

where \( \tilde{x}_k \) is a reduced representation of the dynamical state. Note the control is not included with state reduction, because its approximation could corrupt the input signal and negatively impact performance.

1) Note: Facilitating Optimization: This article’s approach requires that we optimize around the neural network architecture. This architecture shares similar nonconvexity with the original expensive-to-evaluate objective function [18]. Past work has explored the use of convex neural architectures to facilitate this format of optimization [20]. However, input-convex neural networks can compromise the universal function approximator properties of general neural networks [33].

The use of neural function approximation allows us to exploit analytic expressions for the function input–output gradient, as done in [20]. For instance, for a single hidden layer neural network \( f(x) = \sigma_{\text{out}}(W_2\sigma_{\text{hidden}}(W_1x + b_1) + b_2) \) where \( \sigma_{\text{out}}(x) = x \), the Jacobian is given by

\[
\text{Jac}(f(x))_{ij} = W_1(:, i)^T W_2(f(.))\sigma'_{\text{hidden}}(W_1x + b_1).
\]  

Were we to solve the original optimal control problem with no surrogates, any gradients would be computed numerically with finite differences, which is highly inefficient. Numerical gradient calculations scale on the order of \( O[n^3] \) for a function \( f : \mathbb{R}^n \to \mathbb{R} \), which would add significant computational complexity [34]. By supplying the numerical optimization solver with analytic expressions for the input–output gradients of relevant surrogate models, we avoid expensive numerical gradient calculations. Consequently, analytic gradients provide a fruitful opportunity to reduce computational complexity.

In this article, we evaluate and compare two optimization schemes. First, we use numerical optimization with specified analytical gradients. We compare this approach to a sample-based random search. Past work has shown for some applications that random search can provide high-performing results relative to more conventional optimization approaches [35]. In this article, we specifically apply a \((1+λ)\) evolutionary strategy algorithm to solve the receding horizon control problem. Section IV of this article provides more details of this comparison. Overall, however, random search outperformed the gradient-based approach.

III. ROBUSTNESS TO MODELING ERRORS

Surrogate models are inherently imperfect. Uncertainties are expected in approximations of both the objective and constraint functions and, if unaccounted for, these uncertainties can affect the optimality and feasibility of the final solution [34].

This article addresses uncertainties in the constraint functions with a DRO framework. We robustify our surrogate constraint models by optimizing with respect to worst case realizations of modeling error characterized by the test data distribution of residuals. We obtain the worst case realization through the construction of a Wasserstein ambiguity set, which lends a probabilistic out-of-sample safety guarantee. This section details relevant mathematical preliminaries for this approach.

A. Stochastic Optimization With Chance Constraints

A chance-constrained program includes probabilistic constraint statements, with random variables \( \mathbf{R} \) with support \( \Xi \). Consider \( x_k \in \mathbb{R}^n \) is the system state at time step \( k \), \( u(k) \in \mathbb{R}^p \) is the control input, \( \mathbf{R} \in \mathbb{R}^m \) is the random variable in question, and \( g(x_k, u_k, \mathbf{R}) : \mathbb{R}^n \times \mathbb{R}^p \times \mathbb{R}^m \to \mathbb{R}^m \) is the vector of inequality constraints. The chance constraint is

\[
\hat{P}[g(x_k, u_k, \mathbf{R}) \leq 0] \geq 1 - \eta
\]  

where \( \eta \) is our risk metric or the probability of violating the constraint. The chance constraints discussed above depend on known distributions corresponding to each random variable. For many applications, we approximate these distributions using data to create an empirical cumulative distribution function (CDF). In many data-driven applications, the true probability distribution \( \mathbb{P}^\ast \) for the random variable \( \mathbf{R} \) is unknown. Thus, our empirical distribution \( \hat{\mathbb{P}} \) provides an approximation of \( \mathbb{P}^\ast \) from data. Borel’s law of large numbers indicates that as the number of samples \( \ell \to \infty \), \( \hat{\mathbb{P}} \to \mathbb{P}^\ast \). This discrepancy characterizes distributional uncertainty in the random variable. This can affect our solution if \( \hat{\mathbb{P}} \) is inaccurate [36]. The literature presents several means by which we can accommodate this uncertainty. In Section III-B, we discuss the application of the Wasserstein distance within this context.

B. Wasserstein Ambiguity Sets

An empirical distribution composed of samples will inevitably be characterized by some error or uncertainty. In a qualitative sense, this uncertainty can be represented as the distribution lying some distance from the true distribution. In statistics, there are several methods used to describe this type of distance. These include \( \phi \)-divergence and the
Wasserstein metric, the latter of which this article applies for distributionally robust control.

**Definition 1:** Given two marginal probability distributions \( P_1 \) and \( P_2 \) lying within the set of feasible probability distributions \( P(\Xi) \), the Wasserstein distance between them is defined by

\[
W(P_1, P_2) = \inf_{\Pi} \left\{ \int_{\Xi} \|R_1 - R_2\|_\alpha \Pi(dR_1, dR_2) \right\}
\]

where \( \Pi \) is a joint distribution of the random variables \( R_1 \) and \( R_2 \), and \( \alpha \) denotes any norm in \( \mathbb{R}^n \) [25].

The Wasserstein distance allows us to replace the random variable with a “worst-case” realization sourced from a family of distributions within a specified Wasserstein distance of our empirical distribution. This family of distributions forms the Wasserstein ambiguity set. For instance, let us define the ambiguity set as \( B_\epsilon \), a ball of probability distributions with radius \( \epsilon \) centered around our empirical CDF \( \hat{P} \)

\[
B_\epsilon := \{ P \in P(\Xi) \mid W(P, \hat{P}) \leq \epsilon \}
\]

where \( \epsilon \) is the Wasserstein ball radius. Now, we can formulate the robust counterpart of the chance constraint in (6)

\[
\inf_{P \in B_\epsilon} \mathbb{P}[g(x_k, u_k, R) \leq 0] \geq 1 - \eta.
\]

This equation provides the basis for the out-of-sample safety guarantee afforded by this DRO framework. Namely, we are probabilistically guaranteed to satisfy the chance constraint for any true probability distribution within the \( \epsilon \) distance of the empirical distribution.

Several expressions exist for the Wasserstein ball radius which, for a given confidence level \( \beta \), is probabilistically guaranteed to contain the true distribution. We adopt the following formulation of \( \epsilon \) from [37] where \( D \) is the diameter of the support of \( R \) composed of \( \ell \) samples

\[
\epsilon(\ell) = D \sqrt{\frac{2}{\ell} \log \left( \frac{1}{1 - \beta} \right)}
\]

assuming the underlying data is independent and identically distributed (i.i.d.). In [38], this formula is replaced with the following version:

\[
\epsilon(\ell) = C \sqrt{\frac{2}{\ell} \log \left( \frac{1}{1 - \beta} \right)}
\]

where \( C \) is obtained by solving the following scalar optimization program:

\[
C \approx 2 \inf_{a > 0} \left\{ \frac{1}{2a} \left[ 1 + \ln \left( \frac{1}{N} \sum_{k=1}^{N} e^{a(\|\tilde{g}^{(k)} - \hat{\mu}\|_2^2)} \right) \right]^{\frac{1}{2}} \right\}
\]

where the right side bounds the value of \( C \), and \( \tilde{g}^{(k)} \) is a centered and normalized sample of the random variable which comprises our empirical distribution [38]. This replacement is done to eliminate some unnecessary conservatism.

The exact constraint shown in (9) is intractable, given that solving it requires solving an infinite dimensional nonconvex problem. Most ongoing research in DRO focuses on deriving equivalent reformulations of (9) which are more readily solved and implemented.

What is particularly noteworthy about Wasserstein ambiguity sets is their inherent out-of-sample safety guarantee. That is the probabilistic safety guarantee covers, both, cases where we encounter experience and data which is not represented in the empirical distribution. This is principally due to the fact that the Wasserstein distance between two probability distributions bears no assumptions on the shape or support of each distribution. We demonstrate this feature with the following comparison to \( \phi \)-divergence-based reformulations of the constraint in (12). If we were to utilize a \( \phi \)-divergence to reformulate (6) as done in our previous work in [32]

\[
\mathcal{B}_\phi = \{ P \in P(\Xi) \mid \phi(P, \hat{P}) \leq d \}
\]

where \( d \) is a distance-like hyperparameter that must be tuned and chosen from intuition, then existing equivalent reformulations simply perturb the risk level [25]. However, perturbing the risk level provides much more limited out-of-sample guarantees because it limits the realization of the random variable to lie within support that we have already observed. This finding is partially defined by the fact that the \( \phi \)-divergence between two probability distributions with different supports is infinite. As a result, we adopt the Wasserstein distance metric for the remainder of this article.

### C. Equivalent Chance Constraint Reformulation

In this article, we adopt an equivalent reformulation of (9) from [38]. This specific reformulation requires that the constraint function \( g(x_k, u_k, R) \) is affine in \( R \). An in-depth discussion of this reformulation can be referenced in [38]. Here, we restate a brief overview of their methodology and derivation.

We begin with samples of data \( \{R^{(1)}, R^{(2)}, \ldots, R^{(i)}\} \) corresponding to random variable \( R \in \mathbb{R}^m \). This sample comprises our empirical distribution \( \hat{P} \), and the data is drawn from the true underlying distribution \( P^* \). First, we normalize the data samples to form a new random variable \( \tilde{g} \) as follows:

\[
\tilde{g}^{(i)} = \Sigma^{-\frac{1}{2}}(R^{(i)} - \mu)
\]

where \( \Sigma \) is the sample variance of the data and \( \mu \) is the sample mean. This standardization transforms the data samples such that its new mean is 0, and its new variance is \( I_{m \times m} \). Now, we define the support of this normalized distribution as

\[
\Theta = \{ \tilde{g} \in \mathbb{R}^m \mid -\sigma_{\max} I_m \preceq \tilde{g} \preceq \sigma_{\max} I_m \}
\]

Here, \( \sigma_{\max} \in \mathbb{R} \) defines the support of the normalized random variable and \( I_m \) is a column vector of ones. Now, let \( Q^* \) and \( Q^\ast \) represent the true and empirical distributions of the normalized data \( \tilde{g} \). We construct the ambiguity set \( \hat{Q} \) using the “Wasserstein ball” given by (8), allowing us to transform the chance constraint in (6) to

\[
\sup_{Q \in \hat{Q}} Q[V(\tilde{g} \notin V)] \leq \eta
\]

which says the worst case probability that normalized random variable \( \tilde{g} \) is outside set \( V \) is less than \( \eta \), where the supremum is taken over all distributions \( Q \) in ambiguity set \( \hat{Q} \). We wish to obtain the least conservative (i.e., tightest) set \( V \subseteq \mathbb{R}^m \) in

---

*Note: The above text is a transcription of the document content, formatted for readability. The LaTeX equations have been rendered in plain text format for clarity.*

---

*Acknowledgments: This work was supported by the National Science Foundation under Grants CNS-1707606 and CNS-1921503. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.*
order to define the desired Wasserstein uncertainty set \( \mathcal{A} = \{ \alpha \in \mathbb{R}^m \mid \alpha = \Sigma^{(1/2)}b + \mu, \ b \in \mathcal{V} \} \) such that
\[
g(x_k, u_k, R) \leq 0 \quad \forall \ R \in \mathcal{A}. \tag{17}
\]
We restrict the overall shape of the set \( \mathcal{V} \) to be a hypercube, which enables computational tractability
\[
\mathcal{V}(\sigma) = \{ \vartheta \in \mathbb{R}^m \mid -\sigma 1_m < \vartheta < \sigma 1_m \}. \tag{18}
\]
Now, to compute this ambiguity set without introducing unnecessary conservatism, we need to find the minimum value of the hypercube side length \( \sigma \in \mathbb{R} \). The following optimization program details this problem:
\[
\min_{0 \leq \sigma \leq \sigma_{\max}} \sigma \\
\text{s.t.:} \quad \sup_{Q \in \mathcal{Q}} \{ \vartheta \notin \mathcal{V}(\sigma) \} \leq \eta. \tag{19}
\]
Here, we select \( \sigma_{\max} \) using a priori information about the specific problem context.

The derivation in [38] provides a worst case probability formulation, summarized by the following Lemma.

**Lemma 1 (Lemma 2 of [38]):**
\[
\sup_{Q \in \mathcal{Q}} \{ \vartheta \notin \mathcal{V}(\sigma) \} = \inf_{(x) \geq 0} \left\{ \lambda \varepsilon(\ell) + \frac{1}{\ell} \sum_{j=1}^{\ell} \left( 1 - \lambda (\sigma - ||\vartheta^{(j)}||_\infty)^+ \right)^+ \right\} \geq \min \left( \eta, \frac{\sigma}{\ell} \right). \tag{20}
\]
where \((x)^+ = \max(x, 0)\).

We defer to [38] for the proof of this finding. Their result entails that (21) can be reformulated as
\[
\min_{0 \leq \lambda, R \leq \sigma \leq \sigma_{\max}} \sigma \\
\text{s.t.:} \quad h(\lambda, \sigma, \lambda) \leq \eta \tag{22}
\]
where
\[
h(\lambda, \sigma, \lambda) = \lambda \varepsilon(\ell) + \frac{1}{\ell} \sum_{j=1}^{\ell} \left( 1 - \lambda (\sigma - ||\vartheta^{(j)}||_\infty)^+ \right)^+ \tag{23}
\]

The result of this optimization program is the value of \( \sigma \), which is used to reformulate the chance constraints via convex approximation. For a convex approximation of the constraint function in (9), the hypercube \( \mathcal{V}(\sigma) \) becomes the convex hull of its vertices. If for example \( m = 1 \) (i.e., the random variable is 1-D), then \( \mathcal{V}(\sigma) = (-\sigma, \sigma) \)–an open interval. In general, this yields the ambiguity set \( \mathcal{A} = \text{conv}((-\sigma, \sigma]) \) where \( r = \Sigma^{(1/2)}1_m \sigma + \mu \) and \( \text{conv}([-\cdots]) \) represents the convex hull of points \([\cdots]\). We can leverage this to complete the convex approximation of (9) as a set of constraints of the form
\[
g(x_k, u_k) + r \leq 0 \tag{24}
\]
which enumerate through the vertices of the robust hypercube. For an \( m \)-dimensional constraint function, the exact form of the ambiguity set is \( \mathcal{V} = \text{conv}([\vartheta^{(1)}, \ldots, \vartheta^{(2^m)}]) \). The set of constraints are
\[
g(x_k, u_k) + r^{(j)} \leq 0 \quad \forall \ j = 1, \ldots, 2^m. \tag{25}
\]
Algorithm 1 details the method used to compute the offset \( \sigma \).

### D. Modeling Error as a Random Variable

Notice that we have treated the residuals between the true and surrogate constraint functions, \( R \), as stochastic. In reality, the underlying process which generates these residuals can have a deterministic structure, since they can be generated from deterministic models. That said, the training process yields an empirical set of residuals, for which an empirical probability distribution can be constructed. This stochastic modeling choice is convenient for chance-constrained optimization, even if it neglects the underlying generative structure. This does comprise a conservative approximation, as the modeling error is treated as its worst case realization at every location in the state space. However, since safety and robustness are our principal goals, this mild degree of conservatism is acceptable.

### IV. Case Study

Next, we present a case study to validate and characterize the performance of the proposed algorithmic architecture. Our case study is safe fast charging of a lithium-ion battery at low temperatures. Lithium-ion battery fast charging is a highly relevant safety-critical application that possesses a rich and diverse history of research. It also presents a prototypical high-dimensional optimal control problem, in that complex electrochemical battery models are described with hundreds or even thousands of state variables. While reduced-order equivalent circuit models address these dimensionality problems, the granular electrochemical information afforded by the full-order models allows us to confidently take the battery closer to the safe operating envelope boundary. This grants us the ability to exploit electrochemistry to improve charging performance [32].

Low temperatures complicate the fast charging problem, as they sensitize many of the complex electrochemical dynamics. Specifically, the cell side-reaction overpotential constraint, which dictates the rate of lithium plating and cell degradation, can be much more readily violated at low temperatures [39]. Thus, the optimal control problem possesses many opportunities for constraint violation, which allows us to properly validate the efficacy of the proposed DRO framework.

Our case study is structured precisely as follows, where we solve a high-dimensional fast-charging problem using the full-order Doyle-Fuller-Newman model (DFN) [40]. We also compare computation between the full order problem and one
included in past work [32] based on a moderately reduced single particle model. We ensure a comparison of our results with and without the added DRO framework, in order to validate its relative value and contributions to the safety of our algorithmic architecture.

A. Electrochemical Battery Model

High-fidelity battery modeling provides insights into performance, without requiring one to build and experimentally test the cell. The mathematical model formulated in Appendix is the DFN battery model which comes from porous electrode theory, where Li-ions intercalate/deintercalate into porous spherical particles in the negative and positive electrodes. During charging, the Li-ions in the positive electrode deintercalation, dissolve into the electrolyte, and then migrate and diffuse to the negative electrode by passing through the separator. Critically, this full-order electrochemical model reveals insights into the mechanisms within the battery cell which allow us to take the battery farther toward the limit of its safe operating conditions. By exploiting electrochemistry, we can calculate and apply faster, higher-performing charging cycles.

While we relegate the model equations to Appendix, we include some basic, useful information in this section in Table II for reference in discussing this article’s problem formulation and results.

B. Optimal Control Problem Statement

For the DFN fast charging case study, we adopt the following optimal control problem statement within the framework of receding horizon control:

\[
\min_{\mathbf{x}_t, \mathbf{u}} \sum_{k=0}^{T} (\mathbf{SOC}_k - \mathbf{SOC}_{\text{targ}})^2
\]

\[
\text{s.t.}
\begin{align*}
\mathbf{SOC} & \geq \mathbf{SOC}_{\text{targ}} \\
T & \leq T_{\text{max}} \\
0 & \leq I \leq 2.5
\end{align*}
\]

The key constraints are that the side reaction overpotential stays positive, and the temperature does not exceed a maximum allowed threshold. The overpotential constraint is the most critical barrier to preventing rapid aging and potentially catastrophic failure of the cell. If overpotential becomes negative, lithium metal begins to plate on the anode. This phenomenon reduces the capacity of the cell and leads directly to cell failure. The temperature constraints provide indirect ways to avoid rapid aging, as the cell dynamics become more sensitive at temperature extremes.

We adapt this formulation using the distributionally robust surrogate modeling approach to yield

\[
\min_{\mathbf{x}_t} J(\mathbf{x}_t, \mathbf{u}_t)
\]

\[
\text{s.t.}
\begin{align*}
\mathcal{G}_{\eta}(\mathbf{x}_t, \mathbf{u}_t) & \geq r_{\eta} \\
\mathcal{G}_T(\mathbf{x}_t, \mathbf{u}_t) & \leq T_{\text{max}} - r_T
\end{align*}
\]

Table II details several important hyperparameters for this case study. We consider a nickel-manganese-cobalt (NMC) battery cell. The initial electrochemical states correspond to equilibrium with a voltage of \( V = 3.25 \) V. The cell is at the same uniform temperature as the ambient temperature of \( T_{\text{amb}} = 281 \) K. We simulate 150 random charging trajectories to generate the requisite training data to fit the surrogate models. Each trajectory was either terminated if: 1) the target state of charge (SOC) of 0.7 was reached or 2) the episode end time of 55 min was reached. The maximum allowed C-rate for these simulations is 2.5C, where the C-rate for a lithium-ion battery is the parameter describing how much input current would be needed to charge the battery from empty to full in exactly 1 h. A typical target SOC for electric vehicle applications is 0.8 or higher. Software implementations of the DFN model lose some numerical stability when applying high C-rates at higher SOCs. To ensure we can continue utilizing a maximum C-rate of 2.5, we instead choose to set a slightly lower target SOC of 0.7 in our case study. Our algorithm can, however, be adapted to charge a battery cell to a higher SOC.

Using PCA on the state trajectories, we decide to project the state vector \( \mathbf{x} \in \mathbb{R}^{2687} \rightarrow \mathbb{R}^{40} \). This decision is motivated by the explained variance of the data, plotted in Fig. 2. Fig. 2 shows that the first 40 principal components of the state vector data explain 99.74% of the variance in the dataset.

The surrogate models are feed-forward neural networks each with two hidden layers, each with ten neurons and sigmoid activation functions. The distribution of test data residuals for side reaction overpotential constraint function \( \mathcal{G}_{\eta} \) are shown in Table II.

| State Variable | Description | Units |
|----------------|-------------|-------|
| SOC | State of Charge | | |
| \( \eta \) | Side-Reaction Overpotential | | |
| \( T \) | Cell Temperature | | |
| \( I \) | Input Current | | |
| \( r \) | C. Rate | | |
| \( r_{\eta} \) | | | |
| \( r_T \) | | | |

Fig. 2. Individual and cumulative explained variance from PCA of the electrochemical model state trajectories.

\[
0 \leq I \leq 2.5.
\]
The analytic gradients made the fmincon solver, which we supplied with analytical gradient expressions for each function approximator, several orders of magnitude faster compared to using finite differences for gradient calculations. However, the average computation time per time step using fmincon was 9.1007 s whereas random search only required 2.0968 s/timestep on average. We also find that the random search approach yields results of higher relative quality in terms of the overall charging time performance compared to the fmincon solver. The improved performance of random search, in terms of speed and solution equality, led us to use the random search method for our final results included in this article.

Our first benchmark is a hyper-aggressive constant current constant voltage (CCCV) charging protocol with 2.5 C-rate maximum input current and 4.2 V cutoff voltage. A CCCV protocol charges the battery at the maximum allowed current until a cutoff voltage is reached. From that point on, the battery is charged at a rate that keeps the voltage at the specified threshold. Typically, CCCV profiles correspond to thresholds given in the battery cell specifications document, which tend to limit the maximum allowed input current to around 1 C for most NMC cells. For the sake of consistency, we keep the maximum allowed current the same for each method. CCCV contextualizes the relative performance of the proposed method.

As a point of comparison, we also implemented conservative Q-learning (CQL), a popular offline RL algorithm that addresses distributional shift through penalties on out-of-distribution (OOD) actions [15]. The CQL network is a feed-forward network with two hidden layers each composed of 64 neurons, and ReLU activations. The network input is the DFN state projected via the same PCA approach as our method. We discretize the input current into 11 bins between 0 and 2.5 C-rate. The network is trained in tandem with a target network iteratively with the same offline dataset used to learn the surrogate models of our approach. The reward function is given below and is adopted with slight modification from recent work [42] successfully applying actor-critic RL methods to lithium-ion battery fast charging.

$$r = -I - 100(\eta_{s} < 0 | \eta_{s})$$. (28)

A complementary OOD CQL loss is augmented to this reward function when training the networks [15]. CQL is a model-free method, meaning its sample efficiency is not as high as our model-based approach. In [42], model-free actor–critic methods are shown to require on the order of 3e3 episodes of learning to achieve high-performing charging results. Given in this case we are dealing with more than an order of magnitude reduction in available data, the fidelity of these CQL results is actually quite impressive. CQL unfortunately does not provide certificates on safety and feasibility, which is reflected in the final charging profile as shown in Fig. 4. This highlights a comparative advantage of our model-based RL methodology, namely its out-of-sample safety guarantees.

Both CCCV and CQL are relevant benchmarks for the following application and methodological reasons. Firstly, CCCV is by far the most popular fast-charging algorithm in practice. In fact, closed-form optimal fast charging with reduced-order battery models collapses to the CCCV profile [43]. Speaking of methods, both algorithms share with our algorithm a lack of dependence on a priori model knowledge. An explicit understanding of the underlying physics is not needed to achieve good control results with any of these methods, meaning they are operating on the same playing field. Second, CQL is a state-of-the-art learning-based control method that is designed to address many of the same problems our approach addresses. Principal among such problems is the “distributional shift” challenge that is prominent in offline RL. Finally, neither CQL nor any other existing state-of-the-art offline RL method (e.g., model-based offline optimization, or MOPO [44]) provides any explicit safety guarantees when considering the context of lacking model knowledge. This reveals the relative
Fig. 4. Optimal charging results for the DFN model using an NMC cell parameterization. Here, the maximum allowed C-Rate is 2.5C and the target SOC is 0.7. Charging is marked as complete at the vertical dotted lines for each respective trajectory.

value of our novel methodology, insofar as our mechanism toward addressing distributional shift carries with it strong probabilistic safety guarantees.

Fig. 4 shows the optimal fast-charging results for versions of our algorithm with and without DRO. Overall, the CCCV protocol charges in 30.6 min, the non-robust predictive controller in 32.35 min, the full distributionally robust controller in 34.1 min, and the CQL controller in 42 min. The industry benchmark CCCV protocol yields a good performance with respect to charging time with a total time of 30.6 min. However, it significantly violates the safety constraint by up to 0.12 V, and for extended periods of the overall experiment. This would undoubtedly lead to significant degradation and potential failure of the cell. Fig. 5 shows constraint violation for each learning-based method. Without the DRO architecture, the surrogate-based method provides a relatively high-performing charging protocol that charges the battery cell in 32.35 min, only 5.7% slower than the CCCV approach. It also demonstrates improved safety relative to the industry CCCV benchmark. Specifically, the magnitude of the maximum constraint violation in the non-robust version of our algorithm is only 0.0082 V. With the added DRO framework based on Wasserstein ambiguity sets, we see that the charging protocol satisfies the constraint at every instance in time, while also providing a competitive 34.1 min charging time. These results illustrate the theoretical guarantees we expect from the application of Wasserstein ambiguity sets. Relative to the non-robust version, the charging time with the DRO offset is only 5.4% slower, a tradeoff that may be worthwhile for the increased safety and mitigation of aging. CQL violates overpotential constraints and charges slowly in comparison, however, we trained the CQL network with the exact same dataset as used by our method for consistency. An offline dataset with: 1) more trajectories and 2) trajectories that more frequently violate constraints would yield higher-performing CQL results, however, such results would not have any guarantees of adhering to constraints. Table IV shows a comparison of relevant results metrics.

| Algorithm                        | Charge Time [min] | Feasible? |
|----------------------------------|-------------------|-----------|
| CCCV                             | 30.6              | No        |
| CQL                              | 42                | No        |
| Surrogate Optimal Control (No DRO)| 32.35             | No        |
| Robust Surrogate Optimal Control | 34.1              | Yes       |

Table IV: Comparison of Relevant Experiment Metrics
the distributional shift problem which is a significant open

pairs in the final optimal charging profile. This plot highlights

with a ninth-generation Intel i5 processor. In [32], the average
runtime per iteration with the DFN is 2.0968 s, when the algo-

rithm is executed on a Windows desktop workstation equipped

with 2687 state variables. The average computation
time per iteration was 1.7803 s when run on the same machine.

Despite the more than tenfold increase in the cardinality of
the state vector of each model, the computational effort of
the proposed algorithm only changes marginally by 17.81%.
This slight difference is likely due to the more complex neural
network architecture and DRO framework which we employ
in our updated analysis.

D. Computational Effort Analysis
Comparing the computational requirements of this algo-
rithm to those of our preliminary version in [32] reveals a
host of meaningful insights. In this article, we are performing
optimal control on the DFN model, which is characterized by
208 state variables. In past exploratory work, we tested a
more rudimentary version of our algorithm on the single par-
ticle model with electrolyte and thermal dynamics (SPMeT),
a model with 208 state variables. The average computation
time per iteration with the DFN is 2.0968 s, when the algo-

rithm is executed on a Windows desktop workstation equipped

with a ninth-generation Intel i5 processor. In [32], the average
time per iteration was 1.7803 s when run on the same machine.

Despite the more than tenfold increase in the cardinality of
the state vector of each model, the computational effort of
the proposed algorithm only changes marginally by 17.81%.
This slight difference is likely due to the more complex neural
network architecture and DRO framework which we employ
in our updated analysis.

E. Insights From Wasserstein DRO Algorithm
One unique aspect of this work from preliminary results
presented in [32] is the application of Wasserstein ambiguity
sets. Wasserstein ambiguity sets are differentiated from
\( \phi \)-divergence-based chance constraint reformulation by their
robust out-of-sample safety guarantee. We see this difference
by observing that Wasserstein ambiguity sets provide a slightly
more conservative result that that shown in previous work.
This finding is clear from our DFN case study. The DRO
does prevent constraint violation entirely compared to the
non-robust version which only attenuates its magnitude relative
to CCCV. For safety-critical control applications, this added
safety from the out-of-sample safety guarantee. We see this difference
in our updated analysis.

V. CONCLUSION
This article presents a novel framework for optimal control
of high-dimensional dynamical systems. The key challenges
to numerical optimal control addressed by this article include:
1) the “curse of dimensionality” incurred by high-dimensional
systems; 2) formulations that are not linear-quadratic; and
3) ensuring safety/feasibility when constraint model errors
occur.

We identify surrogate models that learn from limited offline
datasets, and absorb state transition dynamics to reduce com-
pounded modeling errors. PCA applied to the training data
allows us to project the high-dimensional data on a reduced
basis. This makes the modeling architecture conducive to
fast identification and evaluation. Finally, we integrate these
models into a receding horizon control framework. Critically,
our strategy utilizes DRO to robustify the solution to errors
in the constraint function surrogate models. The OOD safety
guarantee of Wasserstein DRO directly addresses the open
challenge of distributional shift for offline RL problems.
All combined, we demonstrate that the algorithmic approach

challenge in offline RL research. Consider that when limited
to a static, offline dataset for model training, applying result-
ing control policies to a real, dynamical system creates the
opportunity for the agent to encounter states that fall out of the
distribution of its training data. For high-dimensional nonlinear
dynamical systems, the probability of this occurring is signif-
ificant. Thus, safety must be guaranteed with respect to such
OOD experience. Wasserstein ambiguity sets provide a strong
means to satisfy this requirement, given their out-of-sample
safety guarantee. While the final experimental distribution
does not represent the true underlying distribution of residuals,
it does present a significant deviation from what we observe
in our test data. Besides some slight differences in overall
shape, the experimental residual distribution is more heavily
skewed to higher magnitudes of modeling errors. Importantly
in this case the maximum residual we observe is 0.5033 V ,
which is 2.908 times the magnitude of the largest residual
represented in the test data set. This difference is just one
way of demonstrating how distributional errors can come into
play once we set out to apply an optimal charging policy.
yields tractable and robust control results for high-dimensional dynamical systems.

**APPENDIX**

**A. DFN Electrochemical Battery Model**

We consider the DFN model to predict the evolution of lithium concentration in the solid $c^±_s(x,r,t)$, lithium concentration in the electrolyte $c_e(x,t)$, solid electric potential $\phi^±_s(x,t)$, electrolyte electric potential $\phi_e(x,t)$, ionic current $i^±_n(x,t)$, molar ion fluxes $j^±_n(x,t)$, and battery temperature $T(t)$. The x-dimension runs across the negative electrode, separator, and positive electrode. At each x-coordinate value in the negative and positive electrodes, we consider a particle where spherical lithium intercalation occurs. The governing equations in time are given by

\[
\frac{\partial c^±_e(x,t)}{\partial t} = \frac{1}{r^2} \frac{\partial}{\partial r} \left[ D^±_e \frac{\partial c^±_e(x,t)}{\partial r} \right] + \epsilon^{±}_e \frac{\partial j^±_n(x,t)}{\partial x} \tag{29}
\]

\[
\frac{\partial j^±_n(x,t)}{\partial t} = \frac{\partial}{\partial x} \left[ D^±_n \frac{\partial j^±_n(x,t)}{\partial x} \right] + \frac{1 - r_0^4}{F} \epsilon^{±}_e \frac{\partial j^±_n(x,t)}{\partial x} \tag{30}
\]

for $j \in \{-, \text{sep}, +\}$ and $\dot{Q}$ is the rate of heat transferred to the system [40], defined as

\[
\dot{Q} = I(t)[U^+(t) - U^-(t) - V(t)] - I(t)T(t) \frac{\partial}{\partial T}[U^+(t) - U^-(t)] \tag{31}
\]

and differential equations in space and algebraic equations are given by

\[
\sigma^{\text{eff,±}} \frac{\partial \phi^±_e(x,t)}{\partial x} = i^±_e(x,t) - I(t) \tag{34}
\]

\[
\kappa^{\text{eff}}(c_e) \frac{\partial \phi^±_e(x,t)}{\partial x} = -t^±_e(x,t) + \kappa^±(c_e) \frac{2RT}{F} \left( 1 - \frac{t_0}{t} \right) \times \left( 1 + \frac{\partial \ln f_{i/a}}{\partial \ln c_e}(x,t) \right) \frac{\partial \ln \gamma_e}{\partial x}(x,t) \tag{35}
\]

\[
j^±_n(x,t) = \frac{1}{F} j^±_n(x,t) \left[ e^{\frac{\sigma^±(x,t)}{F} t^±(x,t)} - e^{-\frac{\sigma^±(x,t)}{F} t^±(x,t)} \right] \tag{36}
\]

\[
l^±_0(x,t) = \frac{1}{F} l^±_0(x,t) \left[ e^{\frac{\sigma^±(x,t)}{F} t^±(x,t)} - e^{-\frac{\sigma^±(x,t)}{F} t^±(x,t)} \right] \tag{37}
\]

\[
l^±_0(x,t) = k^±(\ell^±_{ss}(x,t)) \tag{38}
\]

\[
\eta^±(x,t) = \phi^±_e(x,t) - \phi^±_c(x,t) - U^±(\ell^±_{ss}(x,t)) \tag{39}
\]

\[
c^±_{ss}(x,t) = c^±_i(x,R^±_s,t) \tag{40}
\]

where $D^±_e = D_e(c_e) \cdot (c^±_e)^{\text{brag}}$, $\sigma^± = \sigma \cdot (c^±_e)^{\text{brag}}$, $\kappa^± = \kappa(c_e) \cdot (c^±_e)^{\text{brag}}$ are the effective electrolyte diffusivity, effective solid conductivity, and effective electrolyte conductivity given by the Bruggeman relationship. The boundary conditions for solid-phase diffusion PDE (29) are

\[
\frac{\partial c^±_e}{\partial r}(x,0,t) = 0 \tag{41}
\]

The boundary conditions for the electrolyte-phase diffusion PDE (30) are given by

\[
\frac{\partial c^±_e}{\partial x}(x,0,t) = 0 \tag{42}
\]

The boundary conditions for the electrolyte-phase potential ODE (35) are given by

\[
\phi^±_e(x,0,t) = 0 \tag{43}
\]

and also note that $i^±_e(x,t) = I(t)$ for $x \in [0^\text{sep}, L^\text{sep}]$. In addition, the parameters, $D^±_e$, $D_e$, $\kappa$, $k^±$ vary with temperature via the Arrhenius relationship

\[
\psi = \psi_\text{ref} \exp \left[ \frac{E_\psi}{R} \left( \frac{1}{T} - \frac{1}{T_\text{ref}} \right) \right] \tag{52}
\]

where $\psi$ represents a temperature-dependent parameter, $E_\psi$ is the activation energy and $\psi_\text{ref}$ is the reference parameter value at room temperature. The model input is the applied current density $I(t) (\text{A}/\text{m}^2)$, and the output is the voltage measured across the current collectors

\[
V(t) = \phi^+_c(0^+,t) - \phi^-_c(0^-,t). \tag{53}
\]

The level of charge in the cell is defined by the bulk SOC of the negative electrode, namely

\[
\text{SOC}^-(t) = \int_0^{L^-} \frac{\varepsilon^±_i(x,t)}{c^\text{max}_i(\theta^\text{100}\%, \theta^\text{0}\%)} L^- dx \tag{54}
\]

where $\varepsilon^±_i$ represents the volume averaged of a particle in the solid phase defined as

\[
\varepsilon^±_i(x,t) = \frac{3}{2R^i_s} \int_0^{R^±_s} r^2 c^±_i(r,t) dr. \tag{55}
\]

Lithium plating, which is the main battery degradation mechanism, is related to the side reaction overpotential $\eta_s$, defined as

\[
\eta_s(x,t) = \phi^+_s(x,t) - \phi^-_s(x,t) - U_{sr} \geq 0. \tag{56}
\]

To facilitate numerical optimal control, this model is discretized in space and time. There is a rich literature on discretization methods (see [29], [30]). The discretization approach used for this article involves finite difference, Padé approximation [31], and automatic differentiation methods.
REFERENCES

[1] D. E. Kirk, *Optimal Control Theory*. New York, NY, USA: Dover, 1970.

[2] L. T. Biegler, O. Ghattas, M. Heinkenschloss, and B. van Bloemen Waanders, “Large-scale PDE-constrained optimization: An introduction,” in *Large-Scale PDE-Constrained Optimization* (Lecture Notes in Computational Science and Engineering). Boston, MA, USA: Springer, 2003, pp. 3–13.

[3] G. Kerschen, J.-C. Golinval, A. F. Vakakis, and L. A. Bergman, “The method of proper orthogonal decomposition for dynamical characterization and order reduction of mechanical systems: An overview,” *Nonlinear Dyn.*, vol. 41, nos. 1–3, pp. 147–169, Aug. 2005.

[4] J. Hespanha, *Linear Systems Theory*. Princeton, NJ, USA: Princeton Univ. Press, 2009.

[5] G. R. Methekar, V. Reddy, R. D. Brautz, and V. R. Subramanian, “Optimum charging profile for lithium-ion batteries to maximize energy storage and utilization,” *ECS Trans.*, vol. 25, no. 35, pp. 139–146, Apr. 2010.

[6] M. J. Rothenberger, D. J. Docimo, M. Ghanadpishe, and H. K. Fathy, “Genetic optimization and experimental validation of a test cycle that maximizes parameter identifiability for a Li-ion equivalent-circuit battery model,” *J. Energy Storage*, vol. 4, pp. 156–166, Dec. 2015.

[7] J. Garcia and F. Fernandez, “A comprehensive survey on safe reinforcement learning with offline datasets,” 2020, arXiv:2006.09359.

[8] J. Jiang and Y. Guan, “Data-driven chance-constrained stochastic optimization using the Wasserstein metric: Performance guarantees and tractable reformulations,” *Math. Program.*, vol. 171, nos. 1–2, pp. 115–166, Sep. 2018.

[9] A. Nilim and L. E. Ghaoui, “Robust control of Markov decision processes with uncertain transition matrices,” *Oper. Res.*, vol. 53, no. 5, pp. 780–798, 2005.

[10] S. Zhao and Y. Guan, “Data-driven risk-averse stochastic optimization with Wasserstein metric,” *Oper. Res. Lett.*, vol. 46, no. 2, pp. 262–267, Mar. 2018.

[11] C. Duan, W. Fang, L. Jiang, L. Yao, and J. Liu, “Distributionally robust chance-constrained approximate AC-OPF with Wasserstein metric,” *IEEE Trans. Power Syst.*, vol. 33, no. 4, pp. 4924–4936, Sep. 2018.

[12] S. Mohan, Y. Kim, and A. G. Stefanopoulou, “Energy-conscious warm-up of Li-ion cells from subzero temperatures,” *IEEE Trans. Ind. Electron.*, vol. 63, no. 5, pp. 2954–2965, May 2016.

[13] K. Thomas, J. Newman, and R. Darling, “Mathematical modeling of lithium batteries,” in *Advances in Lithium-Ion Batteries*. Boston, MA, USA: Springer, 2002, pp. 345–392. [Online]. Available: http://www.springerlink.com/index/RXM87M4067U87J65.pdf

[14] Z. I. Botev, D. P. Kroese, R. Y. Rubinstein, and P. L’Ecuyer, “The cross-entropy method for optimization,” in *Handbook of Statistics*, vol. 31, C. Rao and V. Govindaraju, Eds. Amsterdam, The Netherlands: Elsevier, 2013, pp. 35–59. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780444538598000035

[15] S. Park et al., “A deep reinforcement learning framework for fast charging of Li-ion batteries,” *IEEE Trans. Transp. Electrific.*, vol. 8, no. 2, pp. 2770–2784, Jun. 2022.

[16] S. Park, D. Lee, H. J. Ahn, C. Tomlin, and S. Moura, “Optimal control of battery fast charging based-on Pontryagin’s minimum principle,” in *Proc. 59th IEEE Conf. Decis. Control (CDC)*, Dec. 2020, pp. 3506–3513.

[17] T. Yu et al., “MOPO: Model-based offline policy optimization,” 2020, arXiv:2005.13239.