Journal of The Electrochemical Society, **164** (4) A949-A959 (2017)
0013-4651/2017/164(4)/A949/11/$37.00 © The Electrochemical Society

Design of Piecewise Affine and Linear Time-Varying Model Predictive Control Strategies for Advanced Battery Management Systems

Marcello Torchio,[a] L. Magni,[a] Richard D. Braatz,[b] and D. M. Raimondo[a,z]

[a] *University of Pavia, 27100 Pavia, Italy*
[b] *Massachusetts Institute of Technology, Cambridge, Massachusetts 02142, USA*

Advanced Battery Management Systems (ABMSs) are necessary for the optimal and safe operation of Li-ion batteries. This article proposes the design of ABMSs based on Model Predictive Control (MPC). In particular, we consider MPC strategies based on piecewise affine approximations (PWAs) of a first-principles electrochemical battery model known in the literature as the pseudo two-dimensional (P2D) model. The use of model approximations is necessary since the P2D model is too complex to be included in the real-time calculations required by MPC. PWAs allow to well describe the electrochemical phenomena occurring inside the battery. On the other side, the accuracy of such models increases with the number of considered partitions, which also increases the model complexity and the online computational cost of MPC. Linear time-varying (LTV) approximations, which are obtained by linearizing accurate PWAs around a nominal trajectory, are proposed as a way to further reduce online computational costs. The obtained results demonstrate the suitability of MPC based on PWARX and LTV model approximations to provide ABMSs with high performance.

Manuscript submitted November 23, 2016; revised manuscript received February 21, 2017. Published March 8, 2017.

The ever increasing dependence on portable electrical devices calls for the development of reliable and efficient lightweight power sources. Modern laptops, music players, smartphones, tablets, and wearable health devices are only some of the applications that make use of batteries. In addition, the modern car market is moving toward the replacement of Internal Combustion Engines (ICEs) with electric powered engines. For these reasons, in recent years, research has focused on the development and production of portable electric power sources. As a result, different chemistries for Electrochemical Accumulators (EAs) have been introduced, with lead acid, NiMh, NiCd, Li-ion, and Li-po being the most common .[1–5] Batteries can be divided into two major categories: primary and secondary. Secondary batteries (SBs), which are rechargeable EAs, are of particular interest. Li-ion SBs represent the state of the art in battery technology.[6] While their electrochemical characteristics are remarkable, the use of Battery Management Systems (BMSs) is required in order to: (*i*) ensure safety, (*ii*) provide good operating performance, and (*iii*) prolong life. When charging Li-ion cells, most industrial BMSs rely on the so-called Constant Current-Constant Voltage (CC-CV) protocol.[7] The approach first applies a galvanostatic charge (the CC stage). Once the cell voltage reaches a threshold value (V_{thr}), a potentiostatic charge (the CV stage) is then applied. During this second phase, the current flowing through the cell decreases exponentially as a function of the governing physics. While the CC-CV approach provides reasonable performance, better results can be achieved by exploiting a mathematical model of the process.[8,9]

Advanced Battery Management Systems (ABMSs) rely on accurate mathematical descriptions to achieve effective control, monitoring, and diagnostics.[8] Two main classes of models are commonly used for this purpose: (*i*) Equivalent Circuit Models (ECMs), and (*ii*) Electrochemical Models (EMs). ECMs approximate the battery dynamics by means of electrical circuits composed mainly of resistors and capacitors.[10–13] In contrast, EMs rely on the first-principles physical laws and are able to represent all the physicochemical phenomena occurring inside a cell. Some examples of EMs are the pseudo two-dimensional, the single-particle, and the multiple-particle models.[14] EMs provide higher accuracy than ECMs but at the cost of an increased computational complexity.

One of the objectives of ABMSs is to provide suitable charging strategies. To achieve this goal, several control schemes have been proposed. The design of traditional PID control algorithms is addressed, in, for example, Refs. 15 and 16 while fuzzy logic techniques are considered in Refs. 17 and 18. The authors in Ref. 19 propose a modified reference governor control algorithm, while optimal control strategies are discussed in Refs. 20 and 21. Model Predictive Control (MPC) strategies have been considered by many authors (see e.g. Refs. 22–28).

Model predictive control refers to a class of algorithms that make explicit use of a process model to optimize the future predictive behavior of a plant, taking operating constraints into account.[29] Due to its characteristics, MPC may be suitable for the design of ABMSs. On the other hand, when dealing with nonlinear/complex/high-order systems, the computational time required to solve MPC online becomes intractable. For this reason, linearized approximations and/or reduced order dynamics of the controlled plant are usually adopted. In practice, this choice can still provide very satisfactory closed-loop stability and performance.

This article proposes MPC strategies for the optimal charging of a Li-ion cell based on linear and piece-wise linear model approximations. The use of linear models has been already considered in, e.g.,[26] where a finite step response model was used for the development of MPC based ABMSs. The approach showed interesting but still not completely satisfactory results, mainly due to the poor approximation of the temperature dynamics provided by a linear time-invariant model. Similarly, an MPC approach based on a finite step response model was considered in Ref. 27 where the aging of the battery was also modeled and controlled. While an interesting health-aware charging strategy was obtained, no advances were made in terms of model approximation. The use of ARX and hybrid models was introduced in Ref. 28 in order to better approximate the nonlinear behavior of the P2D model with simpler models suitable for control purposes. Although linearized models can provide interesting results, the strong nonlinearities driving the thermal dynamics make it extremely difficult to enforce temperature constraints using linear models. For this reason, this article considers also piecewise-affine approximations of the P2D model. In particular, AutoRegressive eXogenous (ARX) and PieceWise affine AutoRegressive eXogenous (PWARX) models are adopted. PWARX models are formulated as a set of affine dynamics, where binary variables are used to represent switches among different submodels. While this choice represents a compromise between accuracy and complexity, its online application within MPC becomes prohibitive as the number of submodels increases. With the aim of further reducing the computational burden, while still providing satisfactory results, LTV approximations of highly accurate PWARXs are also considered. These models are obtained by linearizing the dynamics around a nominal trajectory.[30–33] Such approximations are accurate in the neighborhood of the nominal trajectory, and have

[z] E-mail: davide.raimondo@unipv.it

the benefit of not relying on binary variables, therefore dramatically reducing the online computational cost. The effectiveness of the proposed control algorithms is demonstrated by comparing closed-loop performance on the P2D model. In all the proposed control scenarios, the objective is to charge the Li-ion cell to a reference value of the State Of Charge (SOC) while enforcing constraints on applied input current, temperature, voltage, SOC and anodic side-reaction overpotential. According to the obtained results, the use of PWARX/LTV models within MPC-based ABMSs provides significant improvements in terms of constraint satisfaction when compared to approaches based on linear models only.

With respect to the approaches in Refs. 26–28, we make the following contributions: (*i*) provide a detailed explanation of the identification procedure of models suitable for control purposes (*ii*) design control strategies based on highly accurate PWARX models (*iii*) derive LTV approximations for alleviating the computational burden related to PWARX models. Note that, the use of a highly detailed EM allows to investigate, analyze and even constrain internal electrochemical states which cannot be considered when using ECMs.

This article is organized as follows. Model for simulations: Li-ion pseudo two-dimensional (P2D) model section introduces the P2D model and its partial differential-algebraic formulation. Models for control section presents ARX, PWARX, and LTV models, and Model identification section describes the methodology used for their identification. Model predictive control section addresses the formulation of ARX-, PWARX-, and LTV-based MPC algorithms. Results section presents identification and closed-loop control results. Conclusions section is the conclusion.

Model for Simulations: Li-ion Pseudo Two-dimensional (P2D) Model

The pseudo two-dimensional (P2D) model consists of nonlinear and tightly coupled partial differential and algebraic equations (PDAEs), which are used to express the conservation of charge and mass within the three main sections of a Li-ion cell: (*i*) cathode, (*ii*) separator, and (*iii*) anode. The structure cathode-separator-anode is immersed within an electrolyte solution which facilitates the flow of ions between the electrodes. A porous separator avoids possible short circuits between the electrodes, while allowing the flow of ions. Finally, current collectors are connected to the electrodes in order to provide power to external devices. During a discharge cycle, the ions deintercalate from the anode and, flowing through the separator, intercalate into the cathode. The inverse process takes place during a charging phase. A schematic representation of a Li-ion cell is provided in Figure 1. In the following, the index $i \in \{a, p, s, n, z\}$ refers to a particular section of the battery, whereas p, s, and n refer to the cathode, separator, and anode respectively, while a and z refer to the positive and negative current collectors. For convenience, we introduce the quantities $\hat{x}_0 = l_a$, $\hat{x}_p = l_a + l_p$, $\hat{x}_s = l_a + l_p + l_s$, and $\hat{x}_n = l_a + l_p + l_s + l_n$, where l_i is the thickness of the ith section of the cell and $L_{\text{tot}} = \sum_i l_i$ represents the overall cell thickness. The ionic flux is defined as the sum of two quantities

$$j(x,t) = j_{\text{int}}(x,t) + j_{\text{side}}(x,t),$$

where $t \in \mathbb{R}^+$ represents the time, $x \in \mathbb{R}$ is the one-dimensional spatial variable along which ions flow, $j_{\text{int}}(x,t)$ is the intercalation/deintercalation ionic flux, while $j_{\text{side}}(x,t)$ accounts for side reactions during charging phases at the electrolyte-anode interface. The ionic flux $j(x,t)$ is zero inside the separator. The porous electrodes are composed of active particles in which ions intercalate or deintercalate. Different geometrical approximations of the shape of these particles have been proposed in literature.[34] This work treats the active particles as spheres as usually done in the literature, and the intercalation and deintercalation processes are modeled according to Fick's law of diffusion:

$$\frac{\partial c_s(r,t)}{\partial t} = \frac{1}{r^2}\frac{\partial}{\partial r}\left[r^2 D^s_{\text{eff},i}\frac{\partial c_s(r,t)}{\partial r}\right], \quad i \in \{p,n\}, \qquad [1]$$

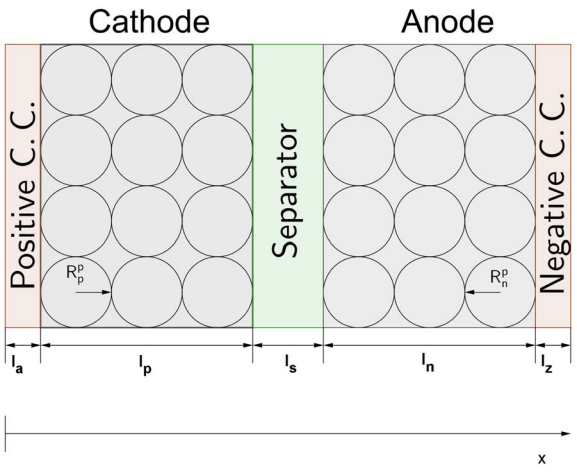

Figure 1. A schematic of a Li-ion cell.

with the BCs,

$$\left.\frac{\partial c_s(r,t)}{\partial r}\right|_{r=0} = 0, \quad \left.\frac{\partial c_s(r,t)}{\partial r}\right|_{r=R_i^p} = -\frac{j_{int}(x,t)}{D^s_{\text{eff},i}},$$

where r is the pseudo second dimension along which the ions diffuse within the particles, $c_s(r,t)$ the concentration of ions in the solid particles, $D^s_{\text{eff},i}$ the effective solid-phase diffusion coefficient and R_i^p the radius of the solid particles. The concentration of ions inside the electrolyte solution is represented with $c_e(x,t)$, and the conservation of these species is modeled by means of the diffusion equation:

$$\epsilon_i\frac{\partial c_e(x,t)}{\partial t} = \frac{\partial}{\partial x}\left[D_{\text{eff},i}\frac{\partial c_e(x,t)}{\partial x}\right] + a_i(1-t_+)j(x,t), \qquad [2]$$

where t_+ is the transference number, a_i the particle surface-area-to-volume ratio, ϵ_i the material porosity. The term $D_{\text{eff},i}$ accounts for the electrolyte effective diffusion coefficients according to the Bruggeman's theory. Zero-flux boundary conditions (BCs) are enforced to impose mass conservation in the liquid phase,

$$\left.\frac{\partial c_e(x,t)}{\partial x}\right|_{x=\hat{x}_0,\hat{x}_n} = 0, \qquad [3]$$

and continuity conditions are enforced at the interfaces across different materials,

$$-D_{\text{eff},p}\left.\frac{\partial c_e(x,t)}{\partial x}\right|_{x=\hat{x}_p^-} = -D_{\text{eff},s}\left.\frac{\partial c_e(x,t)}{\partial x}\right|_{x=\hat{x}_p^+}, \qquad [4]$$

$$-D_{\text{eff},s}\left.\frac{\partial c_e(x,t)}{\partial x}\right|_{x=\hat{x}_s^-} = -D_{\text{eff},n}\left.\frac{\partial c_e(x,t)}{\partial x}\right|_{x=\hat{x}_s^+}. \qquad [5]$$

The conservation of charge in the porous electrodes is modeled according to Ohm's law,

$$\frac{\partial}{\partial x}\left[\sigma_{\text{eff},i}\frac{\partial \Phi_s(x,t)}{\partial x}\right] = a_i F j(x,t), \quad i \in \{p,n\},$$

with the BCs,

$$\left.\sigma_{\text{eff},i}\frac{\partial \Phi_s(x,t)}{\partial x}\right|_{x=\hat{x}_0,\hat{x}_n} = -I_{\text{app}}(t),$$

$$\left.\sigma_{\text{eff},i}\frac{\partial \Phi_s(x,t)}{\partial x}\right|_{x=\hat{x}_p,\hat{x}_s} = 0,$$

where $\Phi_s(x,t)$ is the solid-phase potential, $\sigma_{\text{eff},i}$ the electrodes effective conductivity, F the Faraday's constant, and $I_{\text{app}}(t)$ the applied current density. The cell voltage is defined as $V_{\text{out}}(t) = \Phi_s(x_0,t) - \Phi_s(x_n,t)$. Similarly to the solid-phase potential, the electrolyte potential $\Phi_e(x,t)$ is modeled using a modified Ohm's law,

$$a_i F j(x,t) = -\frac{\partial}{\partial x}\left[\kappa_{\text{eff},i}\frac{\partial \Phi_e(x,t)}{\partial x}\right] + \frac{\partial}{\partial x}\left[\kappa_{\text{eff},i}\,\Xi\,T(x,t)\frac{\partial \ln c_e(x,t)}{\partial x}\right], \quad i \in \{p,s,n\},$$

where $\Xi = 2\frac{R(1-t_+)}{F}$, $\kappa_{\text{eff},i}$ is the effective electrolyte conductivity, R is the universal gas constant, and $T(x,t)$ is the temperature. Without loss of generality, given that only potential differences are measurable, the value of $\Phi_e(x,t)$ is set to zero at the anode side,

$$\Phi_e(x,t)\Big|_{x=\hat{x}_n} = 0,$$

while no-flux conditions are enforced at the other end of the cell,

$$\frac{\partial \Phi_e(x,t)}{\partial x}\Big|_{x=\hat{x}_0} = 0.$$

Continuity BCs are enforced across the different sections of the cell,

$$-\kappa_{\text{eff},p}\frac{\partial \Phi_e(x,t)}{\partial x}\Big|_{x=\hat{x}_p^-} = -\kappa_{\text{eff},s}\frac{\partial \Phi_e(x,t)}{\partial x}\Big|_{x=\hat{x}_p^+},$$

$$-\kappa_{\text{eff},s}\frac{\partial \Phi_e(x,t)}{\partial x}\Big|_{x=\hat{x}_s^-} = -\kappa_{\text{eff},n}\frac{\partial \Phi_e(x,t)}{\partial x}\Big|_{x=\hat{x}_s^+}.$$

Finally, thermal dynamics are included and coupled to the above set of equations

$$\rho_i C_{p,i}\frac{\partial T(x,t)}{\partial t} = \frac{\partial}{\partial x}\left[\lambda_i\frac{\partial T(x,t)}{\partial x}\right] + Q_{\text{ohm},i}(x,t) + Q_{\text{rxn},i}(x,t) + Q_{\text{rev},i}(x,t), \quad i \in \{a,p,s,n,z\},$$

where ρ_i is the material density, $C_{p,i}$ is the specific heat, λ_i is the heat diffusion coefficient, and Q_{ohm}, Q_{rev}, and Q_{rxn} are the ohmic, reversible, and reaction generation rates respectively. The thermal model is in accordance with Ref. 35 (a pouch type cell is considered). Continuity BCs are required to guarantee a continuous heat flux across the different sections of the battery,

$$-\lambda_a\frac{\partial T(x,t)}{\partial x}\Big|_{x=\hat{x}_0^-} = -\lambda_p\frac{\partial T(x,t)}{\partial x}\Big|_{x=\hat{x}_0^+},$$

$$-\lambda_p\frac{\partial T(x,t)}{\partial x}\Big|_{x=\hat{x}_p^-} = -\lambda_s\frac{\partial T(x,t)}{\partial x}\Big|_{x=\hat{x}_p^+},$$

$$-\lambda_s\frac{\partial T(x,t)}{\partial x}\Big|_{x=\hat{x}_s^-} = -\lambda_n\frac{\partial T(x,t)}{\partial x}\Big|_{x=\hat{x}_s^+},$$

$$-\lambda_n\frac{\partial T(x,t)}{\partial x}\Big|_{x=\hat{x}_n^-} = -\lambda_z\frac{\partial T(x,t)}{\partial x}\Big|_{x=\hat{x}_n^+},$$

while Newton's law of cooling is enforced at the end of the two current collectors to account for heat dissipation with the surrounding environment,

$$-\lambda_a\frac{\partial T(x,t)}{\partial x}\Big|_{x=0} = h_{\text{cell}}(T_{\text{env}} - T(x,t)),$$

$$-\lambda_z\frac{\partial T(x,t)}{\partial x}\Big|_{x=L} = h_{\text{cell}}(T(x,t) - T_{\text{env}}),$$

where h_{cell} is the heat exchange coefficient, and T_{env} the environmental temperature. A thorough description of the thermal model development can be found in Ref. 35. The above equations are coupled by means of the ionic flux. Given that the side reactions are considered to occur only at the electrolyte-anode interface, the contribution of $j_{side}(x,t)$ at the cathode side is null and no Solid-Electrolyte Interface (SEI) resistance between cathode and electrolyte is considered. According to,[36] the side reaction portion of the ionic flux is modeled with a Tafel relation of the form:

$$j_{\text{side}}(x,t) = -\frac{i_{0,\text{side}}(t)}{F}\exp\left(\frac{0.5F}{RT(x,t)}\eta_{\text{side}}(x,t)\right),$$

where $i_{0,\text{side}}(t)$ and $\eta_{\text{side}}(x,t)$ are respectively the side reaction exchange current and the side reaction overpotential. The side reaction exchange current $i_{0,\text{side}}(t)$ depends on the battery applied current density. No experimental data are available for the identification of such relation, and the empirical equation described in Ref. 37 is here adopted. The side reaction overpotential is defined as

$$\eta_{\text{side}}(x,t) = \Phi_s(x,t) - \Phi_e(x,t) - U_{SEI} - Fj(x,t)R_f(t)$$

where the term U_{SEI} represents the side reaction Open Circuit Voltage (OCV) and $Fj(x,t)R_f(t)$ accounts for an extra voltage drop due to the presence of the SEI resistance $R_f(t)$. To model the SEI layer behavior, the following equation is considered

$$\frac{\partial}{\partial t}\delta(x,t) = -\frac{M_w}{\rho}j_{\text{side}}(x,t),$$

where M_w is the molar weight of the electrode and $\delta(x,t)$ represents the film thickness. The overall film resistance is given by

$$R_f(t) = R_{\text{SEI}} + \frac{\bar{\delta}(t)}{\nu},$$

where ν is the admittance of the film, R_{SEI} is the initial SEI layer resistance, and

$$\bar{\delta}(t) = \frac{1}{l_n}\int_0^{l_n}\delta(x,t)\,dx.$$

Because of the presence of the SEI layer at the anode side, the contribution of $j_{\text{side}}(x,t)$ at the cathode side is null (the diffusion process of Li-ions within the electrodes is driven from $j_{\text{int}}(x,t)$ only). The intercalation part of the ionic flux is governed by the Butler-Volmer equation

$$j_{\text{int}}(x,t) = 2\frac{i_{0,\text{int}}(x,t)}{F}\sinh\left[\frac{0.5F}{RT(x,t)}\eta_{\text{int}}(x,t)\right],$$

where $i_{0,\text{int}}(x,t)$ is the exchange current density

$$i_{0,\text{int}}(x,t) = Fk_{\text{eff},i}\sqrt{c_e(x,t)(c_{s,i}^{\max} - c_s(R_i^p,t))c_s(R_i^p,t)}, \quad i \in \{p,n\},$$

with $c_{s,i}^{\max}$ the maximum solid-phase concentration. The electrode overpotential $\eta_{\text{int}}(x,t)$ at the anode side is defined as

$$\eta_{\text{int}}(x,t) = \Phi_s(x,t) - \Phi_e(x,t) - U_n - Fj(x,t)R_f(t)$$

while at the cathode side as

$$\eta_{\text{int}}(x,t) = \Phi_s(x,t) - \Phi_e(x,t) - U_p$$

where the terms U_p and U_n represent the cathode and anode OCV respectively. Due to its PDAE representation, the P2D model needs to be reformulated into a set of DAEs or AEs in order to be numerically solved. The P2D model has been developed and experimentally validated in Ref. 38. This article relies on LIONSIMBA[39] which provides a numerical implementation of the P2D model suitable for control purposes. The complete set of equations and parameters used in this work, together with a thorough description of the P2D model, can be found in Ref. 39.

The remainder of this article considers $I_{\text{app}}(t)$ as the input of the Li-ion cell, with the outputs being the voltage $V_{\text{out}}(t)$, the temperature at the negative current collector $T(L_{tot},t)$ (from here on denoted as $T(t)$), the side reaction overpotential at the end of the negative

electrode $\eta_{side}(\hat{x}_n, t)$ (from here on denoted as $\eta_{side}(t)$) and the State Of Charge (SOC) defined as

$$\text{SOC}(t) = \frac{1}{l_n c_{s,n}^{\max}} \int_{\hat{x}_s}^{\hat{x}_n} c_s^{\text{avg}}(x, t) dx$$

where $c_s^{\text{avg}}(x, t)$ is the average solid-phase concentration in each solid particle.

Models for Control

The P2D model provides a detailed description of the phenomena occurring inside a cell. Unfortunately, the complexity of this EM model makes it difficult to use it for control purposes.[40,41] For this reason, order reduction or linearization techniques may be employed.[42–48] In Refs. 26,27, we investigated the use of linear input-output approximations of the P2D model for the development of MPC strategies. With these models we obtained promising results for the case of constant-temperature operation. On the other side, such approximations do not allow to capture the nonlinear nature of the thermal behavior inside a Li-ion cell. When temperature variations and constraints are present, better approximations are required. For this reason, this article considers also piecewise affine and Linear Time Varying (LTV) input-output approximations. Below is a brief overview of the ARX, PWARX, and LTV models relevant to this study.

AutoRegressive eXogenous (ARX) models.—AutoRegressive eXogenous (ARX) models are a particular family of input-output linear models.[49] For a single-input single-output (SISO) system, the ARX one-step ahead output prediction scheme is

$$y(k+1) = \boldsymbol{\theta} \begin{bmatrix} \boldsymbol{s}(k) \\ 1 \end{bmatrix}, \qquad [6]$$

where y is the output, $\boldsymbol{s} \in \mathbb{R}^n$ is the array of *regressors*, $\boldsymbol{\theta} \in \mathbb{R}^{n+1}$ is the set of parameters, and k is the discrete time step. For a given time instant, $\boldsymbol{s}(k)$ is defined as

$$\begin{aligned} \boldsymbol{s}(k) = [&y(k)\ y(k-1)\ \cdots\ y(k-n_a+1) \\ &u(k+1-n_k)\ u(k-n_k)\ \cdots\ u(k-n_k-n_b+2)], \end{aligned}$$

which is composed of the past n_a output values and the past n_b input values, where $n_k \geq 0$ accounts for the *relative degree* of the system, and $n = n_a + n_b$. Model 6 can be straightforwardly extended to the multiple-input multiple-output (MIMO) case. Consider a system having n_u inputs and n_y outputs. Moreover, let the matrices $\boldsymbol{N_a} \in \mathbb{R}^{n_y \times n_y}$, $\boldsymbol{N_b} \in \mathbb{R}^{n_y \times n_u}$, and $\boldsymbol{N_k} \in \mathbb{R}^{n_y \times n_u}$, be analogous to the SISO indexes n_a, n_b, and n_k respectively. The element of row i and column j of $\boldsymbol{N_a}$ accounts for the number of past values of the jth output contributing to the dynamics of the ith output of the system. Similarly the effects of the mth input over the ith output are given by the element of $\boldsymbol{N_b}$ at row i and column m. Finally, the element of row i and column j of $\boldsymbol{N_k}$ accounts for the *relative degree* between input j and output i.

PieceWise affine AutoRegressive eXogenous models.—Define the *regressors set* $\mathcal{X} \subset \mathbb{R}^n$ as a bounded polyhedron that satisfies $\boldsymbol{s}(k) \in \mathcal{X}$ for all $\boldsymbol{s}(k)$ and is partitioned in L polyhedral subregions $\{\mathcal{X}_l\}_{l=1}^{L}$, such that $\mathcal{X} = \bigcup_l \mathcal{X}_l$ and $\mathcal{X}_i \cap \mathcal{X}_j = \emptyset, \forall i \neq j$.

In the case of a SISO system, the PWARX one-step ahead output prediction scheme is[50]

$$y(k+1) = \begin{cases} \boldsymbol{\theta_1} \begin{bmatrix} \boldsymbol{s}(k) \\ 1 \end{bmatrix}, & \text{if } \boldsymbol{s}(k) \in \mathcal{X}_1 \\ \boldsymbol{\theta_2} \begin{bmatrix} \boldsymbol{s}(k) \\ 1 \end{bmatrix}, & \text{if } \boldsymbol{s}(k) \in \mathcal{X}_2 \\ \quad\vdots & \\ \boldsymbol{\theta_L} \begin{bmatrix} \boldsymbol{s}(k) \\ 1 \end{bmatrix}, & \text{if } \boldsymbol{s}(k) \in \mathcal{X}_L \end{cases} \qquad [7]$$

where $\boldsymbol{\theta_l}$ is the set of parameters related to the lth subregion, and the active dynamics (i.e., the set of parameters $\boldsymbol{\theta_l}$ used to predict the next value of the output) are chosen according to the membership of $\boldsymbol{s}(k)$ to a polyhedral partition element $\mathcal{X}_l$. In this modeling approach, a set of affine ARX models are used to approximate the input-output behavior of a system. The SISO formulation 7 can be straightforwardly extended to a general MIMO plant by considering the matrices $\boldsymbol{N_a} \in \mathbb{R}^{n_y \times n_y}$, $\boldsymbol{N_b} \in \mathbb{R}^{n_y \times n_u}$, and $\boldsymbol{N_k} \in \mathbb{R}^{n_y \times n_u}$ as in Eq. 6.

State-space representation of ARX and PWARX models.—When dealing with MPC algorithms, it is more convenient to represent the dynamics of the prediction model in terms of a state-space formulation. According to realization theory,[51] several canonical forms (e.g., observability or controllability) can be adopted to reformulate an input-output model in terms of a state-space model. The adoption of such forms lead to state-space representations with dummy states that do not have a physical meaning. All of these considerations also hold for the PWARX models, where the switching dynamics have to be carefully treated. PWARX models can be reformulated as state-space models using two main alternative representations:[52] (i) the Mixed Logical Dynamical (MLD) scheme, and (ii) the PieceWise Linear Time-Invariant (PWLTI) scheme. This work adopts the MLD formulation, where a set of logic rules, physical laws, and constraints are used to represent the dynamical behavior of a switching system. With the state array $\boldsymbol{x}(k) \in \mathbb{R}^{n_x}$, the output array $\boldsymbol{y}(k) \in \mathbb{R}^{n_y}$, and the input array $\boldsymbol{u}(k) \in \mathbb{R}^{n_u}$, the one-step ahead prediction scheme of a MLD systems is

$$\begin{cases} \boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B_1}\boldsymbol{u}(k) + \boldsymbol{B_2}\boldsymbol{\delta}(k) + \boldsymbol{B_3}\boldsymbol{z}(k) \\ \boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k) + \boldsymbol{D_1}\boldsymbol{u}(k) + \boldsymbol{D_2}\boldsymbol{\delta}(k) + \boldsymbol{D_3}\boldsymbol{z}(k) \\ \boldsymbol{E_2}\boldsymbol{\delta}(k) + \boldsymbol{E_3}\boldsymbol{z}(k) \leq \boldsymbol{E_1}\boldsymbol{u}(k) + \boldsymbol{E_4}\boldsymbol{x}(k) + \boldsymbol{E_5} \end{cases} \qquad [8]$$

where the array $\boldsymbol{\delta}(k) \in \{0, 1\}^{n_\delta}$ represents the set of binary variables (logic rules) used to switch from one submodel to another, $\boldsymbol{z}(k) \in \mathbb{R}^{n_z}$ are *auxiliary variables*,[53] and $\boldsymbol{A}$, $\boldsymbol{B_1}$, $\boldsymbol{B_2}$, $\boldsymbol{B_3}$, $\boldsymbol{C}$, $\boldsymbol{D_1}$, $\boldsymbol{D_2}$, $\boldsymbol{D_3}$, $\boldsymbol{E_1}$, $\boldsymbol{E_2}$, $\boldsymbol{E_3}$, $\boldsymbol{E_4}$, and $\boldsymbol{E_5}$ are matrices of suitable dimensions. The model 8 should be constructed to be well posed, as done in this article.

Linear time-varying dynamics.—Consider the discrete-time dynamical system,

$$\begin{aligned} \boldsymbol{x}(k+1) &= \boldsymbol{f}(\boldsymbol{x}(k), \boldsymbol{u}(k), k), \\ \boldsymbol{y}(k) &= \boldsymbol{g}(\boldsymbol{x}(k), \boldsymbol{u}(k), k), \end{aligned} \qquad [9]$$

where $\boldsymbol{f}(\cdot)$ and $\boldsymbol{g}(\cdot)$ are nonlinear/complex algebraic functions. Consider a nominal input sequence $\tilde{\boldsymbol{u}}^{\mathbf{n}} \in \mathbb{R}^{n_u \times H_{\tilde{u}}}$, where $H_{\tilde{u}}$ is the length of such sequence, which, if fed into the system, results in a set of nominal states $\tilde{\boldsymbol{x}}^{\mathbf{n}} \in \mathbb{R}^{n_x \times H_{\tilde{u}}}$ and outputs $\tilde{\boldsymbol{y}}^{\mathbf{n}} \in \mathbb{R}^{n_y \times H_{\tilde{u}}}$. By linearizing 9 around the nominal trajectories $(\tilde{\boldsymbol{u}}^{\mathbf{n}}, \tilde{\boldsymbol{x}}^{\mathbf{n}})$, the LTV dynamics

$$\begin{aligned} \boldsymbol{\delta x}(k+1) &= \boldsymbol{A}(k)\boldsymbol{\delta x}(k) + \boldsymbol{B}(k)\boldsymbol{\delta u}(k), \\ \boldsymbol{\delta y}(k) &= \boldsymbol{C}(k)\boldsymbol{\delta x}(k) + \boldsymbol{D}(k)\boldsymbol{\delta u}(k), \end{aligned}$$

are obtained, where

$$\boldsymbol{A}(k) = \left.\frac{d\boldsymbol{f}}{d\boldsymbol{x}}\right|_{\tilde{\boldsymbol{x}}^{\mathbf{n}}(k), \tilde{\boldsymbol{u}}^{\mathbf{n}}(k)}, \quad \boldsymbol{B}(k) = \left.\frac{d\boldsymbol{f}}{d\boldsymbol{u}}\right|_{\tilde{\boldsymbol{x}}^{\mathbf{n}}(k), \tilde{\boldsymbol{u}}^{\mathbf{n}}(k)},$$

$$\boldsymbol{C}(k) = \left.\frac{d\boldsymbol{g}}{d\boldsymbol{x}}\right|_{\tilde{\boldsymbol{x}}^{\mathbf{n}}(k), \tilde{\boldsymbol{u}}^{\mathbf{n}}(k)}, \quad \boldsymbol{D}(k) = \left.\frac{d\boldsymbol{g}}{d\boldsymbol{u}}\right|_{\tilde{\boldsymbol{x}}^{\mathbf{n}}(k), \tilde{\boldsymbol{u}}^{\mathbf{n}}(k)},$$

and the terms $\boldsymbol{\delta x}(k) = \boldsymbol{x}(k) - \tilde{\boldsymbol{x}}^{\mathbf{n}}(k)$, $\boldsymbol{\delta u}(k) = \boldsymbol{u}(k) - \tilde{\boldsymbol{u}}^{\mathbf{n}}(k)$, and $\boldsymbol{\delta \tilde{y}}(k) = \boldsymbol{y}(k) - \tilde{\boldsymbol{y}}^{\mathbf{n}}(k)$ account for the deviations of the linearized dynamics from the nominal trajectories.

Model Identification

Model identification requires the collection of data that are sufficiently informative to describe the behavior of the system under

the operating conditions of interest.[49] Here the identification of the Li-ion cell dynamics is obtained by applying a set of different charging/discharging profiles to the P2D model. The procedures for identifying the ARX and PWARX models starting from the collected data are briefly summarized below for the SISO case.

ARX identification algorithm.—Consider a discrete-time SISO system at equilibrium conditions (u_0, y_0). The application of input variations $\boldsymbol{\delta\tilde{u}} \in \mathbb{R}^N$ (on top of u_0) produces output variations $\boldsymbol{\delta\tilde{y}} \in \mathbb{R}^N$ (on top of y_0). For chosen scalars n_a, n_b, and n_k, the array of regressors is defined as

$$\boldsymbol{s}(k) = [\delta y(k)\ \delta y(k-1)\ \cdots\ \delta y(k-n_a+1)$$
$$\delta u(k-n_k+1)\delta u(k-n_k)\ \cdots\ \delta u(k-n_k-n_b+2)].$$

According to 6, the complete characterization of an ARX model requires the definition of the parameter vector $\boldsymbol{\theta}$. In this work, the optimal $\boldsymbol{\theta}^*$ is chosen according to the Least Squares (LS) approach as

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} ||\boldsymbol{\Phi\theta} - \boldsymbol{\delta\tilde{y}}||_2^2, \qquad [10]$$

where

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{s}(1) & \boldsymbol{s}(2) & \cdots & \boldsymbol{s}(N) \\ 1 & 1 & \cdots & 1 \end{bmatrix}^\top \in \mathbb{R}^{N-1\times n+1}.$$

The explicit solution of 10 is $\boldsymbol{\theta}^* = (\boldsymbol{\Phi}^\top\boldsymbol{\Phi})^\dagger\boldsymbol{\Phi}^\top\boldsymbol{\delta\tilde{y}}$, where the symbol $\dagger$ is the matrix inverse operator.

PWARX identification algorithm.—The identification of PWARX models has been addressed by many authors. In Ref. 50 a modified *K-means* algorithm is proposed to cluster the measurement data, followed by a Weighted Least Squares (WLS) technique for parameter identification. A Kohonen neural network-based method was proposed in Ref. 54 for both clustering and identification of PWARX parameters, while[55] proposed a geometric-algebraic approach for the identification of hybrid systems. A thorough comparison of identification techniques for PWARX models is given in Ref. 56 and references therein. Although identification toolboxes for PWARX exist (e.g., HIT,[57] PWAOAFID[58]), the support for MIMO systems is limited, and the scalability with respect to the number of identification data points is poor. For these reasons, we implemented in Matlab[59] a tailored algorithm consisting of two steps:

1. Given the regressors set $\mathcal{X}$ and a number of centroids L to be positioned, find the optimal clustering configuration $\{\mathcal{X}^{*,l}\}_{l=1}^L$ according to the *K-means* algorithm.
2. For each $\mathcal{X}^{*,l}$, $l \in \{1, \cdots, L\}$, compute the optimal parameter vectors $\boldsymbol{\theta}_l^*$ according to the the LS approach in Eq. 10.

The next subsection briefly recalls the K-means algorithm.

K-means algorithm.—The objective of the *K-means* algorithm is to divide N points (which represent the data defined over the regressors set) into L clusters $\mathcal{X}^l$, $l \in \{1, \cdots, L\}$. Given a *distance function* $d(\boldsymbol{x}, \boldsymbol{y})$, a cluster $\mathcal{X}^l$ is defined as $\mathcal{X}^l = \{\boldsymbol{x} \mid d(\boldsymbol{x}, \boldsymbol{C}_l) \leq d(\boldsymbol{x}, \boldsymbol{C}_i),\ \forall i \neq l\}$, where $\boldsymbol{C}_l \in \mathbb{R}^n$ indicates the *centroid* of the lth cluster. The optimal cluster configuration $\mathcal{X}^* := \{\mathcal{X}^{*,1}, \mathcal{X}^{*,2}, \cdots, \mathcal{X}^{*,L}\}$ is obtained by applying an iterative procedure aiming to minimize the cost function

$$J(\mathcal{X}) := \sum_{l=1}^{L}\sum_{\boldsymbol{x}\in\mathcal{X}^l} d(\boldsymbol{x}, \boldsymbol{C}_l).$$

This work uses the Euclidean norm as the distance function $d(\boldsymbol{x}, \boldsymbol{y})$. Under this assumption, the optimal cluster configuration results in a set of polyhedra $\{\mathcal{X}^{*,l}\}_{l=1}^L$, such that $\mathcal{X}^* = \bigcup_l \mathcal{X}^{*,l}$ and $\mathcal{X}^{*,p} \cap \mathcal{X}^{*,j} = \emptyset$, $\forall p \neq j$. Once the LS approach has been applied on each region $\mathcal{X}^{*,l}$, the MLD formulation of the PWARX model can be obtained using the MPT Toolbox.[60]

LTV identification.—As discussed in Linear time-varying dynamics section, LTV representations are used to approximate dynamics around nominal trajectories of inputs and states. Such trajectories can be obtained in several ways, based on: (i) prior physical or heuristic knowledge of the plant,[61] (ii) some empirical rules, or (iii) optimization approaches that exploit a mathematical model of the plant.[62,63] When optimization-based approaches are applied over nonlinear/complex dynamics, computational expensive procedures are employed offline to obtain sub-optimal (or, if possible, optimal) solutions to the given problem. This work relies on this latter approach, which is discussed in MPC formulation for LTV systems section.

Model Predictive Control

MPC is used in a wide variety of industrial applications.[64,65] This control paradigm exploits mathematical models to provide an optimal control law by considering future references of the controlled plant, while ensuring the enforcement of operating constraints on the inputs and outputs. Denote the arrays of states, inputs, and outputs as

$$\begin{aligned} \tilde{\boldsymbol{x}}_{k+1:k+H_p+1|k} &= [\boldsymbol{x}(k+1|k)^\top \cdots \boldsymbol{x}(k+H_p+1|k)^\top]^\top, \\ \tilde{\boldsymbol{u}}_{k:k+H_u|k} &= [\boldsymbol{u}(k|k)^\top \cdots \boldsymbol{u}(k+H_u|k)^\top]^\top, \\ \tilde{\boldsymbol{y}}_{k:k+H_p|k} &= [\boldsymbol{y}(k|k)^\top \cdots \boldsymbol{y}(k+H_p|k)^\top]^\top, \end{aligned} \qquad [11]$$

where H_p and H_u are the prediction and control horizons, respectively, and $\boldsymbol{x}(i|j)$ denotes the value of $\boldsymbol{x}$ at time instant i starting from time instant j. For computational reasons, it is common to define $H_u \leq H_p$ in order to reduce the number of optimization variables. When $H_u < H_p$, the control moves are kept fixed after the $k + H_u$ (i.e., $\boldsymbol{u}(k+H_u|k) = \boldsymbol{u}(k+H_u+1|k) = \cdots = \boldsymbol{u}(k+H_p|k)$). At every discrete time instant k, MPC provides an optimal input sequence by solving an online optimization,

$$\min_{\tilde{\boldsymbol{u}}_{k:k+H_u|k}} J(\tilde{\boldsymbol{u}}_{k:k+H_u|k}, \boldsymbol{x}_{\mathbf{init}}) \qquad [12]$$

subject to

$$\boldsymbol{x}(j+1|k) = \boldsymbol{f}(\boldsymbol{x}(j|k), \boldsymbol{u}(j|k), j) \qquad [13]$$

$$\begin{aligned} \boldsymbol{y}(j|k) &= \boldsymbol{g}(\boldsymbol{x}(j|k), \boldsymbol{u}(j|k), j) \\ \boldsymbol{x}(k|k) &= \boldsymbol{x}_{\mathbf{init}} \\ \boldsymbol{u}_{\min} &\leq \boldsymbol{u}(j|k) \leq \boldsymbol{u}_{\max} \\ \boldsymbol{x}_{\min} &\leq \boldsymbol{x}(j|k) \leq \boldsymbol{x}_{\max} \\ \boldsymbol{y}_{\min} &\leq \boldsymbol{y}(j|k) \leq \boldsymbol{y}_{\max}, \end{aligned} \qquad [14]$$

where $j \in [k; k+H_p]$,

$$J(\tilde{\boldsymbol{u}}_{k:k+H_u|k}, \boldsymbol{x}_{\text{init}}) = ||\tilde{\boldsymbol{u}}_{k:k+H_u|k} - \tilde{\boldsymbol{u}}_{\text{ref}}||_R^2 + ||\tilde{\boldsymbol{y}}_{k:k+H_p|k} - \tilde{\boldsymbol{y}}_{\text{ref}}||_{Q_y}^2, \qquad [15]$$

and $\tilde{\boldsymbol{u}}_{\text{ref}}$, and $\tilde{\boldsymbol{y}}_{\text{ref}}$ are reference setpoints (or trajectories) for the inputs and outputs respectively, the notation $||\boldsymbol{x}||_Q = ||Q^{1/2}\boldsymbol{x}||_2$ denotes the weighted 2-norm, $Q_y \in \mathbb{R}^{H_p n_y \times H_p n_y}$ is a positive semidefinite matrix that weights the deviation of the outputs from their references, and $R \in \mathbb{R}^{H_u n_u \times H_u n_u}$ is a positive definite matrix that accounts for the deviation of the inputs with respect to their reference value. The solution of the optimization provides an optimal control sequence $\tilde{\boldsymbol{u}}^*_{k:k+H_u|k}$. Given that $\tilde{\boldsymbol{u}}^*_{k:k+H_u|k}$ is obtained based only on the predictions of a mathematical model, its entire application to the plant would not reject unknown disturbances to the system. For this reason, the so-called *Receding Horizon* (RH) approach is adopted where, at each time step k, only the first element of the optimal control sequence is applied to the plant,[66–68]

$$\boldsymbol{u}_{\text{RH}}(k) = \boldsymbol{u}^*(k|k). \qquad [16]$$

At the next time instant, the array of the initial states $\boldsymbol{x}_{\mathbf{init}}$ is updated with the new measurements and a new optimization is solved.

MPC formulation for LTI systems.—Consider a Linear Time-Invariant (LTI) description of the process to be controlled, which has

$$\begin{aligned} \boldsymbol{f}(\boldsymbol{x}(k), \boldsymbol{u}(k), k) &= \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k), \\ \boldsymbol{g}(\boldsymbol{x}(k), \boldsymbol{u}(k), k) &= \boldsymbol{C}\boldsymbol{x}(k) + \boldsymbol{D}\boldsymbol{u}(k), \end{aligned} \quad [17]$$

with the cost function 15. In this case, the optimization 12 is a Quadratic Program (QP). In particular, by replacing the states with their explicit dependence on the inputs, it is possible to obtain suitable matrices $\underline{\boldsymbol{A}} \in \mathbb{R}^{n_c \times n_u \cdot H_u}$, $\boldsymbol{H} \in \mathbb{R}^{H_u n_u \times H_u n_u}$ and vectors $\underline{\boldsymbol{b}} \in \mathbb{R}^{n_c}$, $\boldsymbol{f} \in \mathbb{R}^{H_u n_u}$ such that 12 can be rewritten as

$$\min_{\tilde{\boldsymbol{u}}_{k:k+H_u|k}} \frac{1}{2} \tilde{\boldsymbol{u}}^{\top}_{k:k+H_u|k} \boldsymbol{H} \tilde{\boldsymbol{u}}_{k:k+H_u|k} + \boldsymbol{f}^{\top} \tilde{\boldsymbol{u}}_{k:k+H_u|k} \quad [18]$$

subject to

$$\underline{\boldsymbol{A}} \tilde{\boldsymbol{u}}_{k:k+H_u|k} \leq \underline{\boldsymbol{b}}$$

MPC formulation for MLD systems.—When MLD systems 8 are considered, due to the presence of binary variables $\tilde{\boldsymbol{\delta}}_{k:k+H_p|k} \in \{0, 1\}^{n_\delta H_p}$, the optimization 12 becomes a Mixed Integer QP (MIQP),

$$\min_{\tilde{\boldsymbol{\Upsilon}}} \frac{1}{2} \tilde{\boldsymbol{\Upsilon}}^{\top} \boldsymbol{H} \tilde{\boldsymbol{\Upsilon}} + \boldsymbol{f}^{\top} \tilde{\boldsymbol{\Upsilon}} \quad [19]$$

subject to

$$\begin{aligned} \underline{\boldsymbol{A}} \tilde{\boldsymbol{\Upsilon}} &\leq \underline{\boldsymbol{b}} \\ \tilde{\boldsymbol{\delta}}(j|k) &\in \{0, 1\}, \end{aligned}$$

where $j \in [k; k + H_p]$ and $\tilde{\boldsymbol{\Upsilon}} = [\tilde{\boldsymbol{u}}^{\top}_{k:k+H_u|k}, \tilde{\boldsymbol{\delta}}^{\top}_{k:k+H_p|k}, \tilde{\boldsymbol{z}}^{\top}_{k:k+H_p|k}]^{\top}$ is a suitable array containing all the problem variables. Due to the presence of binary variables, the complexity of the optimization is always higher than in the LTI/LTV case.

MPC formulation for LTV systems.—Finally, when a Linear Time-Varying (LTV) description,

$$\begin{aligned} \delta\boldsymbol{x}(k+1) &= \boldsymbol{A}(k)\delta\boldsymbol{x}(k) + \boldsymbol{B}(k)\delta\boldsymbol{u}(k), \\ \delta\boldsymbol{y}(k) &= \boldsymbol{C}(k)\delta\boldsymbol{x}(k) + \boldsymbol{D}(k)\delta\boldsymbol{u}(k), \end{aligned} \quad [20]$$

is considered with the cost function 15, the optimization 12 is still a QP. However, differently from the LTI case, the matrices $\boldsymbol{H}(k)$, $\underline{\boldsymbol{A}}(k)$ and the arrays $\boldsymbol{f}(k)$, $\underline{\boldsymbol{b}}(k)$ are time varying and need to be recomputed at each time step k. By introducing the optimization variables $\delta\tilde{\boldsymbol{u}}_{k:k+H_u|k}$ similarly to 11, the reformulation

$$\min_{\delta\tilde{\boldsymbol{u}}_{k:k+H_u|k}} \frac{1}{2} \delta\tilde{\boldsymbol{u}}^{\top}_{k:k+H_u|k} \boldsymbol{H}(k) \delta\tilde{\boldsymbol{u}}_{k:k+H_u|k} + \boldsymbol{f}^{\top}(k) \delta\tilde{\boldsymbol{u}}_{k:k+H_u|k} \quad [21]$$

subject to

$$\underline{\boldsymbol{A}}(k) \delta\tilde{\boldsymbol{u}}_{k:k+H_u|k} \leq \underline{\boldsymbol{b}}(k)$$

is obtained. LTV systems can be used to approximate nonlinear/complex dynamics around a nominal trajectory. The latter can be computed by solving, at a given initial time k_0, an optimization of the form

$$\min_{\tilde{\boldsymbol{u}}^{\mathbf{n}}_{k_0:k_0+H_{\bar{u}}}} J(\tilde{\boldsymbol{u}}^{\mathbf{n}}_{k_0:k_0+H_{\bar{u}}}, \boldsymbol{x}_{\text{init}}) \quad [22]$$

subject to

$$\boldsymbol{x}^{\mathbf{n}}(j+1) = \boldsymbol{f}(\boldsymbol{x}^{\mathbf{n}}(j), \boldsymbol{u}^{\mathbf{n}}(j), j) \quad [23]$$

$$\begin{aligned} \boldsymbol{y}^{\mathbf{n}}(j) &= \boldsymbol{g}(\boldsymbol{x}^{\mathbf{n}}(j), \boldsymbol{u}^{\mathbf{n}}(j), j) \\ \boldsymbol{x}^{\mathbf{n}}(k_0) &= \boldsymbol{x}_{\text{init}} \\ \boldsymbol{u}_{\min} &\leq \boldsymbol{u}^{\mathbf{n}}(j) \leq \boldsymbol{u}_{\max} \\ \boldsymbol{x}_{\min} &\leq \boldsymbol{x}^{\mathbf{n}}(j) \leq \boldsymbol{x}_{\max} \\ \boldsymbol{y}_{\min} &\leq \boldsymbol{y}^{\mathbf{n}}(j) \leq \boldsymbol{y}_{\max} \end{aligned} \quad [24]$$

where $j \in [k_0; k_0 + H_{\bar{u}}]$ and $\tilde{\boldsymbol{u}}^{\mathbf{n}}_{k_0:k_0+H_{\bar{u}}} \in \mathbb{R}^{n_u H_{\bar{u}}}$, $\tilde{\boldsymbol{x}}_{k_0:k_0+H_{\bar{u}}} \in \mathbb{R}^{n_x H_{\bar{u}}}$, and $\tilde{\boldsymbol{y}}_{k_0:k_0+H_{\bar{u}}} \in \mathbb{R}^{n_y H_{\bar{u}}}$ are defined similarly to 11. Due to the nonlinear/complex dynamics in Eq. 23, this optimization is solved offline. To obtain a nominal trajectory useful for control purposes, the optimization is solved over a horizon sufficiently long to guarantee the attainment of a neighborhood of the desired target.

LTV closed-loop MPC scheme.—Starting from $k = k_0$, the optimal solution $\tilde{\boldsymbol{u}}^{\mathbf{n},*}_{k_0:k_0+H_{\bar{u}}}$ and $\tilde{\boldsymbol{x}}^{\mathbf{n},*}_{k_0:k_0+H_{\bar{u}}}$ provided by Eq. 22 are applied within a MPC context as:

1. Obtain the LTV approximation of the nonlinear/complex dynamics using the sub-sequences $\tilde{\boldsymbol{u}}^{\mathbf{n},*}_{k:k+H_p}$ and $\tilde{\boldsymbol{x}}^{\mathbf{n},*}_{k:k+H_p}$
2. At time k, solve the optimization 21 to obtain $\delta\tilde{\boldsymbol{u}}_{k:k+H_u|k}$ and update the subsequence $\tilde{\boldsymbol{u}}^{\mathbf{n},*}_{k:k+H_u} \leftarrow \tilde{\boldsymbol{u}}^{\mathbf{n},*}_{k:k+H_u} + \delta\tilde{\boldsymbol{u}}_{k:k+H_u|k}$
3. According to the RH approach 16, apply $\boldsymbol{u}^{\mathbf{n},*}(k)$ to the real plant
4. If $k + 1 + H_p > k_0 + H_{\bar{u}}$ then extend the nominal input trajectory by repeating its last element, i.e.,

$$\begin{aligned} \tilde{\boldsymbol{u}}^{\mathbf{n},*}_{k+1:k+1+H_p} = [&\boldsymbol{u}^{\mathbf{n},*}(k+1), \boldsymbol{u}^{\mathbf{n},*}(k+2), \cdots, \boldsymbol{u}^{\mathbf{n},*}(k+H_{\bar{u}}), \\ &\boldsymbol{u}^{\mathbf{n},*}(k+H_{\bar{u}})]. \end{aligned}$$

5. Update the initial states $\delta\boldsymbol{x}_{\text{init}}$ using the plant measurements
6. Compute $\tilde{\boldsymbol{x}}^{\mathbf{n},*}_{k+1:k+1+H_p}$ according to $\tilde{\boldsymbol{u}}^{\mathbf{n},*}_{k+1:k+1+H_p}$
7. Set $k \leftarrow k + 1$ and go back to Step 1

Constraint softening.—When dealing with nonlinear systems, such as the P2D, the use of model approximations for control purposes brings inevitable mismatch which may lead to constraint violations on the real plant. A common approach to guarantee feasibility is to *soften* the state and output constraints by suitably modifying the optimization. The softening of the outputs constraints can be carried out by adding a set of optimization variables to the general formulation 12, which leads to the optimization

$$\min_{\tilde{\boldsymbol{u}}_{k:k+H_u|k}, \tilde{\boldsymbol{\Gamma}}_{k:k+H_p|k}} \hat{J}(\tilde{\boldsymbol{u}}_{k:k+H_u|k}, \tilde{\boldsymbol{\Gamma}}_{k:k+H_p|k}, \boldsymbol{x}_{\text{init}}) \quad [25]$$

subject to

$$\boldsymbol{x}(j+1|k) = \boldsymbol{f}(\boldsymbol{x}(j|k), \boldsymbol{u}(j|k), j) \quad [26]$$

$$\begin{aligned} \boldsymbol{y}(j|k) &= \boldsymbol{g}(\boldsymbol{x}(j|k), \boldsymbol{u}(j|k), j) \\ \boldsymbol{x}(k|k) &= \boldsymbol{x}_{\text{init}} \\ \boldsymbol{u}_{\min} &\leq \boldsymbol{u}(j|k) \leq \boldsymbol{u}_{\max}, \\ \boldsymbol{x}_{\min} &\leq \boldsymbol{x}(j|k) \leq \boldsymbol{x}_{\max} \\ \boldsymbol{y}_{\min} - \boldsymbol{\Gamma}(j|k) &\leq \boldsymbol{y}(j|k) \leq \boldsymbol{y}_{\max} + \boldsymbol{\Gamma}(j|k), \\ \boldsymbol{\Gamma}(j|k) &\geq 0, \end{aligned} \quad [27]$$

where $\tilde{\boldsymbol{\Gamma}}_{k:k+H_p|k} \in \mathbb{R}^{n_y H_u}$ is defined similarly to 11, the cost function $\hat{J}$ is defined as

$$\hat{J}(\tilde{\boldsymbol{u}}_{k:k+H_u|k}, \tilde{\boldsymbol{\Gamma}}_{k:k+H_p|k}, \boldsymbol{x}_{\text{init}}) = J(\tilde{\boldsymbol{u}}_{k:k+H_u|k}, \boldsymbol{x}_{\text{init}}) + ||\tilde{\boldsymbol{\Gamma}}_{k:k+H_p|k}||^2_{\boldsymbol{Q}_{\boldsymbol{\Gamma}}},$$

and $\boldsymbol{Q}_{\boldsymbol{\Gamma}} \in \mathbb{R}^{H_p n_y \times H_p n_y}$ is a positive semidefinite matrix used to weight the constraint violations.

Results

Identification results.—To identify the ARX and PWARX approximations of the Li-ion cell dynamics, a set of charging and discharging input profiles were applied to the P2D model. The cell starts from a steady-state condition characterized by $\text{SOC}_0 = 20\%$, $V_{\text{out},0} = 3.73$ V, $T_0 = 298.15$ K and $\eta_{side,0} = 0.15$ V, which corresponds to the application of $I_{\text{app},0} = 0$ A m^{-2}. The identification dataset $\mathcal{I}$ (i.e., the

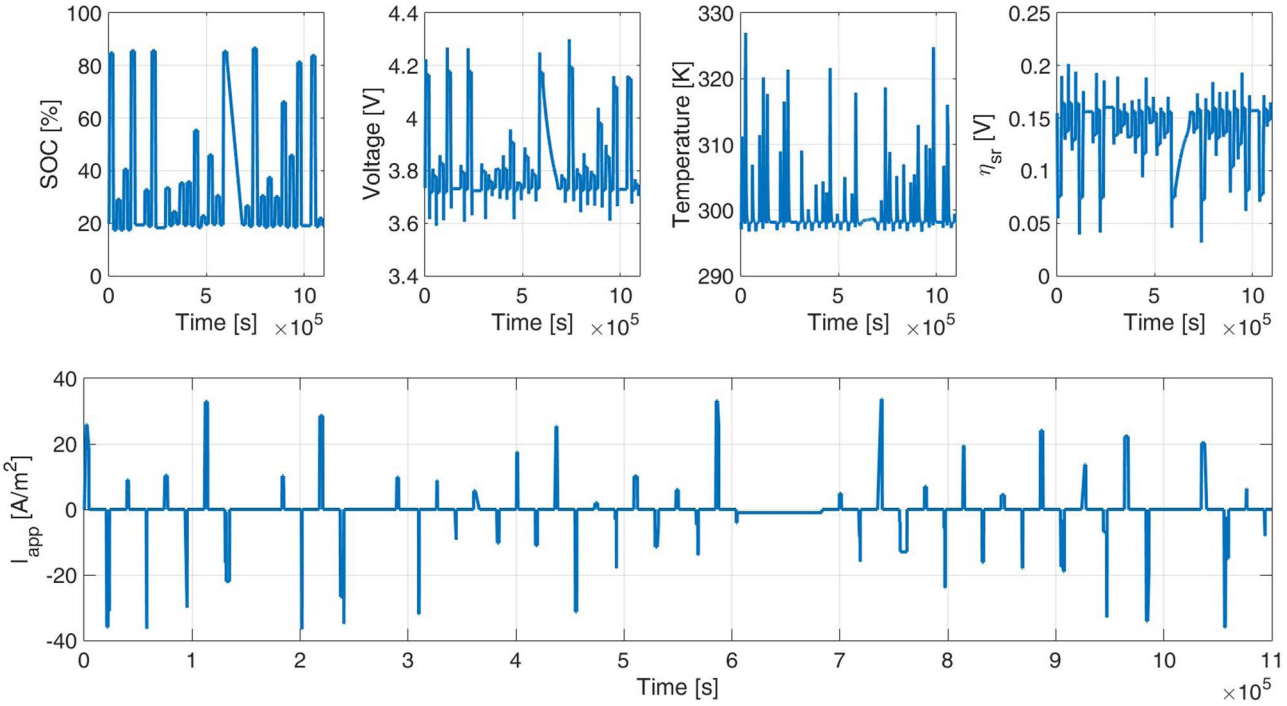

Figure 2. Li-ion cell data used for the identification of the PWARX models.

values of $I_{\text{app}}(t)$, V_{out}, $T(t)$, and SOC(t)) are reported in Fig. 2. A similar dataset, $\mathcal{V}$, was used for validation purposes. ARX and PWARX models were obtained according to the procedures summarized in ARX identification algorithm and PWARX identification algorithm sections respectively, with $\boldsymbol{N_a}$, $\boldsymbol{N_b}$, and $\boldsymbol{N_k}$ as

$$\boldsymbol{N_a} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 2 & 0 & 0 \\ 2 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \boldsymbol{N_b} = \begin{bmatrix} 1 \\ 2 \\ 5 \\ 5 \end{bmatrix}, \quad \boldsymbol{N_k} = \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \end{bmatrix}. \qquad [28]$$

This choice was made upon the analysis of the cell dynamics. Note that, the first row of $\boldsymbol{N_a}$ refers to the SOC(t), the second to $V_{\text{out}}(t)$, the third to $T(t)$ and the fourth to $\eta_{side}(t)$. As shown in Fig. 2, the SOC(t) exhibits an integral behavior with respect to the applied current density, whereas $V_{\text{out}}(t)$ also exhibits higher order dynamics during relaxation periods. Moreover, the dynamics of the voltage shows a direct feedthrough with respect to $I_{\text{app}}(t)$, which motivates the structure of $\boldsymbol{N_k}$. The temperature is mainly affected by the applied current density and the SOC. Whereas SOC(t), $V_{\text{out}}(t)$ and $\eta_{side}(t)$ are well described by linear models, a linear approximation of $T(t)$ provides poor performance (see first row of Table I, where the results refer to the dataset $\mathcal{V}$). Indeed, inspection of Fig. 3 indicates that (i) undershoots in the temperature profile can be observed for certain values of the SOC, (ii) a change of rising slope can be found according to the duration of the charging or discharging current, and (iii) the temperature rises in the presence of either positive or negative values of $I_{\text{app}}(t)$. To better capture these phenomena, a PWARX model was employed to approximate the temperature dynamics, while the behaviors of SOC, voltage and side reaction overpotential at the end of the negative electrode are still described with ARX models. The choice made in Eq. 28 implies that the temperature regressors set $\mathcal{X}_T \subset \mathbb{R}^8$. Two different PWARX models were identified, with 4 and 8 clusters respectively. The performance of the obtained approximations are reported in Table I, where the fitness function J_{fit} is computed as

$$J_{\text{fit}} = 100\% \left(1 - \frac{\left\| \boldsymbol{y}_v - \hat{\boldsymbol{y}} \right\|_2}{\left\| \boldsymbol{y}_v - \bar{\boldsymbol{y}}_v \right\|_2} \right), \qquad [29]$$

with $\boldsymbol{y}_v$ the vector of outputs of the validation dataset $\mathcal{V}$, $\hat{\boldsymbol{y}}$ the predictions obtained according to a particular model, and $\bar{\boldsymbol{y}}_v$ the mean value of $\boldsymbol{y}_v$. Since the ARX models used in the three experiments are the same for SOC(t), $V_{\text{out}}(t)$ and $\eta_{side}(t)$, their fitness values remain unchanged. On the other hand, the use of PWARX models improves significantly the approximation of the temperature behavior as observed in Fig. 4, which compares the temperature profiles for a subset of the validation profile. In particular, the ARX model sometimes even fails to identify the correct sign of the temperature change. The PWARX models produce much more accurate temperature predictions, even when only a few clusters are considered. Fig. 5 reports the

Table I. Comparison of the fitness function 29 among different models. Where the number of clusters is 0, a linear ARX model is used.

# of clusters	SOC	Voltage	Temperature	η_{sr}
0	97%	83%	−4%	87%
4	97%	83%	69%	87%
8	97%	83%	89%	87%

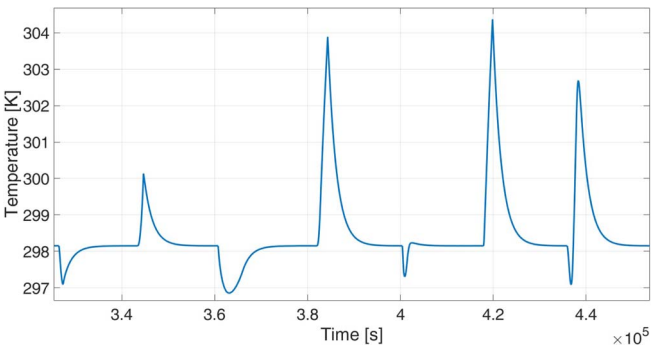

Figure 3. Temperature profile for the identification dataset $\mathcal{I}$.

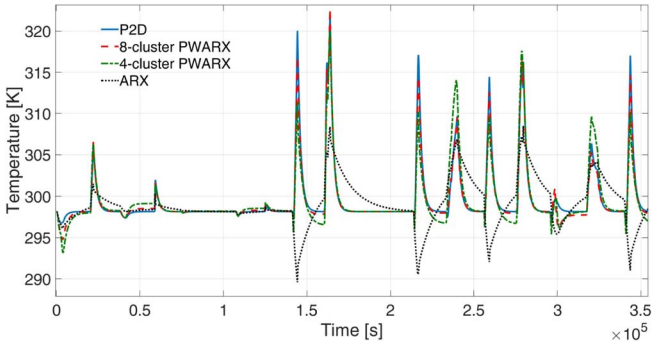

Figure 4. The temperature profiles for three different identified models and the P2D dynamics using the $\mathcal{V}$ dataset.

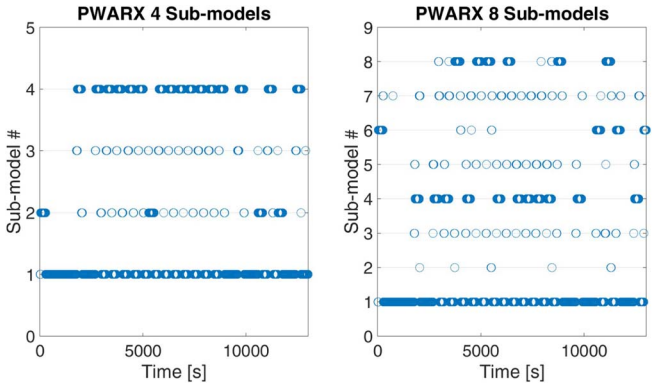

Figure 5. Switches among the different submodels for the 4- and 8-cluster PWARX models. These data were obtained during the validation of the hybrid models.

switches among the different submodels, for PWARX models with 4 and 8 clusters.

Control results.—According to Identification results section, the PWARX model with 8 clusters was the most accurate approximation of the P2D model. On the other hand, due to its hybrid nature, its model form has the highest online computational cost when used in an MPC algorithm. Indeed, its MPC formulation would require the solution of an MIQP at each time instant, with an online computational cost of $\sim$3600 s[c], which is much higher than the sampling time $T_s = 80$ s. To reduce the online computational cost while still providing satisfactory results, an LTV approximation of the 8-cluster PWARX model was computed.

The nominal trajectory used to compute the 8-cluster LTV model was obtained according to problem 23 with initial condition $\text{SOC}_0 = 20$ %, $V_{\text{out},0} = 3.79$ V, $T_0 = 298.15$ K and $\eta_{side,0} = 0.15$ V (steady-state condition corresponding to $I_{\text{app},0} = 0$ A m^{-2}), $\boldsymbol{y} = [\text{SOC}\, V_{\text{out}}\, T\, \eta_{side}]^\top$, $\boldsymbol{u} = I_{\text{app}}$, $H_{\bar{u}} = 30$ steps, and optimization parameters as in Table II. In this case, the optimization 23 was solved without any softening of the output constraints.

To optimally charge the Li-ion cell, MPC strategies based on ARX, PWARX, and LTV models were designed using the same parameters as in Table II and $H_p = H_u = 30$ steps. The effectiveness of the different controllers was evaluated using the P2D model as the real plant. Due to the mismatch between the P2D dynamics and the models used for control, soft constraints were considered as in Eq. 25 with $\boldsymbol{Q}_\Gamma = \text{diag}\{3000, 3000, 3000, 3000\}$. All simulations were performed using Matlab and the numerical implementation of the P2D model provided by the LIONSIMBA toolbox,[39] which relies on the numerical integrator IDAS[70] to solve the resulting set of DAEs. The MPT Toolbox[60] was used to compute the MLD dynamics, and the commercial solver CPLEX[69] to solve the resulting MIQP and QP problems on a i5@ 2.7-GHs 64-bit CPU system with 16 Gbytes of RAM running Windows 10 Pro.

[c]The commercial solver CPLEX[69] was used to solve the resulting MIQP and QP problems on a i5@ 2.7-GHz 64-bit CPU system with 16 Gbytes of RAM running Windows 10 Pro.

Table II. Optimization parameters.

Parameter	Value
$\boldsymbol{y}_{\min}$	$[0\%,\ 2.5\,\text{V},\ 290\,\text{K},\ 0\,\text{V}]^\top$
$\boldsymbol{y}_{\max}$	$[95\%,\ 4.2\,\text{V},\ 303.65\,\text{K},\ -\,\text{V}]^\top$
$\boldsymbol{u}_{\min}$	0 A m^{-2}
$\boldsymbol{u}_{\max}$	40 A m^{-2}
$\Delta\boldsymbol{u}_{\min}$	-10 A m^{-2} s^{-1}
$\Delta\boldsymbol{u}_{\max}$	10 A m^{-2} s^{-1}
$\boldsymbol{Q}_y$	diag{10, 0, 0, 0}
$\boldsymbol{R}$	1
$\boldsymbol{y}_{\text{ref}}$	$[50,\ 0,\ 0,\ 0]^\top$
$\boldsymbol{u}_{\text{ref}}$	0
T_s	80 s

The closed-loop responses are compared in Fig. 6 for MPC based on ARX (solid blue), PWARX (dashed red), and LTV (dot-dashed yellow) models. MPC with the ARX model attained the desired SOC in the shortest time (1280 s) but had the highest temperature constraint violation (305.7 K). Moreover, due to the significant mismatch between the P2D model and its ARX approximation, the closed-loop current profile exhibited undesired fluctuations, mainly between 800 and 1100 seconds. The use of a PWARX model with 4 clusters provided significant improvement both in terms of constraint satisfaction and current profile. This improved performance comes with increased complexity (2.5 s to solve each optimization compared to 0.02 s for the ARX model) but is still fast enough for online application (the sampling time is $T_s = 80$ s). Finally, the use of an LTV model leads to even better performance, with a smoother input profile when compared to the PWARX model. In fact, the better model approximation leads to an anticipated current drop, which allows the MPC algorithm to (i) avoid undesired fluctuations in the input profile, (ii) attain the reference SOC value in a shorter time (1840 s vs. 2100 s for the PWARX model), and (iii) almost not exceed the temperature constraints (303.9 K vs. 304.3 K for the PWARX model). Since the LTV-based MPC requires the solution of a QP (rather than an MIQP), its online cost (0.022 s) is comparable to the ARX case. Note also that we required the side-reaction overpotential at the end of the negative electrode to be greater than zero. By doing so, we prevent lithium plating on the particles, thus limiting the degradation of the cell. A potential limitation of the LTV-based approach is that it relies on the linearization of the 8-cluster PWARX around a nominal trajectory. Given that such trajectory was obtained offline starting from a particular initial condition, the closed-loop performance may be poor when different starting conditions are considered. For this reason, the developed LTV-based MPC was also tested for steady states with $\text{SOC}_0 = 15\%$ and $\text{SOC}_0 = 25\%$. According to Fig. 7, which compares the LTV with the 4-cluster PWARX, even in the presence of initial condition uncertainties, the LTV-based approach still provides better closed-loop performance.

Note that, even though an LTV approximation could be directly obtained starting from the P2D dynamics, we decided to use the 8-cluster PWARX model instead for the following reasons: (i) computing the optimal trajectory using the P2D model requires the solution of a nonlinear optimization problem. Due to the nature of such a problem, there is no guarantee to obtain the (global) optimum. Besides, (ii) when linearizing the P2D dynamics, one would actually have to linearize the discretized (in space) version of such a model. The use of, for example, a finite volume scheme (as in LIONSIMBA), with a reasonable number of volumes to guarantee a good approximation, would lead to a high-order LTV model. Even being linear, the

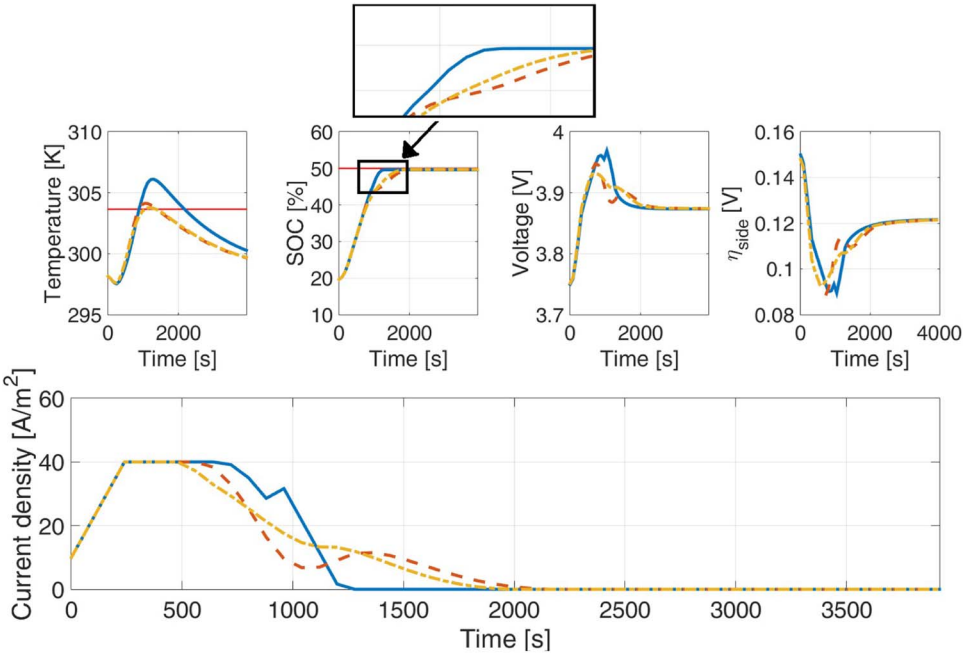

Figure 6. State profiles for three ABMSs: ARX-based (solid blue line), PWARX-based (dashed-orange line), and LTV-based (dot-dashed yellow line).

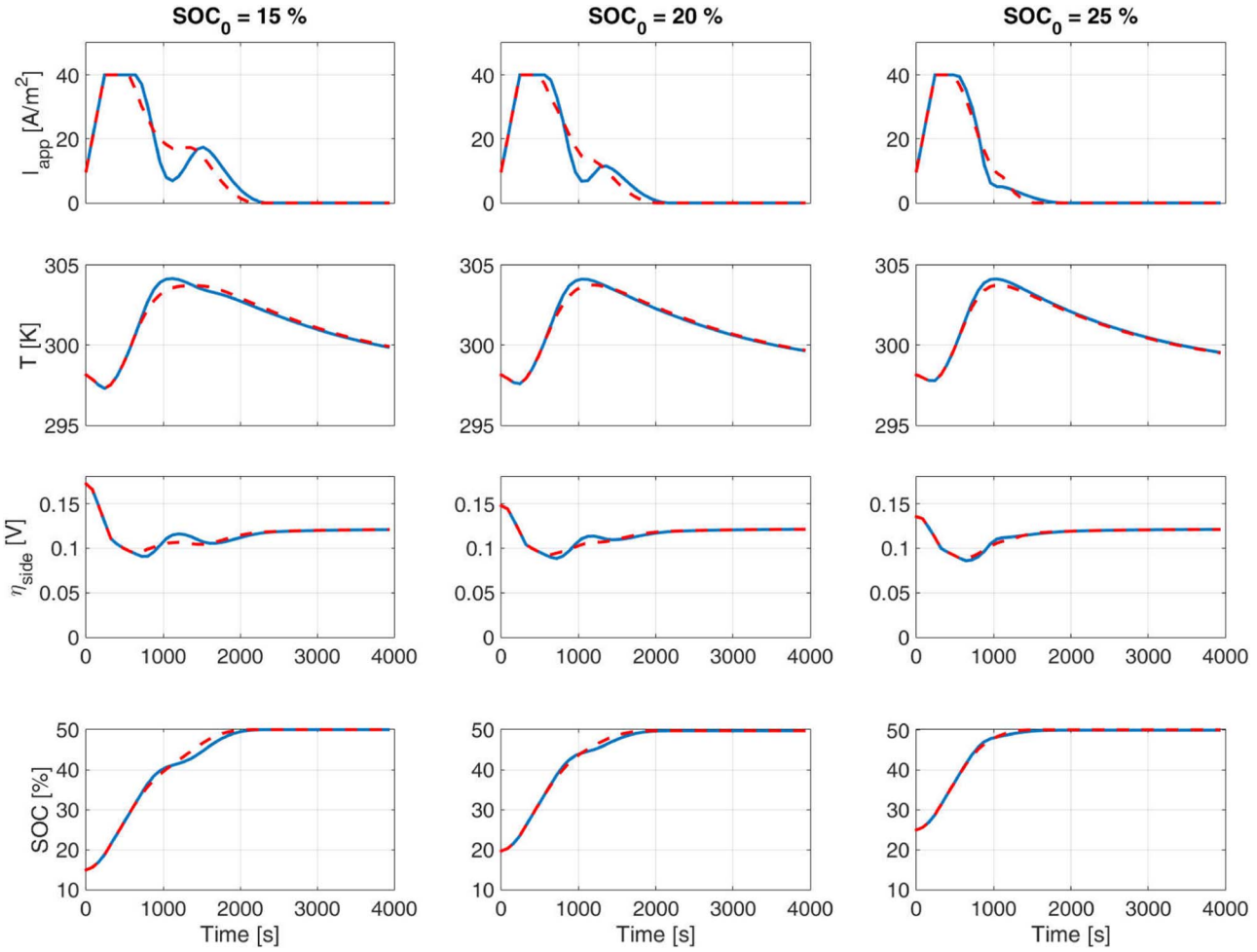

Figure 7. State profiles for PWARX (solid blue) and LTV (dashed red) models starting from different initial conditions for the SOC.

complexity of the resulting model would prevent the use of long MPC horizons, thus limiting its performance.

Conclusions

This article considers the optimal charging of a Li-ion cell using a variety of MPC strategies that were validated using the well-known P2D model based on porous electrode theory. The complexity of the P2D model, which is a set of highly nonlinear and tightly coupled PDAEs, makes its direct usage within a MPC framework impractical. For this reason, the P2D dynamics were represented by means of a linear ARX model. Although the ARX model is able to provide interesting results, the presence of strong nonlinearities for the thermal behavior called for the adoption of more sophisticated approximations. As a step toward the improvement of the closed-loop performance, PWARX models were proposed. A comparison among the prediction accuracy of the ARX, 4-cluster, and 8-cluster PWARX models highlighted the capabilities of the piecewise affine dynamics to better approximate the P2D nonlinearities and lead to improved closed-loop performance. A drawback of PWARX models is their use of binary variables, which leads to the formulation of MIQPs that need to be solved online by the MPC algorithm. Since the complexity of such problems grows exponentially with the number of clusters, their real-time application can become expensive. To reduce online computational cost, the 8-cluster PWARX model was approximated with an LTV representation obtained around a set of nominal trajectories. With respect to the ARX model, the MPC strategies based on the 4-cluster PWARX and the LTV approximations provided better closed-loop performance. Thanks to the absence of binary variables, the LTV-based MPC algorithm is formulated as a QP instead of an MIQP and so is the least expensive for online applications. Even in the presence of perturbations of the initial states, the LTV strategy provided good closed-loop performance. To further exploit the potentiality of both the P2D model and MPC algorithms, control strategies with constraints over the side reaction overpotential have been proposed. The outcomes of this strategies have further emphasized the remarkable capabilities of predictive techniques to guarantee high control performance, while ensuring the enforcement of key physical constraints.

The results highlighted the suitability of the MPC algorithms for the development of future ABMSs. By predicting the plant dynamics while enforcing operational constraints, the optimal control law provided by MPC can anticipate possible unwanted behaviors and improve the overall closed-loop performance.

References

1. S. K. Dhar, S. R. Ovshinsky, P. R. Gifford, D. A. Corrigan, M. A. Fetcenko, and S. Venkatesan, "Nickel/metal hydride technology for consumer and electric vehicle batteries – A review and update", *Journal of Power Sources*, **65**, 1 (1997).
2. D. Linden, *Handbook of Batteries and Fuel Cells* (McGraw-Hill Book Company, New York, 1984).
3. P. Ruetschi, "Review on the lead-acid battery science and technology", *Journal of Power Sources*, **2**, 3 (1977).
4. W. Zhang, "A review of the electrochemical performance of alloy anodes for lithium-ion batteries", *Journal of Power Sources*, **196**, 13 (2011).
5. X. Hu, S. Li, and H. Peng, "A comparative study of equivalent circuit models for Li-ion batteries", *Journal of Power Sources*, **198**, 359 (2012).
6. P. Van den Bossche, F. Vergels, J. Van Mierlo, J. Matheys, and W. Van Autenboer, "SUBAT: An assessment of sustainable battery technology", *Journal of Power Sources*, **162**, 913 (2006).
7. W. Shen, T. T. Vo, and A. Kapoor, "Charging algorithms of lithium-ion batteries: An overview", in *Proceedings of the IEEE Conference on Industrial Electronics and Applications* (2012) pp. 1567.
8. N. A. Chaturvedi, R. Klein, J. Christensen, J. Ahmed, and A. Kojic, "Algorithms for advanced battery-management systems", *IEEE Control Systems Magazine*, **30**, 49 (2010).
9. V. Ramadesigan, P. W. C. Northrop, S. De, S. Santhanagopalan, R. D. Braatz, and V. R. Subramanian, "Modeling and simulation of lithium-ion batteries from a systems engineering perspective", *Journal of the Electrochemical Society*, **159**, R31 (2012).
10. K. M. Tsang, L. Sun, and W. L. Chan, "Identification and modeling of lithium ion battery", *Energy Conversion and Management*, **51**, 2857 (2010).
11. V. H. Johnson, A. A. Pesaran, T. Sack, and S. America, *Temperature-dependent Battery Models for High-power Lithium-ion Batteries* (National Renewable Energy Laboratory, Golden, Colorado, 2001).
12. B. Y. Liaw, G. Nagasubramanian, R. G. Jungst, and D. H. Doughty, "Modeling of lithium ion cells – A simple equivalent-circuit model approach", *Solid State Ionics*, **175**, 835 (2004).
13. H. He, R. Xiong, and J. Fan, "Evaluation of lithium-ion battery equivalent circuit models for state of charge estimation by an experimental approach", *Energies*, **4**, 582 (2011).
14. S. K. Rahimian, S. Rayman, and R. E. White, "Comparison of single particle and equivalent circuit analog models for a lithium-ion cell", *Journal of Power Sources*, **196**, 8450 (2011).
15. K. Tsang and W. Chan, "A simple and low-cost charger for lithium-ion batteries", *Journal of Power Sources*, **191**, 633 (2009).
16. K. Tsang and W. Chan, "Current sensorless quick charger for lithium-ion batteries", *Energy Conversion and Management*, **52**, 1593 (2011).
17. J. W. Huang, Y. H. Liu, S. C. Wang, and Z. Z. Yang, "Fuzzy-control-based five-step Li-ion battery charger", in *Proceedings of the International Conference on Power Electronics and Drive Systems* (2009) pp. 1547.
18. G. C. Hsieh, L. R. Chen, and K. S. Huang, "Fuzzy-controlled Li-ion battery charge system with active state-of-charge controller", *IEEE Transactions on Industrial Electronics*, **48**, 585 (2001).
19. S. J. Moura, N. A. Chaturvedi, and M. Krstic, "Constraint management in Li-ion batteries: A modified reference governor approach", in *Proceedings of the American Control Conference* (2013) pp. 5332.
20. S. J. Moura, J. C. Forman, J. L. Stein, and H. K. Fathy, "Control of film growth in lithium ion battery packs via switches", in *Proceedings of the ASME Dynamic Systems and Control Conference*, Vol. **1** (2009) pp. 139.
21. B. Suthar, V. Ramadesigan, P. W. C. Northrop, B. Gopaluni, S. Santhanagopalan, R. D. Braatz, and V. R. Subramanian, "Optimal control and state estimation of lithium-ion batteries using reformulated models", in *Proceedings of the American Control Conference* (2013) pp. 5350.
22. M. A. Xavier and M. S. Trimboli, "Lithium-ion battery cell-level control using constrained model predictive control and equivalent circuit models", *Journal of Power Sources*, **285**, 374 (2015).
23. R. Klein, N. A. Chaturvedi, J. Christensen, J. Ahmed, R. Findeisen, and A. Kojic, "Optimal charging strategies in lithium-ion battery", in *Proceedings of the American Control Conference* (2011) pp. 382.
24. A. Mahamadi and S. Sastry, "Model predictive controller for battery management systems", in *Proceedings of the International Conference on Computing, Control, Networking, Electronics and Embedded Systems Engineering* (2015) pp. 21.
25. M. Pathak, D. Sonawane, S. Santhanagopalan, and V. R. Subramanian, "Analyzing and minimizing capacity fade through model predictive control – Theory and experimental validation", in *Proceedings of the Pacific Rim Meeting on Electrochemical and Solid-state Science* (2016) abstract 303B.
26. M. Torchio, N. A. Wolff, D. M. Raimondo, L. Magni, U. Krewer, R. B. Gopaluni, J. A. Paulson, and R. D. Braatz, "Real-time model predictive control for the optimal charging of a lithium-ion battery", in *Proceedings of the American Control Conference* (2015) pp. 4536.
27. M. Torchio, L. Magni, R. D. Braatz, and D. M. Raimondo, "Optimal health-aware charging protocol for Lithium-ion batteries: A fast model predictive control approach", in *Proceedings of the 11th International Symposium on Dynamics and Control of Process Systems, including Biosystems* (2016) pp. 827.
28. M. Torchio, L. Magni, R. D. Braatz, and D. M. Raimondo, "Optimal charging of a Li-ion cell: A hybrid model predictive control approach", in *Proceedings of the IEEE Conference on Decision and Control* (2016) pp. 4053.
29. J. M. Maciejowski, *Predictive Control: With Constraints* (Prentice Hall, Piscataway, New Jersey, 2002).
30. H. Borhan, A. Vahidi, A. M. Phillips, M. L. Kuang, I. V. Kolmanovsky, and S. D. Cairano, "MPC-based energy management of a power-split hybrid electric vehicle", *IEEE Transactions on Control Systems Technology*, **20**, 593 (2012).
31. O. Barbarisi, G. Palmieri, S. Scala, and L. Glielmo, "LTV-MPC for yaw rate control and side slip control with dynamically constrained differential braking", in *Proceedings of the European Control Conference* (2009) pp. 4810.
32. P. Falcone, M. Tufo, F. Borrelli, J. Asgari, and H. E. Tseng, "A linear time varying model predictive control approach to the integrated vehicle dynamics control problem in autonomous systems", in *Proceedings of the IEEE Conference on Decision and Control* (2007) pp. 2980.
33. P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat, "A real-time model predictive control approach for autonomous active steering", in *Proceedings of the IFAC Workshop on Nonlinear Model Predictive Control for Fast Systems* (Grenoble, France, 2006).
34. C. Y. Wang, W. B. Gu, and B. Y. Liaw, "Micro-macroscopic coupled modeling of batteries and fuel cells I. Model development", *Journal of the Electrochemical Society*, **145**, 3407 (1998).
35. K. Kumaresan, G. Sikha, and R. E. White, "Thermal model for a Li-ion cell", *Journal of the Electrochemical Society*, **155**, A164 (2008).
36. P. Ramadass, B. Haran, P. M. Gomadam, R. White, and B. N. Popov, "Development of First Principles Capacity Fade Model for Li-Ion Cells", *Journal of The Electrochemical Society*, **151**, A196 (2004).
37. M. Rashid and A. Gupta, "Mathematical model for combined effect of SEI formation and gas evolution in Li-ion batteries", *ECS Electrochemistry Letters*, **3**, A95 (2014).
38. C. M. Doyle, *Design and Simulation of Lithium Rechargeable Batteries*, Ph.D. thesis, University of California Berkeley (1995).

39. M. Torchio, L. Magni, R. B. Gopaluni, R. D. Braatz, and D. M. Raimondo, "LIONSIMBA: A Matlab framework based on a finite volume model suitable for Li-ion battery design, simulation, and control", *Journal of The Electrochemical Society*, **163**, A1192 (2016).
40. D. Mayne, "Nonlinear model predictive control: Challenges and opportunities", in *Nonlinear Model Predictive Control*, edited by L. Magni, D. M. Raimondo, and F. Allgöwer, (Springer, Berlin Heidelberg, 2000) pp. 23.
41. L. Magni, D. M. Raimondo, and F. Allgöwer, eds., *Nonlinear Model Predictive Control* (Springer, Berlin Heidelberg, 2009).
42. P. V. Kokotovic, R. O'Malley, and P. Sannuti, "Singular perturbations and order reduction in control theory – An overview", *Automatica*, **12**, 123 (1976).
43. A. Mesbah, J. A. Paulson, R. Lakerveld, and R. D. Braatz, "Plant-wide model predictive control for a continuous pharmaceutical process", in *Proceedings of the American Control Conference* (2015) pp. 4301.
44. M. Geuss, B. Lohmann, B. Peherstorfer, and K. Willcox, "A black-box method for parametric model order reduction", *IFAC-PapersOnLine*, **48**, 168 (2015).
45. P. Saraswat and G. Parmar, "A comparative study of differential evolution and simulated annealing for order reduction of large scale systems", in *Proceedings of the International Conference on Communication Control and Intelligent Systems* (2015) pp. 277.
46. F. Kuhne, W. F. Lages, and J. M. G. da Silva Jr., "Model predictive control of a mobile robot using linearization", in *Proceedings of the International Conference on Mechatronics and Robotics* (Aachen, Germany, 2004) pp. 525.
47. D. Simon, J. Löfberg, and T. Glad, "Nonlinear model predictive control using feedback linearization and local inner convex constraint approximations", in *Proceedings of the European Control Conference* (2013) pp. 2056.
48. J. Deng, V. Becerra, and R. Stobart, "Input constraints handling in an MPC/feedback linearization scheme", *International Journal of Applied Mathematics and Computer Science*, **19**, 219 (2009).
49. L. Ljung, "System identification", in *Signal Analysis and Prediction*, edited by A. Prochazka, N. Kingsbury, P. J. W. Payner, and J. Uhlir, (Birkhäuser, Basel, Switzerland, 1998) pp. 163.
50. G. Ferrari-Trecate, M. Muselli, D. Liberati, and M. Morari, "A clustering technique for the identification of piecewise affine systems", *Automatica*, **39**, 205 (2003).
51. T. Matsuo and Y. Hasegawa, *Realization Theory of Discrete-time Dynamical Systems* (Springer Verlag, Berlin Heidelberg, 2003).
52. A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints", *Automatica*, **35**, 407 (1999).
53. H. P. Williams, *Model Building in Mathematical Programming* (John Wiley & Sons, New York, 2013).
54. Z. Lassoued and K. Abderrahim, "A Kohonen neural network based method for PWARX identification", in *Adaptation and Learning in Control and Signal Processing*, Vol. **11**(1), (2013) pp. 742.
55. R. Vidal, S. Soatto, Y. Ma, and S. Sastry, "An algebraic geometric approach to the identification of a class of linear hybrid systems", in *Proceedings of the IEEE Conference on Decision and Control*, Vol. **1** (2003) pp. 167.
56. A. L. Juloski, W. Heemels, G. Ferrari-Trecate, R. Vidal, S. Paoletti, and J. Niessen, "Comparison of four procedures for the identification of hybrid systems", in *Hybrid Systems: Computation and Control*, edited by M. Morari and L. Thiele, (Springer Verlag, Berlin Heidelberg, 2005) pp. 354.
57. G. Ferrari-Trecate, "Hybrid Identification Toolbox (HIT)", (2005).
58. J. Stevek and S. Kozak, "Matlab toolbox for PWA identification of nonlinear systems", in *Proceedings of the International Conference on Process Control* (2011) pp. 111.
59. S. MATLAB, *version 7.10.0 (R2010a)* (The MathWorks Inc., Natick, Massachusetts, 2010).
60. M. Herceg, M. Kvasnica, C. Jones, and M. Morari, "Multi-Parametric Toolbox 3.0", in *Proceedings of the European Control Conference* (2013) pp. 502.
61. J. Schubert, R. Simutis, M. Dors, I. Havlik, and A. Lubbert, "Hybrid modeling of yeast production processes – Combination of a-priori knowledge on different levels of sophistication", *Chem. Eng. Technol.*, **17**, 10 (1994).
62. B. Srinivasan, S. Palanki, and D. Bonvin, "Dynamic optimization of batch processes: I. Characterization of the nominal solution", *Computers & Chemical Engineering*, **27**, 1 (2003).
63. J. V. Breakwell, "The optimization of trajectories", *Journal of the Society for Industrial and Applied Mathematics*, **7**, 215 (1959).
64. S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology", *Control Engineering Practice*, **11**, 733 (2003).
65. E. F. Camacho and C. Bordons, *Model Predictive Control in the Process Industry* (Springer Verlag, London, 2012).
66. W. H. Kwon and S. H. Han, *Receding horizon control: Model predictive control for state models* (Springer-Verlag, London, 2005).
67. J. Mattingley, Y. Wang, and S. Boyd, "Receding horizon control", *IEEE Control Systems*, **31**, 52 (2011).
68. D. Q. Mayne and H. Michalska, "Receding horizon control of nonlinear systems", *IEEE Transactions on Automatic Control*, **35**, 814 (1990).
69. "IBM ILOG CPLEX Optimizer", http://www-01.ibm.com/software/integration/optimization/cplex-optimizer (2010).
70. A. C. Hindmarsh, P. N. Brown, K. E. Grant, S. L. Lee, R. Serban, D. E. Shumaker, and C. S. Woodward, "SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers", *ACM Transactions on Mathematical Software (TOMS)*, **31**, 363 (2005).