CHALMERS
UNIVERSITY OF TECHNOLOGY

Electrical vehicles charging: an optimal control approach via Pontryagin's Maximum Principle

Downloaded from: https://research.chalmers.se, 2026-09-15 10:02 UTC

Citation for the original published paper (version of record):
Montalto, L., Murgovski, N., Fredriksson, J. (2025). Electrical vehicles charging: an optimal control approach via Pontryagin's Maximum Principle. IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC. http://dx.doi.org/10.1109/ITSC60802.2025.11423091

N.B. When citing this work, cite the original published paper.

© 2025 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, or reuse of any copyrighted component of this work in other works.

(article starts on next page)

Electrical vehicles charging: an optimal control approach via Pontryagin's Maximum Principle

Lorenzo Montalto, Nikolce Murgovski and Jonas Fredriksson

***Abstract*—This study provides a semi-analytical solution to the charging optimization problem for electric vehicles, using Pontryagin's Maximum Principle (PMP) to derive explicit expressions for optimal control and costate trajectories. Such expressions enable much faster trajectory computation than numerical optimization, making this method suitable for larger trip-planning problems. In addition, these explicit expressions reveal the fundamental structure of the optimal solution and how all the parameters and variables affect it, leading to more interpretable results and deeper theoretical insights than fully numerical or black-box methods allow. The method nearly eliminates online computation, while having an average difference with a benchmark solution of about 2.74%.**

***Index Terms*—optimal control, PMP, optimization, electromobility**

I. Introduction

The legislation on CO2 emissions in Europe is becoming increasingly stringent (although arguably not enough [1]) to reach climate neutrality by 2050 [2]. The transport sector is among the most critical ones when it comes to CO2 emissions by sector [3], which means that making it greener quickly is essential. One way of doing so is by incentivizing the use of battery electric vehicles (BEVs) to replace internal combustion engine (ICE) vehicles.

Promoting BEV adoption requires addressing consumers' concerns. One such concern is range anxiety [4], that is, the fear of the battery running out before reaching a destination. A hardware-side solution could be increasing battery size. In fact, due to its impact on production cost and weight, optimizing the battery size is a vital problem being explored in the literature [5] [6].

A software-side solution to range anxiety is intelligent charge planning, i.e., choosing charging stops to ensure the vehicle reaches its destination. In particular, the goal of the charge planning problem is to optimally reach a desired level of state of charge (SoC) while minimizing objectives such as energy use, charging time, and cost. Optimal methods for charging of EVs have been widely discussed in the literature (see [7] for an overview).

An important question in optimization is whether the global optimum can be attained, and if yes, how efficiently. If the optimization problem is convex, it can indeed be ensured, under certain optimality conditions, that global optimum can be attained. Efficient, off-the-shelf solvers also exist for convex programs. However, optimization problems are not always convex, especially when dealing with highly nonlinear models. The global optimum of non-convex problems can generally not be attained, except for special cases (see [8], where a non-convex EV charging problem is solved via its convex dual).

A method that can handle non-convex and mixed-integer problems is Dynamic Programming (DP). This method is suitable if the problem has optimal substructure, that is, if it can be broken into subproblems whose solutions combine into a global one. With DP, subproblem solutions are stored in memory for convenient reuse, as in [9]. However, DP quickly becomes intractable as problem size increases.

To avoid the computational complexity of DP, without requiring convexity, a classic optimal control strategy is Model Predictive Control (MPC), as in [10]. MPC computes control inputs that optimize performance while satisfying constraints, which is what makes MPC a widely used control algorithm. As the name suggests, MPC relies on a model of the system to predict the future and plan the control accordingly. A drawback of MPC is the need to solve an optimization problem at each sampling step, potentially making it computationally heavy.

Another powerful optimal control technique is Pontryagin's Maximum Principle (PMP) [11], which is widely regarded as a core result in optimal control theory, as it provides necessary conditions for a trajectory to be optimal. PMP-based methods allow the retrieval of an analytical or semi-analytical solution to an optimization problem. This is of theoretical interest, since explicit expressions for the optimal control inputs would reveal the underlying structure of the optimal solution, but also has practical applications, since the retrieval of optimal solutions through explicit expressions is usually much faster compared to solving an optimization problem. While harder to implement than MPC, their main computational burden is handled only once, offline. This is why these methods have the potential of being very computationally effective: once the expressions for the optimal control inputs have been obtained, retrieving an optimal solution from them is quite fast. In [12], a PMP-based method is used to solve the thermal and energy battery management optimization problem in EV, with emphasis on the life span

All authors are with the Department of Electrical Engineering, Chalmers University of Technology, Gothenburg 412 96, Sweden (e-mail: lorenzo.montalto@chalmers.se). This article is within the SEC project CHARGE: Charging and Trip Planning of Electric Vehicles, project number 13048. The authors would also like to thank Viktor Larsson (Volvo Cars), and Niklas Legnedahl (Zeekrtech) for our fruitful discussions.

of the battery. Here, a fully analytical solution is provided for a very simple case, while a more general case is solved numerically. In [13], a PMP-based strategy is used to solve the energy management problem for plug-in-hybrid EV. In particular, the authors show how the PMP can be used when having mixed-integer control inputs and how to address singular controls. A pedagogical description on how to deal with singular controls and singular arcs can be found in [14].

The novelty introduced by this work is a method to solve the EV charging and battery thermal management (BTM) problem through the computationally efficient solution of a two-point boundary value problem (2PBVP) formulated through PMP. First, unbounded control laws are derived by minimizing the Hamiltonian of the problem, with nonlinearities ensuring no singular arcs. Then, the 2PBVP is formulated and solved numerically using single-shooting techniques to retrieve the initial conditions and the charging time needed to simulate charging. When simulating the charging process, control inputs are computed from unbounded expressions or saturated at one of the bounds, whichever minimizes the Hamiltonian while respecting constraints. This method:

- produces comparable results to solving a standard optimization problem with an off-the-shelf solver using multiple shooting, having an average difference with the benchmark results of about 2.74%;
- effectively eliminates the online computation time (99.87% reduction) to obtain the optimal solution;
- produces expressions for the unbounded optimal control inputs, revealing their structure and how the variables and the parameters affect the optimal solution.

These reasons make this method interesting from a theoretical standpoint, as it allows the analytical inspection of the optimal solution, and from a practical one, since it significantly improves the computational viability of optimal methods to solve the charging and BTM problem.

The article is structured as follows. Section II formulates the system model (electrical and thermal domains) and the charging optimization problem. In section III, the PMP and the optimality conditions are introduced and the expressions for the unbounded optimal control inputs are derived. In IV, the 2PBVP and the Newton-based algorithm used to solve it numerically are defined. Section V presents the results of the PMP-based method and compares them to benchmark results. Finally, conclusions and future works are discussed in Section VI.

II. Problem formulation

A. States and control inputs

The vehicle is modeled as a nonlinear system

$$\dot{\mathbf{x}}(t) = f(\mathbf{x},\mathbf{u}) = \begin{bmatrix} f_{T_\mathrm{b}}(\mathbf{x},\mathbf{u}) & f_{\mathrm{SoC}}(\mathbf{x},\mathbf{u}) \end{bmatrix}^\intercal \tag{1}$$

with two states, vector $\mathbf{x}$, and three control inputs, vector $\mathbf{u}$

$$\mathbf{x}(t) = \begin{bmatrix} T_\mathrm{b}(t) & \mathrm{SoC}(t) \end{bmatrix}^\intercal \tag{2}$$

$$\mathbf{u}(t) = \begin{bmatrix} P_\mathrm{b}(t) & P_\mathrm{hvch}(t) & P_\mathrm{hp}(t) \end{bmatrix}^\intercal \tag{3}$$

where T_b and SoC are battery temperature and state of charge, respectively, P_b is the battery internal chemical power, P_hvch is the High Voltage Coolant Heater (HVCH) power, and P_hp is the Heat Pump (HP) power. Throughout the article, nested parenthesis have been omitted for increased readability (i.e. $f(x(t),u(t))$ has been written as $f(x,u)$).

The system is modeled as in [15], as described in the following subsections. The model and the parameters used refer to an average personal electric vehicle.

B. Electrical model of the system

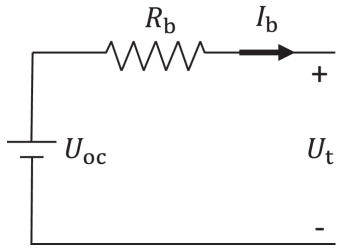


Fig. 1. Equivalent circuit of the battery pack

The battery pack is modeled as shown in Fig. 1, where U_oc is the open circuit voltage and R_b the battery's internal resistance,

$$R_\mathrm{b}(T_\mathrm{b}) = \begin{bmatrix} T_\mathrm{b}(t)^2 & T_\mathrm{b}(t) & 1 \end{bmatrix}^\intercal \cdot \mathbf{r}_\mathrm{b} \tag{4}$$

$$U_\mathrm{oc}(\mathrm{SoC}) = \begin{bmatrix} \mathrm{SoC}^2(t) & \mathrm{SoC}(t) & 1 \end{bmatrix}^\intercal \cdot \mathbf{u}_\mathrm{oc} \tag{5}$$

where $\mathbf{r}_\mathrm{b} \in \mathbb{R}^{3\times1}$ and $\mathbf{u}_\mathrm{oc} \in \mathbb{R}^{3\times1}$ are coefficients determined by fitting existing data. The battery resistance R_b has been modeled so that it decreases as T_b increases. Conversely, the open circuit voltage U_oc has been modeled so that it increases as SoC increases.

The state of charge is defined as the ratio between the available energy in the battery pack and its nominal total energy. Its dynamics are described as

$$f_\mathrm{SoC}(\mathbf{x},\mathbf{u}) = -\frac{P_\mathrm{b}(t)}{C_\mathrm{b} U_\mathrm{oc}(\mathrm{SoC})} \tag{6}$$

which means that a negative P_b charges the battery.

C. Thermal model of the system

The heat generated by Joule losses is modeled as

$$Q_\mathrm{joule}(\mathbf{x},\mathbf{u}) = \left(\frac{P_\mathrm{b}(t)}{U_\mathrm{oc}(\mathrm{SoC})}\right)^2 \cdot R_\mathrm{b}(T_\mathrm{b}), \tag{7}$$

while the heat exchanged between the battery and the environment is defined as

$$Q_\mathrm{amb}(T_\mathrm{b}) = (T_\mathrm{b}(t) - T_\mathrm{amb}) \cdot \gamma_0 \tag{8}$$

where T_{amb} is the ambient temperature and γ_0 is the coefficient of heat transfer between battery and ambient.

The battery can be heated up via the HVCH, which generates heat according to the following model

$$Q_{\text{hvch}}(P^{\text{b}}_{\text{hvch}}) = \eta_{\text{hvch}} \cdot P_{\text{hvch}}(t) - \alpha_{\text{hvch}} \cdot P^2_{\text{hvch}}(t) \quad (9)$$

where η_{hvch} is HVCH efficiency, α_{hvch} is a coefficient which captures the quadratic dependence in P_{hvch}.

The vehicle is equipped with a heat pump (HP) to channel heat from the battery to the cabin via

$$\begin{aligned} f_{\text{hp}}\left(T_{\text{b}}, P_{\text{hp}}\right) = &\left(p_{\text{Qhp},1} T_{\text{b}}(t) + p_{\text{Qhp},0}\right) P_{\text{hp}}(t) \\ &- \alpha_{\text{hp}} P^2_{\text{hp}}(t) \end{aligned} \quad (10)$$

where $p_{\text{Qhp},1}$, $p_{\text{Qhp},0}$ and α_{hp} are coefficients obtained by fitting data. To prevent heat transfer when the battery temperature is too low, while still avoiding points of non-differentiability, the heat transfer can be modeled as a smoothed rectified linear unit (ReLU). To that goal, we first define the smoothed step function

$$\sigma_{\text{hp}}(Z) = \frac{1}{2}\left(\frac{Z}{\sqrt{Z^2 + \epsilon^2_\sigma}} + 1\right) \quad (11)$$

which tends to 1 if $Z \geq 0$ and to 0 otherwise. The transition between 0 and 1, when $Z \approx 0$, becomes steeper the smaller ϵ_σ is. The argument Z could be any real-valued expression. The heat transferred by the HP from the battery to the cabin can then be modeled as

$$Q_{\text{hp}}(T_{\text{b}}, P_{\text{hp}}) = f_{\text{hp}}\left(T_{\text{b}}, P_{\text{hp}}\right) \cdot \sigma(T_{\text{b}}(t) - T^{\text{thres}}_{\text{b}}) \quad (12)$$

where $T^{\text{thres}}_{\text{b}}$ is the minimum battery temperature needed to be able to use the heat pump. This way, $Q_{\text{hp}} \approx 0$ if $T_{\text{b}} < T^{\text{thres}}_{\text{b}}$.

The cabin can also be heated up via the HVCH. The HVCH power to heat up the cabin is given by the total power demand for cabin heating, minus the heat channeled to the cabin by the HP

$$P^{\text{c}}_{\text{hvch}}(\cdot) = \frac{P_{\text{c}}(T_{\text{amb}}) - Q_{\text{hp}}(T_{\text{b}}, P_{\text{hp}})}{\eta_{\text{Qhvch}}} \quad (13)$$

where P_{c} is the power demand for cabin heating, and η_{Qhvch} is the electrical-to-heat efficiency of the HVCH for cabin heating.

The battery has been modeled as a lumped mass with only the battery temperature as a thermal state, which evolves according to

$$\begin{aligned} f_{T_{\text{b}}}(\mathbf{x}, \mathbf{u}) = \frac{1}{c_{\text{b}} m_{\text{b}}}\big(&Q_{\text{joule}}(\mathbf{x}, \mathbf{u}) + Q_{\text{hvch}}(P_{\text{hvch}}) \\ &- Q_{\text{amb}}(T_{\text{b}}) - Q_{\text{hp}}(T_{\text{b}}, P_{\text{hp}})\big) \end{aligned} \quad (14)$$

where c_{b} is the heat capacity of the battery cells, m_{b} is the mass of the battery pack, and C_{b} is the battery capacity.

D. Bounds on the battery power and energy balance

The battery power P_{b} is the control input responsible for charging. Its lower bound is a function of T_{b} and SoC, defined as

$$\begin{aligned} P_{\text{b,min}}(\mathbf{x}) = &\left(\begin{bmatrix} T_{\text{b}}(t)^2 & T_{\text{b}}(t) & 1 \end{bmatrix}^\intercal \cdot \mathbf{p}_{T_{\text{b}}}\right) \cdot \\ &\left(\begin{bmatrix} \text{SoC}(t)^2 & \text{SoC}(t) & 1 \end{bmatrix}^\intercal \cdot \mathbf{p}_{\text{SoC}}\right) \end{aligned} \quad (15)$$

where $\mathbf{p}_{T_{\text{b}}} \in \mathbb{R}^3$ and $\mathbf{p}_{\text{SoC}} \in \mathbb{R}^3$ are vectors of coefficients obtained by fitting data. The lower bound decreases as T_{b} increases and as SoC decreases.

Since the battery power is never positive during charging (as, according to (6), that would mean the battery is discharging), we get

$$P_{\text{b,min}}(\mathbf{x}) \leq P_{\text{b}}(t) \leq 0 \quad (16)$$

As the vehicle is charging, we need to ensure that the power taken from the grid, P_{grid}, does not exceed a certain limit. This grid power is defined as the difference between the consumed power and P_{b} (which represents the input power, since it is the control input responsible for charging)

$$\begin{aligned} P_{\text{grid}}(\cdot) = &\frac{R_{\text{b}}(T_{\text{b}})}{U^2_{\text{oc}}(\text{SoC})} P^2_{\text{b}}(t) + P^{\text{c}}_{\text{hvch}}(\cdot) + P_{\text{c}}(T_{\text{amb}}) \\ &+ P_{\text{hp}}(t) + P_{\text{hvch}}(t) + P_{\text{aux}} - P_{\text{b}}(t). \end{aligned} \quad (17)$$

where $P^{\text{max}}_{\text{grid}}$ is a known limit on the charging power. To ensure that this power does not exceed a certain limit, the following constraint needs to be imposed

$$0 \leq P_{\text{grid}}(\cdot) \leq P^{\text{max}}_{\text{grid}}. \quad (18)$$

E. Charging optimization problem

The goal of the charging optimization problem is to attain a desired level of SoC while minimizing the energy taken from the grid, i.e. the stage cost S, and the charging time, i.e. the terminal cost V

$$\min_{\mathbf{x},\mathbf{u},t_{\text{f}}} \; J(\mathbf{x}, \mathbf{u}, t_{\text{f}}) = \underbrace{w_{\text{t}} t_{\text{f}}}_{V(t_{\text{f}})} + \int_0^{t_{\text{f}}} \underbrace{w_{\text{e}} P_{\text{grid}}(\cdot)}_{S(\cdot)}\, dt \quad (19a)$$

$$\text{s.t.: } \mathbf{x}(0) = \mathbf{x}_0,\ \text{SoC}(t_{\text{f}}) = \text{SoC}_{\text{des}} \quad (19b)$$

$$(6),\ (14), (16), (18) \quad (19c)$$

$$x_i \in [x_i^{\text{min}}, x_i^{\text{max}}],\ i = 1, 2 \quad (19d)$$

$$u_i \in [u_i^{\text{min}}, u_i^{\text{max}}],\ i = 1, 2, 3 \quad (19e)$$

$$t_{\text{f}} \in [0, t_{\text{f}}^{\text{max}}] \quad (19f)$$

where w_{t} and w_{e}, which can be selected based on user preferences, model the trade-off between minimizing charging time and minimizing energy bought from the grid, t_{f} is the final time, $\mathbf{x}_0 \in \mathbb{R}^2$ are known initial conditions for the states, SoC_{des} is the desired SoC level to be reached at the end of charging, x_i^{min}, x_i^{max}, u_i^{min} and u_i^{max} are known box constraints on the states and on the control inputs, respectively. Notice that t_{f} and $T_{\text{b}}(t_{\text{f}})$ are free, while $\text{SoC}(t_{\text{f}})$ is fixed.

The trajectories obtained by solving this problem numerically via multiple shooting will be used as benchmark and compared with those described in sections III and IV.

III. Pontryagin's Maximum Principle (PMP)

We define two costates (one for each state)

$$\boldsymbol{\lambda}(t) = \begin{bmatrix} \lambda_{T_\mathrm{b}}(t) & \lambda_{\mathrm{SoC}}(t) \end{bmatrix}^\intercal \tag{20}$$

and the Hamiltonian of the problem in (19), defined according to the theory of the PMP [16], is

$$H(\cdot) = S(\cdot) + \lambda_{T_\mathrm{b}}(t) f_{T_\mathrm{b}}(\mathbf{x}, \mathbf{u}) + \lambda_{\mathrm{SoC}}(t) f_{\mathrm{SoC}}(\mathbf{x}, \mathbf{u}). \tag{21}$$

The optimality conditions can now be stated as

$$H(\mathbf{x}^*, \mathbf{u}^*, \boldsymbol{\lambda}^*, t) \leq H(\mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}, t),\ \forall t \in [t_0, t_f], \tag{22a}$$

$$\frac{d\boldsymbol{\lambda}(t)}{dt} = -\nabla_x H(\cdot), \tag{22b}$$

$$\lambda_{T_\mathrm{b}}(t_\mathrm{f}) - \frac{\partial V^*(t_\mathrm{f})}{\partial x} = 0, \tag{22c}$$

$$H^*(t_\mathrm{f}) + \frac{\partial V^*(t_\mathrm{f})}{\partial t_\mathrm{f}} = 0. \tag{22d}$$

where (22a) means that solving the optimization problem in (19) is the same as minimizing H (here, $*$ indicates an optimal solution), while (22b) provides an expression for the costates' dynamics. Conditions (22c) and (22d) stem from the fact that the optimal control problem has a free final state, T_b, and free final time, t_f, respectively.

When control constraints are not active, the optimal trajectory for the P_b can be found as the trajectory that fulfills $\frac{\partial H}{\partial P_\mathrm{b}} = 0$, since the goal is to minimize H. Solving the equation for P_b, we get the optimal unconstrained battery power

$$P^*_{\mathrm{b,uc}}(\mathbf{x}, \boldsymbol{\lambda}) = \frac{c_\mathrm{b} m_\mathrm{b} \lambda_{\mathrm{SoC}}(t) U_{\mathrm{oc}}(\mathrm{SoC})}{2C_\mathrm{b}\left(\lambda_{T_\mathrm{b}}(t) + c_\mathrm{b} m_\mathrm{b} w_\mathrm{e}\right) R_\mathrm{b}(T_\mathrm{b})}. \tag{23}$$

We can see how this expression is directly proportional to U_{SoC}. Since $P_\mathrm{b} < 0$ while charging, the charging power decreases in magnitude (i.e., P_b becomes "less negative") as SoC increases, which is reasonable. We can also notice that w_e is in the denominator, meaning that a higher cost for electricity from the grid would, reasonably, result in a lower charging power and longer charging time, which generally reduces charging losses.

Analogously, the optimal trajectory for P_{hvch} can be found as the trajectory that fulfills $\frac{\partial H}{\partial P_{\mathrm{hvch}}} = 0$, which can be solved for P_{hvch}, since (9) is quadratic in P_{hvch}, obtaining the optimal unconstrained HVCH power

$$P^*_{\mathrm{hvch,uc}}(\lambda_{T_\mathrm{b}}) = \frac{\eta_{\mathrm{hvch}} \lambda_{T_\mathrm{b}}(t) + c_\mathrm{b} m_\mathrm{b} w_\mathrm{e}}{2\alpha_{\mathrm{hvch}} \lambda_{T_\mathrm{b}}(t)}. \tag{24}$$

Finally, the procedure to obtain an expression for the HP power is analogous to the ones for the battery power and the HVCH power, since P_hp enters H quadratically, thanks to Q_hp being quadratic in P_hp. However, this is not enough, as the Hamiltonian is not necessarily convex in P_hp and obtaining a stationary point $\frac{\partial H}{\partial P_{hp}} = 0$ does not necessarily lead to a minimum of H. Instead, we are interested in a stationary point with a nonnegative local curvature, enforced by

$$\begin{cases} \dfrac{\partial H}{\partial P_{hp}} = 0 \\ \dfrac{\partial^2 H}{\partial P_{hp}^2} \geq 0. \end{cases} \tag{25}$$

Solving (25) for P_hp, we obtain an expression for the optimal unbounded HP power

$$\begin{aligned} P^*_{\mathrm{hp,uc}}(\cdot) = {} & \frac{1}{2\alpha_\mathrm{hp}\sigma_\mathrm{hp}(T_\mathrm{b} - T_\mathrm{b}^\mathrm{thres})\left(\frac{w_\mathrm{e}}{\eta_\mathrm{Qhvch}} + \frac{\lambda_{T_\mathrm{b}}(t)}{c_\mathrm{b}\, m_\mathrm{b}}\right)} \cdot \\ & \cdot \left[w_\mathrm{e} \left(\frac{\sigma_\mathrm{hp}(T_\mathrm{b})\left(Q_{\mathrm{hp,P0}} + Q_{\mathrm{hp,P1}}\, T_\mathrm{b}(t)\right)}{\eta_\mathrm{Qhvch}} - 1 \right) \right. \\ & \left. + \frac{\lambda_{T_\mathrm{b}}(t)\, \sigma_\mathrm{hp}(T_\mathrm{b} - T_\mathrm{b}^\mathrm{thres})\left(Q_{\mathrm{hp,P0}} + Q_{\mathrm{hp,P1}}\, T_\mathrm{b}(t)\right)}{c_\mathrm{b}\, m_\mathrm{b}} \right]. \end{aligned} \tag{26}$$

These expressions can be derived offline either by hand or with symbolic toolboxes. The part that is to be solved online, and that therefore needs to be computationally feasible, is described in the next section.

IV. Two-point boundary value problem (2PBVP)

The expressions (23), (24) and (26) for the unconstrained optimal control inputs depend on the co-states, whose initial values we do not have. We also need the charging time to simulate the system. Therefore, to be able to use these expressions and simulate charging, we need to find the following unknowns

$$\boldsymbol{\omega} = \begin{bmatrix} \lambda_{T_\mathrm{b}}(0) & \lambda_{\mathrm{SoC}}(0) & t_\mathrm{f} \end{bmatrix}^\intercal. \tag{27}$$

Let $\tau \in [0, 1]$ represent a normalized time, such that $t = \tau t_\mathrm{f}$. Then, by collecting the final conditions within the vector of residuals

$$r(\boldsymbol{\omega}) = \begin{bmatrix} \mathrm{SoC}(1) - \mathrm{SoC}_\mathrm{des} \\ \frac{\partial V^*(t_\mathrm{f})}{\partial T_\mathrm{b}} - \lambda_{T_\mathrm{b}}(1) \\ H^*(t_\mathrm{f}) + \frac{\partial V^*(t_\mathrm{f})}{\partial t_\mathrm{f}} \end{bmatrix}$$

the unknowns can be obtained by solving the two-point boundary value problem

$$\min_{\boldsymbol{\omega}} \ \|r(\boldsymbol{\omega})\| \tag{28a}$$

$$\text{s.t.: } \frac{\mathbf{x}(\tau)}{d\tau} = t_\mathrm{f} f(x, u^*), \tag{28b}$$

$$\frac{\boldsymbol{\lambda}(\tau)}{d\tau} = -t_\mathrm{f} \nabla_x H(\mathbf{x}, \mathbf{u}^*, \boldsymbol{\lambda}, \tau), \tag{28c}$$

$$\mathbf{x}(0) = \mathbf{x}_0 \tag{28d}$$

where the cost function is typically the first, or second norm of the residual. The problem is then solved numerically through a quasi-Newton method (since an approximate Jacobian is computed instead of an exact one), according to Alg. 1, starting from an initial guess and correcting it until convergence. Informally, the idea behind Alg. 1 goes as follows

1) guess an initial value for $\boldsymbol{\omega}_\mathrm{g}$ using a neural network,
2) simulate charging, starting from $\boldsymbol{\omega}_\mathrm{g}$, and check if $||\mathbf{r}(\boldsymbol{\omega}_\mathrm{g})|| < \delta_\mathrm{tol}$
3) if the condition is not satisfied, correct the guess $\boldsymbol{\omega}_\mathrm{g}$ (i.e. perform a quasi-Newton step),
4) go to step 2

where $\delta_\mathrm{tol} > 0$ is a small value. The quasi-Newton step uses a numerically computed Jacobian $J_{r,\omega}$. The neural network used to generate an initial guess $\boldsymbol{\omega}_\mathrm{g}$ takes as input the ambient temperature T_amb, the initial state of charge $\mathrm{SoC}(0)$, the desired final state of charge $\mathrm{SoC}(s_f)$ and the cost parameter for charging time w_t (w_e is then set as $1 - w_\mathrm{t}$) and outputs an initial guess for $\lambda_{T_\mathrm{b}}(0)$, $\lambda_\mathrm{SoC}(0)$ and t_f.

Due to the high nonlinearity of the problem and the use of a quasi-Newtod method, the algorithm's convergence depends on the quality of the initial guess. Therefore, no formal proof of convergence can be obtained.

During charging simulation, control inputs are selected to minimize H (see Alg. 2). Three options are considered for P_hvch and P_hp: unconstrained optimum ($P^*_\mathrm{hvch,uc}$ and $P^*_\mathrm{hp,uc}$, respectively), and their lower and upper bounds (in case the unconstrained optimum are unfeasible). For P_b, only the unconstrained optimum $P^*_\mathrm{b,uc}$ and the lower bound $P_\mathrm{b,min}$ are considered, since the upper bound would be either 0 or positive, and both should not happen during charging. To do that, multiple versions of H are derived, depending on which configuration of the control inputs is considered. This becomes clearer when expressing H as a function of the control inputs: $H(P_\mathrm{b}, P_\mathrm{hvch}, P_\mathrm{hp})$. For instance, one optimal setting, at a certain time instant, might be P_b at its lower bound, P_hvch at its unconstrained optimum in (24) and no P_hp. In that case, we will consider the Hamiltonian that corresponds to $H(P_\mathrm{b,min}, P^*_\mathrm{hvch,uc}, 0)$. Since we have three options for P_hvch and P_hp and two for P_b, we get $3{\cdot}3{\cdot}2 = 18$ different Hamiltonians. At every time instant, all 18 are evaluated and the configuration with the lowest H is selected.

Algorithm 1: Solve 2PBVP through quasi-Newton method

Data: $\mathbf{x}_0$, $\boldsymbol{\omega}_\mathrm{g}$, $\mathrm{SoC}_\mathrm{des}$, δ_tol, N_max
Result: $\boldsymbol{\omega}$, $\mathbf{x}$, $\boldsymbol{\lambda}$

$\mathbf{r}_\mathrm{min} = \boldsymbol{\infty}$;
for $i \leftarrow 0$ **to** N_max **do**
  Simulate charging with Alg. 2 and compute residuals $\mathbf{r}(\boldsymbol{\omega}_\mathrm{g})$;
  if $||\mathbf{r}(\boldsymbol{\omega}_\mathrm{g})||_2 < \delta_\mathrm{tol}$ **then**
    Break and return current $\boldsymbol{\omega}_\mathrm{g}$, $\mathbf{x}$ and $\boldsymbol{\lambda}$;
  else
    if $||\mathbf{r}(\boldsymbol{\omega}_\mathrm{g})||_2 < ||\mathbf{r}_\mathrm{min}||_2$ **then**
      $\mathbf{r}_\mathrm{min} \leftarrow \mathbf{r}$;
      $\boldsymbol{\omega}_\mathrm{g}^\mathrm{best} \leftarrow \boldsymbol{\omega}_\mathrm{g}$;
  Perturb the guess: $\tilde{\boldsymbol{\omega}}_\mathrm{g} \leftarrow \boldsymbol{\omega}_\mathrm{g} + \delta\boldsymbol{\omega}$;
  Simulate charging with Alg. 2 and compute residuals $\tilde{\mathbf{r}}(\tilde{\boldsymbol{\omega}}_\mathrm{g})$;
  Numerically compute Jacobian of $\mathbf{r}$: $J_{r,\omega} \leftarrow \frac{\tilde{\mathbf{r}}-\mathbf{r}}{\delta\boldsymbol{\omega}_\mathrm{g}}$;
  Update guess (quasi-Newton step): $\boldsymbol{\omega}_\mathrm{g} \leftarrow \boldsymbol{\omega}_\mathrm{g} - J^{-1}_{r,\omega}\mathbf{r}$;
Return current $\boldsymbol{\omega}_\mathrm{g}$, $\mathbf{x}$ and $\boldsymbol{\lambda}$;

Algorithm 2: Charging simulation from initial guess

Data: $\mathbf{x}_0$, $\boldsymbol{\omega}_\mathrm{g}$, $\mathrm{SoC}_\mathrm{des}$, H
Result: $\mathbf{r}$, $\mathbf{x}$, $\boldsymbol{\lambda}$

$\mathbf{x}(0) \leftarrow \mathbf{x}_0$;
$[\boldsymbol{\lambda}(0),\, t_\mathrm{f}] \leftarrow \boldsymbol{\omega}_\mathrm{g}$;
for $t \leftarrow t_0$ **to** t_f **do**
  $\left[P_\mathrm{b}(t),\, P_\mathrm{hvch}(t),\, P_\mathrm{hp}(t)\right] \leftarrow \arg\min_{P_\mathrm{b},\, P_\mathrm{hvch},\, P_\mathrm{hp}} H(\cdot)$;
  Compute next states according to (6) and (14);
  Compute next costates according to (22b);
Return $\mathbf{r}$, $\mathbf{x}$ and $\boldsymbol{\lambda}$;

V. Results

All tests were run on a machine with 32GB RAM, 1.8GHz processor and 64-bit Windows 10. Problem (19) was solved with IPOPT [17] through CasADi [18] to obtain benchmark trajectories, which are then compared with the ones obtained with Alg. 1, as described in III and IV. For every test run (20 in total), $T_\mathrm{amb} \in [-15, 30]°C$, $\mathrm{SoC}(0) \in [0.1, 0.4]$ and $\mathrm{SoC}(t_\mathrm{f}) \in [0.7, 0.9]$ were randomized. Table I shows the parameters used during the tests. The PMP-based method leads to a reduction in computation time of about 99.87%, with an average difference between the benchmark solution and the one obtained with the PMP-based strategy of around 2.74%. This discrepancy is due to the fact that (12) is modeled as a smoothed ReLU, an approximation which becomes more accurate the smaller ϵ_σ is. The stopping tolerance δ_tol and the discretization also play a role in a method that would otherwise be exact.

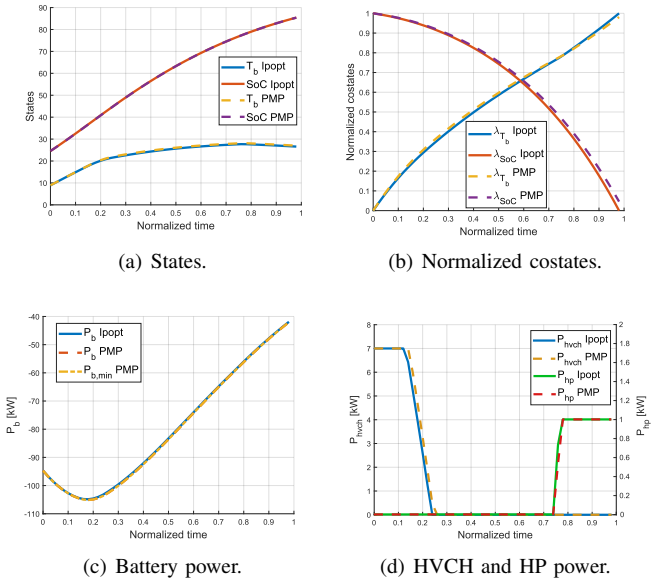
Fig. 2 shows an example run where the states, the (normalized) costates and the control inputs obtained with the PMP-based method are compared with the benchmark ones. It can be seen how the trajectories obtained with the two methods are nearly identical, as it should be, since we are comparing two different methods to solve the same problem. The added value of our method is shown in Table II, where it can be seen that the the PMP-based method correctly solves the 2PBVP (i.e., the vector of unknowns $\boldsymbol{\omega}$ has values similar to the benchmark) in significantly less time compared to IPOPT. This shows that the PMP-based method not only is of theoretical interest, since it uncovers the structure of the optimal solution, but it also allows a much faster retrieval of said solution.

VI. Conclusions and Future Work

This study presents a PMP-based method to solve the charging optimization problem for electric vehicles. The method solves a 2PBVP, starting from an initial guess obtained from a neural network, combining analytical results

TABLE I
PARAMETERS USED DURING TESTING

$\mathbf{r}_\mathrm{b} = [0.001, -0.317, 32.711]^\intercal\ \Omega\mathrm{K}^{-1}$
$\mathbf{u}_\mathrm{oc} = [85.297, -5.505, 382.849]^\intercal$ V, γ_0 =9.2 $\mathrm{W\,K}^{-1}$
$\eta_\mathrm{hvch} = 0.87$, α_hvch =$1 \times 10^{-5}\,\mathrm{W}^{-1}$, $\eta_\mathrm{Qhvch} = 0.95$
ϵ_σ =$1 \times 10^{-7}\,\mathrm{K}^2$, $T_\mathrm{b}^\mathrm{thres}$ =278.15 K
$p_{\mathrm{Qhp},0} = -18.056$, $p_{\mathrm{Qhp},1}$ =0.065 K^{-1}, α_hp =$1 \times 10^{-5}\,\mathrm{W}^{-1}$
c_b =1015 $\mathrm{J\,kg}^{-1}\,\mathrm{K}^{-1}$, m_b =371.79 kg, C_b =7.02×10^5 A s
$\mathbf{p}_{T_\mathrm{b}} = [-0.091, 56.967, -8.582]^\intercal$
$\mathbf{p}_\mathrm{SoC} = [-470.284, 107.891, 381.463]^\intercal$
$\mathbf{x}^\mathrm{min} = [253.15\mathrm{K}, 0]^\intercal$, $\mathbf{x}^\mathrm{max} = [313.15\mathrm{K}, 1]^\intercal$
$\mathbf{u}^\mathrm{min} = [-150, 0, 0]^\intercal$kW, $\mathbf{u}^\mathrm{max} = [0, 7, 1]^\intercal$kW
P_aux =0.5 kW, $t_\mathrm{f}^\mathrm{max}$ =2 h, $P_\mathrm{grid}^\mathrm{max}$ =200 kW
$N_\mathrm{max} = 10^6$, $\delta_\mathrm{tol} = 0.01$, $\delta\boldsymbol{\omega} = [0.1, 10, 100]$



(a) States.
(b) Normalized costates.
(c) Battery power.
(d) HVCH and HP power.

Fig. 2. Example run with comparison between IPOPT (continuous) and PMP (dashed)

from the PMP and a numerical procedure based on a quasi-Newtod method. The algorithm is able to solve the problem in a fraction of the time needed by a common off-the-shelf solver like IPOPT. Even if IPOPT already solves the problem quickly (see table II), this method serves as a building block to solve the comprehensive charge and trip-planning problem, i.e. the problem of routing a vehicle from A to B while optimally choosing where to recharge, managing the battery temperature and charging. Such a scenario may require solving the charging optimization problem multiple times, together with potential multiple routing problems between the various charging stations. If the building block for charging can be essentially solved instantaneously, then it can be combined with other similar blocks to build a computationally efficient solution to a bigger problem. Explicit expressions for the Hamiltonian could also be used as functions for policy evaluation in a Reinforcement Learning framework. Testing

TABLE II
RESULTS COMPARISON BETWEEN IPOPT AND PMP FOR THE EXAMPLE RUN

	ex. time	t_f	$\lambda_{T_\mathrm{b}}(0)$	$\lambda_\mathrm{SoC}(0)$
IPOPT	69.86 ms	46.04 min	-3.3 SEK/K	-165.27 SEK
PMP	0.18 ms	45.71 min	-3.51 SEK/K	-173.77 SEK

with real-world data and a more realistic model would be valuable to further verify the validity of this method, as well as comparing it with other benchmarks aside IPOPT.

REFERENCES

[1] S. M. S. U. Eskander and S. Fankhauser, "Reduction in greenhouse gas emissions from national climate legislation," *Nature Climate Change*, vol. 10, no. 8, pp. 750–756, 2020.
[2] E. Commission, "European climate law."
[3] H. Ritchie, P. Rosado, and M. Roser, "Data page: CO_2 emissions from transport," 2023. Data adapted from Climate Watch.
[4] N. Rauh, T. Franke, and J. F. Krems, "Understanding the impact of electric vehicle driving experience on range anxiety," *Human Factors*, vol. 57, no. 1, pp. 177–187, 2015. PMID: 25790577.
[5] M. Redelbach, E. D. Özdemir, and H. E. Friedrich, "Optimizing battery sizes of plug-in hybrid and extended range electric vehicles for different user types," *Energy Policy*, vol. 73, pp. 158–168, 2014.
[6] A. Dinc and M. Otkur, "Optimization of electric vehicle battery size and reduction ratio using genetic algorithm," in *2020 11th International Conference on Mechanical and Aerospace Engineering (ICMAE)*, pp. 281–285, 2020.
[7] M. Amjad, A. Ahmad, M. H. Rehmani, and T. Umer, "A review of evs charging: From the perspective of energy optimization, optimization approaches, and charging techniques," *Transportation Research Part D: Transport and Environment*, vol. 62, pp. 386–417, 2018.
[8] S. Sojoudi and S. H. Low, "Optimal charging of plug-in hybrid electric vehicles in smart grids," in *2011 IEEE Power and Energy Society General Meeting*, pp. 1–6, 2011.
[9] S. Pourazarm, C. G. Cassandras, and A. Malikopoulos, "Optimal routing of electric vehicles in networks with charging nodes: A dynamic programming approach," in *2014 IEEE International Electric Vehicle Conference (IEVC)*, pp. 1–7, 2014.
[10] W. Tang and Y. J. Zhang, "A model predictive control approach for low-complexity electric vehicle charging scheduling: Optimality and scalability," *IEEE Transactions on Power Systems*, vol. 32, no. 2, pp. 1050–1063, 2017.
[11] R. V. Gamkrelidze, *Discovery of the maximum principle*. Springer, 2006.
[12] S. Bauer, A. Suchaneck, and F. Puente León, "Thermal and energy battery management optimization in electric vehicles using pontryagin's maximum principle," *Journal of Power Sources*, vol. 246, pp. 808–818, 2014.
[13] R. Schmid, J. Buerger, and N. Bajcinca, "Energy management strategy for plug-in-hybrid electric vehicles based on predictive pmp," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 6, pp. 2548–2560, 2021.
[14] A. E. Bryson, *Applied optimal control: optimization, estimation and control*. Routledge, 2018. Chapter 8.
[15] A. Hamednia, N. Murgovski, J. Fredriksson, J. Forsman, M. Pourabdollah, and V. Larsson, "Optimal thermal management, charging, and eco-driving of battery electric vehicles," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 6, pp. 7265–7278, 2023.
[16] D. S. Naidu, *Optimal control systems*. CRC press, 2018. Chapter 2, Section 2.7, Subsection 2.7.4.
[17] I. I. Dikin, "Iterative solution of problems of linear and quadratic programming," *Sov. Math., Dokl.*, vol. 8, pp. 674–675, 1967.
[18] J. Andersson, J. Gillis, and G. e. a. Horn, "Casadi: a software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2018.