الجمهورية الجزائرية الديمقراطية الشعبية

THE PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA

وزارة التعليم العالي و البحث العلمي

THE MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

جامعة عمّار ثليجي بالأغواط

AMAR TELIDJI UNIVERSITY OF LAGHOUAT

كلية التكنولوجيا

FACULTY OF TECHNOLOGY

قسم الالكتروتقني

DEPARTMENT OF ELECTROTECHNIC

Master's Thesis

Domain: Science and Technology
Field: Electrical Engineering
Option: Automation and Systems

Presented by:
HASSANI Yassine Radouane and BENDJILALI Mohammed Tayeb

THEME

Optimal Power Management of an e-Racing Vehicle

Jury Members:

Atallah BENALIA	Pr.	President
Khaled RAYANE	Dr.	Examiner
M. Djameleddine BOUGRINE	Dr.	Supervisor
Bachir BENDJEDIA	Dr.	Co-Supervisor

Academic Year 2024/2025

Abstract

The Motor Vehicles Challenge, organized by the IEEE Vehicular Technology Society, is an annual competition focused on the development of advanced energy management strategies to enhance electric vehicle (EV) performance.

In the 2024 edition, the study focused on detailed modeling of powertrain losses, particularly in high-efficiency silicon carbide-based power converters and internal permanent magnet (IPM) motors. The system under consideration is a high-performance electric racing vehicle equipped with a hybrid energy storage system (HESS) that integrates lithium-ion batteries and supercapacitors.

The primary control objective is to minimize energy consumption and powertrain losses while satisfying thermal, electrical, and operational constraints. Equivalently, this objective can be formulated as maximizing the total energy stored at the final time in both the battery and the supercapacitor.

To achieve this, two optimal control approaches were investigated: Pontryagin's Minimum Principle (PMP) and Dynamic Programming (DP). While PMP provides efficient, locally optimal solutions based on necessary conditions, DP offers globally optimal control policies by exploring the entire state space at the expense of increased computational complexity. The comparative results highlight the trade-off between real-time feasibility and global optimality, providing valuable insights into energy-efficient control of high-performance EVs.

Index Terms: Energy management, power electronics converters, internal permanent magnet motor, hybrid energy storage system, electric vehicle, optimal control, energy loss minimization, dynamic programming.

الملخص

تُعدّ مسابقة تحدي المركبات، التي تنظمها جمعية EEEI لتكنولوجيا المركبات، حدثًا سنويًا يركّز على تطوير استراتيجيات متقدمة لإدارة الطاقة بهدف تحسين أداء المركبات الكهربائية. في نسخة عام 4202، تم التركيز على نمذجة دقيقة لخسائر منظومة الدفع، لا سيما في محولات القدرة عالية الكفاءة المعتمدة على كربيد السيليكون، ومحركات المغناطيس الدائم الداخلية (MPI). وقد تم تطبيق هذه النماذج على مركبة كهربائية مخصصة للسباقات، مجهزة بنظام هجين لتخزين الطاقة يجمع بين بطاريات الليثيوم-أيون والمكثفات الفائقة. يتمثل الهدف الأساسي في تقليل استهلاك الطاقة والخسائر في منظومة الدفع الكهربائي، مع مراعاة القيود الحرارية والكهربائية والتشغيلية. ويمكن صياغة هذا الهدف بشكل مكافئ كتعظيم الطاقة المخزنة في كل من البطارية والمكثف الفائق في نهاية فترة التشغيل. لتحقيق هذا الهدف، تم اعتماد طريقتين من نظرية التحكم المثالي: مبدأ بونترياغين الأدنى (PMP) والبرمجة الديناميكية (PD). فعلى الرغم من أن البرمجة الديناميكية تتطلب موارد حسابية كبيرة من حيث الوقت والذاكرة، إلا أنها تضمن حلولًا مثالية على المستوى العالمي، مما يكمل النتائج المحلية التي يوفرها مبدأ بونترياغين. وتبرز النتائج المقارنة المزايا والتحديات المرتبطة بكل منهما، ما يوفّر رؤى قيّمة لتصميم استراتيجيات فعالة لإدارة الطاقة في المركبات الكهربائية عالية الأداء. **الكلمات المفتاحية:** إدارة الطاقة، محولات إلكترونيات القدرة، محرك مغناطيس دائم داخلي، نظام تخزين طاقة هجين، مركبة كهربائية، تحكم مثالي، تقليل خسائر الطاقة، البرمجة الديناميكية.

Acknowledgements

First and foremost, we are profoundly grateful to Almighty God for bestowing upon us the strength, perseverance, and determination that enabled us to complete this work.

We wish to express our sincere appreciation to **Doctor BOUGRINE Djamaleddine**, our supervisor, whose exceptional guidance, continuous support, and thoughtful advice have been pivotal throughout the course of this dissertation. His expertise and dedication played a fundamental role in the realization of this research.

Our heartfelt thanks also go to **Doctor BENDJEDIA Bachir**, our co-supervisor, for her valuable contributions and insightful guidance. Her expertise, especially in the modeling aspects and critical review of our writing, was instrumental in shaping the quality of this work.

We are truly honored by the time and attention devoted by the examination committee. We extend our deep gratitude to **Professor** [**BENALIA Atallah**], President of the Jury, and **Doctor** [**RAYANE Khaled**], Member of the Jury, for their careful evaluation and constructive remarks, which enriched the final version of this dissertation.

We also take this opportunity to sincerely thank our families and friends, whose encouragement, patience, and unwavering support have been a constant source of motivation and strength throughout our academic path.

Finally, we gratefully acknowledge the contributions of all faculty members and administrative staff of the **Faculty of Technology** and the **Department of Electrical Engineering** at **Laghouat University**, whose support and commitment to excellence have deeply shaped our academic experience.

Dedication

We wholeheartedly dedicate this work to our cherished families — the unwavering pillars who stood by us with strength and love throughout our academic journey. Your unconditional support, boundless encouragement, and enduring faith in our abilities have formed the bedrock of every achievement we've realized.

Our sincere and profound gratitude goes to our esteemed supervisor, **Doctor BOUGRINE Djamaleddine**, whose insightful guidance, deep expertise, and unwavering dedication have greatly enriched our knowledge and shaped the course of this research.

We also extend our heartfelt appreciation to **Doctor BENDJEDIA Bachir**, who has been a guiding presence since the very beginning. His early encouragement and continued support have played a vital role in our development and progress throughout this journey.

This work is further dedicated to all those who are devoted to fostering a more sustainable and forward-thinking future in the fields of automation and systems. We hope that the insights and contributions presented here will serve to inspire and empower future researchers and professionals to break new ground in this ever-evolving discipline.

To every individual who walked beside us during this demanding journey—offering words of support, acts of kindness, and motivation—we extend our heartfelt thanks. Your presence has been a source of strength, and your belief in us, a lasting gift.

Contents

List of Figures

List of Tables

Nomenclature :

Symbol	Definition
η	Converter efficiency
η_{bat}	Battery efficiency
η_{sc}	Supercapacitor efficiency
γ	Exponent in iron loss model
λ_d	Flux in d-axis
λ_q	Flux in q-axis
ω_e	Electrical angular speed
Ω_m	Mechanical angular speed
ω_m	Motor speed
ϕ_{pm}	Permanent magnet flux
c_{fe}	Iron loss coefficient
c_{str}	Stray loss coefficient
C_{rated}	Capacitance of the SC pack
E_d, E_q	Back EMF in d-/q-axis
E_{sw}	Switching energy
f_{sw}	Switching frequency
I_d, I_q	Motor currents in d-/q-axis
i_{bat}	Battery output current
i_{cell}	Battery cell current
$i_{\text{ch,bat}}$	Battery charging current
i_{sc}	Supercapacitor current
J	Moment of inertia
k	Current direction indicator
k_1, k_2, k_5, k_6	Weights in cost function
L_d, L_q	Inductance in d-/q-axis
m_{bat}	Duty cycle of battery converter
n_{bat_par}	Number of battery cells in parallel
n_{bat_seri}	Battery cells in series
n_{sc}	Number of SC modules
p	Pole pairs
P_v	Mechanical power
P_{cu}	Copper loss
P_{fe}	Iron loss
P_{str}	Stray loss

Symbol	**Definition**
P_{sw}	Switching loss
P_{tot}	Total electric machine loss
P_{loss}	Total powertrain loss
$P_{\text{HS loss}}, P_{\text{LS loss}}$	High/low side MOSFET loss
Q_{rated}	Rated battery capacity
R_s	Stator winding resistance
R_{int}	Battery internal resistance
R_{th}	Thermal resistance
R_{esr}	SC pack equivalent series resistance
T_e	Electromagnetic torque
T_j	Junction temperature
$T_{j,\text{max}}$	Max junction temperature
T_{amb}	Ambient temperature
T_{load}	Load torque
U_d, U_q	Input voltages in d-/q-axis
u_{bat}	Battery pack voltage
u_{cell}	Single battery cell voltage
u_{dc}	DC bus voltage
u_{OCV}	Open circuit voltage
$u_{\text{sc}}, U_{\text{sc}}$	Supercapacitor voltage
u_{sc0}	Initial SC voltage
SoC	State of charge
$\dot{\text{SoC}}$	SoC derivative
$\dot{U}_{\text{sc}}$	SC voltage derivative
λ_1, λ_2	Adjoint variables (PMP)
H	Hamiltonian function
N	Number of SiC MOSFET modules
u_{EV}	Electric vehicle input
EV	Electric Vehicle
EMS	Energy Management Strategy
HESS	Hybrid Energy Storage System
PMP	Pontryagin Minimum Principle
IPM	Interior Permanent Magnet motor
IPMSM	Interior Permanent Magnet Synchronous Motor
OCV	Open Circuit Voltage

General Introduction :

The global transportation sector is undergoing a profound transformation in response to climate change, the depletion of fossil fuels, and the need to decarbonize mobility [8]. One of the most promising developments in this context is the emergence of electric vehicles (EVs), which offer high energy efficiency, reduced local emissions, and decreased reliance on finite hydrocarbon resources [8, 10]. However, the transition from internal combustion engines to fully electric powertrains introduces numerous technical challenges, particularly in energy storage, power conversion efficiency, and optimal real-time energy management—challenges that become even more critical in high-performance applications such as electric racing [6].

Electric racing vehicles place demanding requirements on the powertrain, requiring rapid power delivery, frequent regeneration, and optimal use of limited onboard energy. A widely adopted solution is the Hybrid Energy Storage System (HESS), which combines the high energy density of lithium-ion batteries with the high power density and responsiveness of supercapacitors [14]. While this architecture improves performance and flexibility, it also increases the complexity of system modeling and energy management.

To fully exploit the benefits of HESS-based architectures, intelligent Energy Management Strategies (EMS) must be developed. These strategies govern the power flow between energy sources and the electric motor while accounting for component constraints, efficiency losses, and real-time driving conditions [5, 6]. The design of such strategies plays a critical role in determining vehicle performance, especially in scenarios with limited energy availability such as electric racing.

In this context, optimal control theory offers a rigorous framework for formulating and solving energy management problems. Two prominent methods are the Pontryagin Minimum Principle (PMP) and Dynamic Programming (DP). PMP provides necessary conditions for optimality by constructing a Hamiltonian system that includes both state and costate dynamics [19, 20], while DP solves the problem globally by recursively evaluating the cost-to-go function using Bellman's principle [25, 28]. Each approach offers unique strengths and trade-offs in terms of solution quality, computational complexity, and ability to handle constraints.

This thesis is conducted in the context of the IEEE Vehicular Technology Society's Motor Vehicles Challenge (MVC) 2024 [6]. The challenge involves a detailed simulation framework for an electric racing vehicle equipped with a hybrid energy system, incorporating high-fidelity models of battery and supercapacitor dynamics, silicon carbide (SiC) converters, and core motor losses [7, 10].

The objective of this work is to model the powertrain of the EV, formulate the energy management problem, and apply both PMP and DP to compute optimal control strategies under various driving scenarios. By comparing the results of the two approaches, the study highlights their respective strengths, limitations, and applicability to real-world EV control.

The thesis is organized as follows:

- **Chapter 1** presents the modeling of the EV powertrain, including detailed representations of the battery, supercapacitor, converters, and electric motor.
- **Chapter 2** introduces the theoretical foundations of the Pontryagin Minimum Principle and Dynamic Programming, supported by simple academic examples to illustrate their use in optimal control.
- **Chapter 3** applies both methods to the EV system, analyzes their performance under different driving cycles (trivial case, ARTEMIS, UDDS), and discusses the resulting control policies and energy efficiency outcomes.

This work contributes to the field of energy-efficient electric vehicle control by demonstrating how both local and global optimal control methods can be used to manage power in high-performance EVs, offering insights into practical trade-offs between computational efficiency, optimality, and system complexity.

Chapter 1

ELECTRIC VEHICLE POWERTRAIN MODEL

1.1 Introduction :

Electric vehicles (EVs) represent a key solution for sustainable and energy-efficient transportation systems. The increasing demands on performance, range, and efficiency have led to the integration of advanced powertrain architectures and hybrid energy storage systems (HESS) [8, 25].

In the context of the IEEE VTS Motor Vehicles Challenge 2024, this chapter presents a comprehensive modeling of the powertrain system of a high-performance electric racing vehicle, building upon methodologies from previous editions of the challenge [1, 5, 6].

The studied architecture incorporates a lithium-ion battery pack and a supercapacitor (SC) module, interconnected through bi-directional DC/DC converters, forming a hybrid energy storage system. This combination leverages the high energy density of batteries and the high power density of supercapacitors to enhance both efficiency and dynamic response [4, 14].

The powertrain further includes a three-phase inverter and an Interior Permanent Magnet Synchronous Motor (IPMSM), known for its high efficiency and torque density [19, 20]. Each component is modeled to reflect realistic operational behavior, including dynamic and static losses [10, 12].

A detailed loss model is established for all key subsystems: battery and SC converters, inverter, and electric machine [7, 13]. Furthermore, the system is evaluated under various standard driving cycles—UDDS, WLTC Class 3, and ARTEMIS—to test energy management strategies under different driving conditions [21, 22, 23, 24].

1.2 Energy Storage Components Model :

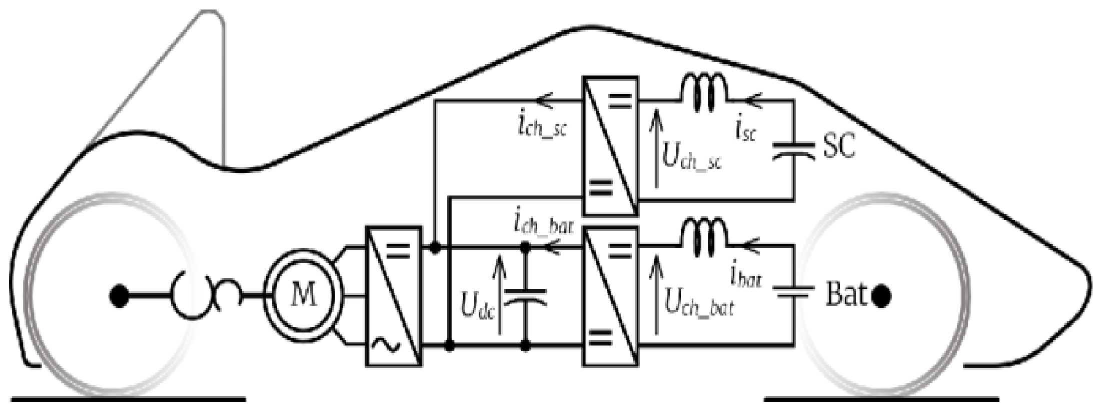


Figure 1.1: Configuration Of The Studied EV

The configuration of the studied Electric vehicle (EV), which serves as the challenge platform for this project [6], is illustrated in Fig. 1.1. This platform is derived from a high-performance racing vehicle developed at the University of Nottingham. Its original architecture has been enhanced by integrating a hybrid energy storage system (HESS) composed of a conventional lithium-ion battery pack and an additional supercapacitor module. The two storage systems are interconnected via two bidirectional DC/DC converters, enabling flexible energy exchange between the battery and the supercapacitor.

This HESS configuration is designed to leverage the high energy density of the battery and the high power density of the supercapacitor, aiming to improve dynamic performance, extend battery lifespan, and potentially increase racing autonomy. Effective management of the HESS is critical, particularly under aggressive driving conditions, where the supercapacitor can absorb or deliver peak power, thereby reducing battery stress.

The electrical energy from the HESS is supplied to an internal permanent magnet synchronous motor (IPMSM) via a three-phase inverter. This motor, known for its high efficiency and power density, provides traction by driving the rear wheel through a mechanical gearbox. During operation, the system accounts for the energy losses associated with each component, including the bidirectional DC/DC converters, inverter, and the electric traction motor, to ensure realistic performance analysis and energy flow estimation.

1.2.1 Modeling Of Battery System :

The battery system consists of a number of cells that are connected in series and parallel as described in Fig. 1.1. Each cell is modeled by its equivalent circuit as follows [8, 14].

$$\begin{cases} u_{\text{cell}} = u_{\text{OCV}}(\text{SoC}) - i_{\text{cell}} \cdot R_{\text{int}} \\ \text{SoC} = 1 - \frac{1}{3600 \cdot Q_{\text{rated}}} \int i_{\text{cell}} \, dt \end{cases} \tag{1.1}$$

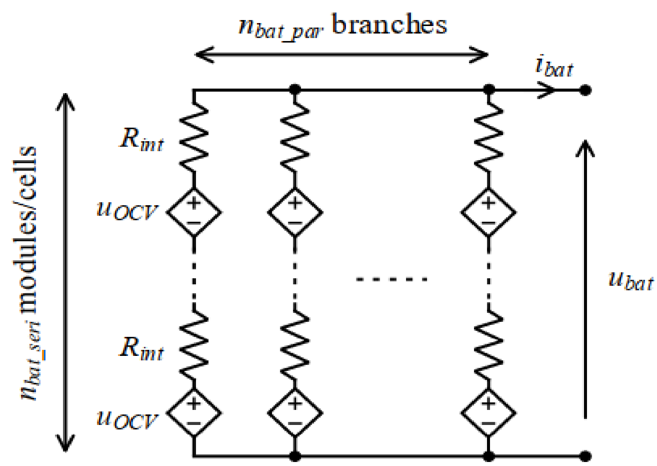


Figure 1.2: Connection In Battery System

Where, u_{cell} and i_{cell} are voltage and current of each cell, respectively. R_{int} is the internal resistance of battery cell, Q_{rated} denotes the cell capacity. In the model (1.2), the open circuit voltage u_{OCV} is a function of the state-of-charge of the battery cell which is provided by an experimental characteristic.

At the pack level, the overall electrical characteristics are determined by the configuration of cells in series and parallel. The total output voltage u_{bat} of the battery pack is the cumulative sum of the voltages of all cells connected in series, while the current i_{bat} drawn from the pack is shared equally among the parallel branches, assuming uniform cell behavior. This relationship is mathematically expressed as: The relationship of the output voltage u_{bat} and current i_{bat} of the whole battery pack and those of each cell can be described by (1.2)

$$\begin{cases} u_{\text{bat}} = n_{\text{bat\_seri}} \cdot u_{\text{cell}} \\ i_{\text{cell}} = \frac{1}{n_{\text{bat\_par}}} \cdot i_{\text{bat}} \end{cases} \tag{1.2}$$

1.2.2 Modeling Of Super Capacitor System :

The supercapacitor (SC) system consists of multiple individual cells or modules arranged in series and parallel to meet the voltage and energy requirements of the application. In this challenge, a simplified yet effective electrical model is employed to describe the dynamic behavior of the SC pack, as given in Equation (1.3):

$$u_{\text{sc}} = u_{\text{sc0}} - R_{\text{esr}} \cdot i_{\text{sc}} - \frac{1}{C_{\text{rated}}} \int i_{\text{sc}} \, dt \tag{1.3}$$

In this model, u_{sc} is the terminal voltage of the complete supercapacitor pack, and u_{sc0} denotes its initial voltage. The term R_{esr} represents the equivalent series resistance, which accounts for the internal resistive losses during charge and discharge. C_{rated} is the total equivalent capacitance of the SC pack, and i_{sc} is the instantaneous current flowing into or out of the system. The last term models the voltage drop due to the accumulation or depletion of charge over time, following the basic principle of capacitor behavior.

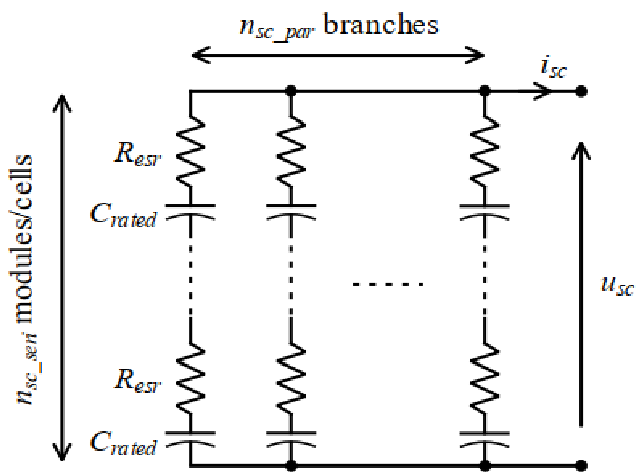


Figure 1.3: Connection In Supercapacitor System

The overall electrical characteristics of the SC pack are influenced by the number of modules connected in series and in parallel. These relationships are defined in Equation (1.4):

$$\begin{cases} u_{\text{sc}} = n_{\text{sc\_seri}} \cdot u_{\text{modul}} \\ i_{\text{cell}} = \frac{1}{n_{\text{sc\_par}}} \cdot i_{\text{sc}} \end{cases} \tag{1.4}$$

Here, $n_{\text{sc\_seri}}$ is the number of modules connected in series, which determines the total voltage, while $n_{\text{sc\_par}}$ is the number of parallel branches, affecting the total current handling capability. u_{modul} is the voltage across a single module, and i_{cell} is the current through each cell, assuming uniform behavior across modules. This modular arrangement ensures the scalability of the SC system for high-power applications. [14], such as peak power support and regenerative braking in electric vehicles.

1.3 Power Electronics Converter Model :

DC-DC bi-directional power converters are used to connect each energy storage source to the DC bus [7, 10]. For the battery converter, the voltage and current are modeled by the following equations (the supercapacitor converter follows the same structure):

$$\begin{cases} u_{\text{ch,bat}} = m_{\text{bat}} u_{\text{dc}} \\ i_{\text{ch,bat}} = m_{\text{bat}} i_{\text{bat}} \eta^{-k} \end{cases} \tag{1.5}$$

where:

- m_{bat} is the duty cycle of the converter,
- η is the efficiency of the converter,
- k is a parameter representing the direction of the battery current i_{bat},

with:

$$k = \begin{cases} 1, & \text{if } i_{\text{bat}} < 0 \\ -1, & \text{if } i_{\text{bat}} > 0 \end{cases}$$

1.4 Electric Machine Model :

The model of the traction IPM (Interior Permanent Magnet) motor is developed in the d-q reference frame [8, 9, 10] .and consists of three parts: **electrical**, **electro-mechanical**, and **mechanical** models.

Electrical Model :

The electrical equations relate the input voltages U_d, U_q and motor currents I_d, I_q:

$$I_d = \frac{U_d - E_d}{L_d s + R_s}, \quad I_q = \frac{U_q - E_q}{L_q s + R_s} \tag{1.6}$$

where:

- L_d, L_q are the d- and q-axis inductances,
- R_s is the stator winding resistance,
- E_d, E_q are the back electromotive forces (EMF).

Electro-Mechanical Model :

The torque generated by the motor is given by:

$$T_e = \frac{3p}{2} \left(\phi_{\text{pm}} I_q + (L_d - L_q) I_d I_q\right) \tag{1.7}$$

The EMFs in the d- and q-axes are calculated as:

$$E_d = -L_q I_q \omega_e, \quad E_q = L_d I_d \omega_e + \phi_{\text{pm}} \omega_e \tag{1.8}$$

where:

- T_e is the electromagnetic torque,
- ϕ_{pm} is the permanent magnet flux,
- ω_e is the electrical angular speed,
- p is the number of pole pairs,
- $\omega_e = p\Omega_m$, where Ω_m is the rotor mechanical speed.

Mechanical Model :

The mechanical dynamics of the motor are given by:

$$T_e - T_{\text{load}} = J \frac{d\Omega_m}{dt} \tag{1.9}$$

where:

- T_{load} is the load torque,
- J is the moment of inertia of the motor,
- Ω_m is the rotor mechanical speed.

1.5 Formulation Of Losses:

1.5.1 Machine Losses Model :

The electric machine losses are primarily attributed to the physical properties of the motor components [8, 19]. These losses can be categorized as follows :

- **Iron Loss** (P_{fe}) : This includes hysteresis losses and eddy current losses occurring in the motor core when subjected to an alternating magnetic field.
- **Copper Loss** (P_{cu}) : Losses generated due to the resistance of the stator winding during current flow.
- **Stray Loss** (P_{str}) : These include losses caused by factors such as winding layout, slot harmonics, and mechanical imperfections.

The total motor loss P_{loss} in this paper is written as follows.

Motor Copper Losses :

$$P_{\text{cu}} = \frac{3}{2} R_s \left(I_d^2 + I_q^2\right) \tag{1.10}$$

Motor Iron (Core) Losses :

$$P_{\text{fe}} = c_{\text{fe}} \cdot \omega_e^{\gamma} (\lambda_d^2 + \lambda_q^2) \tag{1.11}$$

Motor Stray Losses :

$$P_{\text{stray}} = c_{\text{str}} \cdot \omega_e^2 \left(I_d^2 + I_q^2\right) \tag{1.12}$$

The Motor Losses Will Be Expressed By :

$$P_{\text{motor}} = P_{\text{cu}} + P_{\text{fe}} + P_{\text{stray}}$$

$$P_{\text{motor}} = \frac{3}{2}R_s(I_d^2 + I_q^2) + c_{\text{fe}}\omega_e^{\gamma}(\lambda_d^2 + \lambda_q^2) + c_{\text{str}}\omega_e^2(I_d^2 + I_q^2) \tag{1.13}$$

$$\begin{aligned} P_{\text{motor}} = & \frac{3}{2}R_s\left(I_d^2 + I_q^2\right) \\ & + c_{\text{fe}} \cdot \omega_e^{1.6}\left[(L_d I_d + \phi_{\text{pm}})^2 + (L_q I_q)^2\right] \\ & + c_{\text{str}} \cdot \omega_e^2\left(I_d^2 + I_q^2\right) \end{aligned} \tag{1.14}$$

Where :

- R_s: Motor winding resistance [Ω]
- I_d, I_q: d-axis and q-axis motor currents [A]
- ω_e: Electrical angular speed [$\mathrm{rad\,s^{-1}}$]
- c_{fe}: Iron loss coefficient [$\mathrm{Ws^{1.6}}$]
- c_{str}: Stray loss coefficient [$\mathrm{Ws^2}$]
- γ: Iron loss exponent, empirically chosen as $\gamma = 1.6$
- $\lambda_d = L_d I_d + \phi_{\text{pm}}$: d-axis magnetic flux [Wb]
- $\lambda_q = L_q I_q$: q-axis magnetic flux [Wb]
- L_d, L_q: d- and q-axis inductances [H]
- ϕ_{pm}: Permanent magnet flux linkage [Wb]

1.5.2 Powertrain Total Losses :

The total losses, denoted as P_{loss}, represent the overall energy losses in the following components of the electric powertrain [10, 20]:

- Battery DC-DC Converter (P_{bat})
- Supercapacitor DC-DC Converter (P_{sc})
- Inverter (P_{inv})
- Traction Motor (P_{motor})

In the first we will start with :

Battery Losses :

$$P_{\text{bat}} = n_{\text{bat}}\left[\left(\frac{i_{\text{ch\_bat}}}{n_{\text{bat}}}\right)^2 R_{\text{on}} + \left|\frac{i_{\text{ch\_bat}}}{n_{\text{bat}}}\right| \cdot R_{\text{d}}\right] \tag{1.15}$$

Where :

- n_{bat}: Number of battery converter phases
- $i_{\text{ch\_bat}}$: Battery charging current [A]
- R_{on}: On-state resistance [Ω]
- R_{d}: Diode conduction resistance [Ω]

Supercapacitor Losses :

$$P_{\text{sc}} = n_{\text{sc}} \left[\left(\frac{i_{\text{sc}}}{n_{\text{sc}}} \right)^2 R_{\text{on}} + \left| \frac{i_{\text{sc}}}{n_{\text{sc}}} \right| \cdot R_{\text{d}} \right] \quad , \quad i_{\text{sc}} = \frac{T_e \cdot \omega_m}{U_{\text{sc}}} - i_{\text{ch\_bat}} \tag{1.16}$$

Where :

- n_{sc}: Number of SC converter phases
- i_{sc}: Supercapacitor current [A]
- T_e: Electromagnetic torque [N m]
- ω_m: Mechanical angular speed [rad s^{-1}]
- U_{sc}: Supercapacitor voltage [V]

Inverter Losses :

$$P_{\text{inv}} = 3n_{\text{inv}} \left[\left(\frac{i_{\text{inv}}}{n_{\text{inv}}} \right)^2 R_{\text{on}} + \left| \frac{i_{\text{inv}}}{n_{\text{inv}}} \right| \cdot R_{\text{d}} \right] \quad , \quad i_{\text{inv}} = \sqrt{I_d^2 + I_q^2} \tag{1.17}$$

Where :

- n_{inv}: Number of inverter phases
- i_{inv}: Inverter output current (computed from I_d, I_q)

So the total power loss can be written as:

$$P_{\text{loss}} = P_{\text{bat}} + P_{\text{sc}} + P_{\text{inv}} + P_{\text{motor}} \tag{1.18}$$

1.6 The Driving Cycles :

To ensure a comprehensive evaluation of The Proposed Energy Management Strategy (EMS) under diverse driving conditions, three (03) standard driving cycles are considered Here : ARTEMIS, UDDS, and WLTC class 3. Each Cycle represents a specific type of Driving Environment and Behavior, providing a robust testing framework for EMS performance [21, 22, 23].

1.6.1 UDDS – Urban Dynamometer Driving Schedule :

the UDDS cycle, also known as the LA-4 cycle, is a standard procedure developed by the United States Environmental Protection Agency (EPA). It simulates typical urban driving conditions characterized by low-speed operations, frequent stops, and mild accelerations. This cycle is predominantly used for measuring emissions and fuel consumption in city driving [21].

- **Average speed :** ∼31.5 km/h.
- **Duration :** 1369 seconds.
- **Stop duration :** ∼18% of the cycle.
- **Application :** Ideal for testing urban energy efficiency and stop-and-go behavior.

1.6.2 WLTC Class 3 – Worldwide Harmonized Light Vehicles Test Cycle :

the WLTC cycle was developed by the UNECE to standardize global emissions and energy consumption testing. class 3 targets High-Performance vehicles with powerful engines. It includes four phases: Low, Medium, High, and Extra-High speed, thus providing a broad coverage of realistic driving scenarios[22].

- **Average speed:** ∼46.5 km/h.
- **Duration:** 1800 seconds.
- **Characteristics:** Mix of urban, suburban, and highway conditions.
- **Application:** Suitable for evaluating EMS performance across diverse conditions.

1.6.3 ARTEMIS – Assessment And Reliability Of Transport Emission Models And Inventory System :

The ARTEMIS driving cycle is derived from real-world driving data collected across various European road types. It comprises three Sub-Cycles : Urban, Rural, and Motorway, and reflects realistic driving dynamics, including aggressive accelerations, frequent decelerations, and idling periods[23].

- **Origin:** Based on measured on-road European driving behavior.
- **Structure:** Includes urban, rural, and motorway segments.
- **Characteristics:** Realistic acceleration profiles and transient driving effects.
- **Advantage:** High representativeness of actual driving patterns.

Here is a comparison table that highlights the key differences between the three driving cycles: Comparison Of The Driving Cycles [24]

Feature	UDDS	WLTC Class 3	ARTEMIS
Origin	U.S. EPA(Urban)	UNECE (Global standard)	European real-world data
Purpose	Urban driving simulation	General mixed driving (city to highway)	Realistic simulation of urban, rural, and motorway driving
Average speed	∼31.5 km/h	∼46.5 km/h	Varies by sub-cycle (Urban: ∼20 km/h, Rural: ∼52 km/h, Motorway: ∼95 km/h)
Duration	1369 s	1800 s	Urban: 994 s, Rural: 1082 s, Motorway: 1068 s
Speed range	Low(0–90 km/h)	Low to extra-high (0–131 km/h)	Full range (0–130+ km/h) with realistic transitions
Acceleration profile	Smooth,synthetic	Mixed,semi-realistic	Realistic and aggressive (transient behavior included)
Applications	Emissions and energy testing for urban use	Global harmonized testing for certification	Research and simulation of real driving behaviors in Europe

Table 1.1: Comparison Of The Driving Cycles

In electric vehicle (EV) applications, T_{load} accounts for all resistive forces (such as aerodynamic drag and rolling resistance) translated to the motor shaft.

1.7 EMR-Based Powertrain Control :

Overall, this section bridges the gap between theoretical modeling and its implementation in a dynamic simulation environment, offering a practical basis for performance analysis and control validation of EV powertrains.

The implementation of the presented power converter loss and efficiency model for both battery and SC DC-DC converters is shown in Fig.(1.4), where the losses and efficiency can be simulated and obtained together with the powertrain system control model.

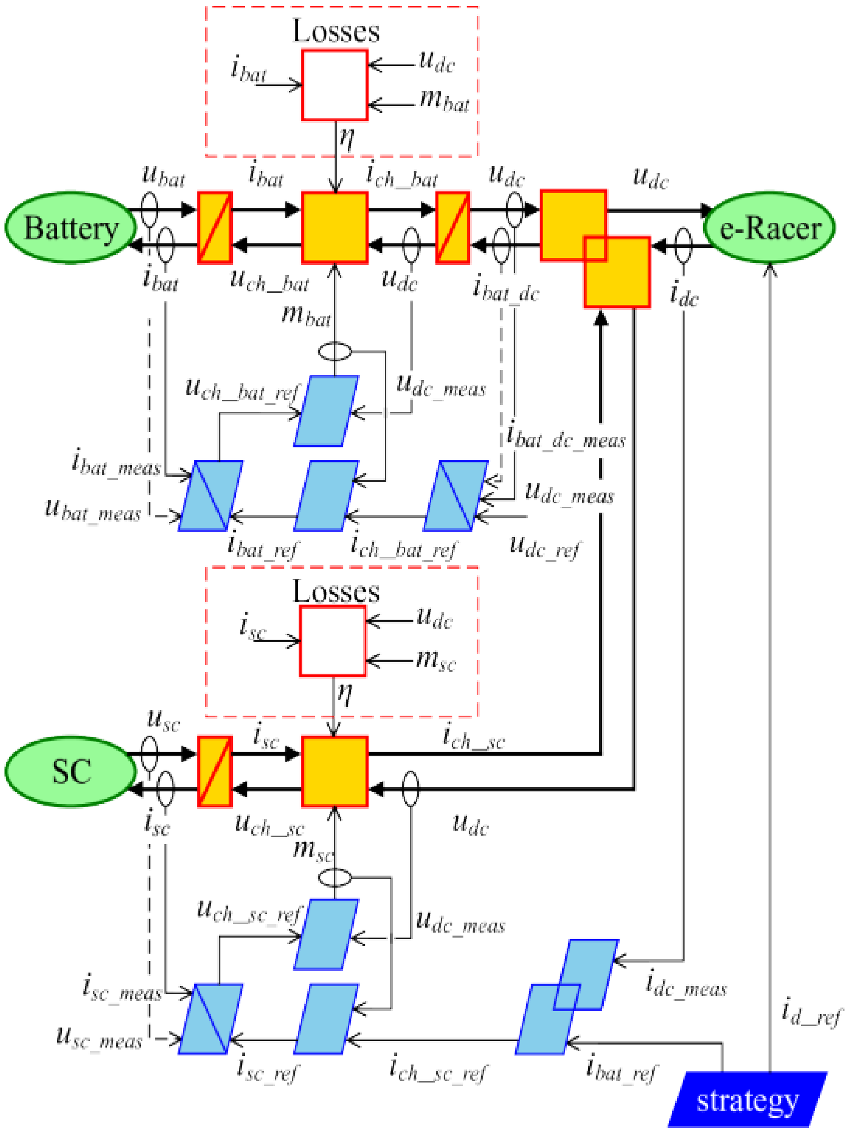


Figure 1.4: Powertrain System Representation Including Power Converter Losses

As illustrated in Fig. 1.4, we will use the approach of Energetic Macroscopic Representation (EMR) for the control of the powertrain [12], as EMR has been widely applied for energy management and control of complex electro-mechanical systems such as electric vehicles [13, 14], railways [15], and subways [16].

1.8 Conclusion :

In this chapter, a detailed and modular modeling of the electric vehicle powertrain has been presented, reflecting the architecture of the platform used in the IEEE VTS Motor Vehicles Challenge 2024. The model includes the hybrid energy storage system (battery and supercapacitor), bi-directional DC/DC converters, inverter, and an IPMSM electric traction motor, with each component accurately described in terms of both dynamic behavior and internal losses.

Loss formulations were developed to account for key inefficiencies in the converters and motor, ensuring realistic energy flow simulations. These models were applied under various standardized driving cycles (UDDS, WLTC Class 3, and ARTEMIS), enabling the analysis of system performance under diverse driving profiles.

Moreover, the powertrain control strategy is based on the Energetic Macroscopic Representation (EMR) framework, which facilitates the organization of energy flows and ensures consistent integration of control loops. The EMR-based model provides a solid basis for implementing and validating advanced control strategies, such as optimal energy management algorithms.

The comprehensive modeling and loss-aware approach described in this chapter serve as the foundation for the control optimization and simulation studies presented in the following chapters.

Chapter 2

Optimal Control Theory

2.1 Introduction :

Optimal control theory is a cornerstone of modern applied mathematics and engineering, offering rigorous tools to determine how to steer dynamical systems toward desired behaviors while minimizing or maximizing a performance criterion [26]. This performance is typically captured through a cost functional (to be minimized) or a reward functional (to be maximized), defined over a time horizon and dependent on both the system's state and its control inputs.

Among the most prominent and widely studied mathematical frameworks to address such control problems are:

- **Pontryagin's Minimum Principle (PMP) :** which provides a set of necessary conditions for optimality derived from the calculus of variations and the Hamiltonian formulation of dynamics [27, 28].
- **Dynamic Programming (DP) :** which offers a global, recursive strategy based on Bellman's Principle of Optimality and value functions [29].

Understanding both PMP and DP is essential for control system designers and optimization practitioners. While PMP is often preferred for continuous-time systems and analytical insight [27], DP is advantageous for real-time implementation, decision-making under uncertainty, and high-level control strategies [25, 29].

In this chapter, we explore both Pontryagin's Minimum Principle and Dynamic Programming in detail, laying the theoretical groundwork and comparing their strengths, limitations, and suitability in the context of optimal energy management for electric vehicle systems [25, 27, 28].

2.2 Pontryagin Minimum Principle (PMP) :

The Pontryagin Minimum Principle (PMP) establishes a set of necessary conditions for determining optimal control strategies in dynamical systems. It is particularly powerful in solving problems where the objective is to minimize a performance index (or cost functional) subject to differential constraints arising from the system dynamics [25, 26, 27, 28].

A classical optimal control problem suitable for PMP involves the following components:

- A system of differential equations describing the state evolution, initialized with a known condition;
- A cost functional to be minimized, typically composed of an integral (running cost) and a terminal cost;
- Constraints on the admissible control inputs;
- A set of adjoint (costate) equations derived from the Hamiltonian formulation.

Consider A Dynamical System Governed By :

$$\dot{x}(t) = f(x(t), u(t), t), \quad x(0) = x_0 \tag{2.1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}^m$ is the control input, and $t \in [0, t_f]$ is time.

The Objective Is To Minimize The Following Cost Functional :

$$\xi = h(x(t_f)) + \int_0^{t_f} L(x(t), u(t), t)\, dt \tag{2.2}$$

where $h(x(t_f))$ represents the terminal cost and $L(x, u, t)$ denotes the running cost.

According to PMP, if $u^*(t)$ is an optimal control and $x^*(t)$ is the corresponding optimal state trajectory, then there exists a costate vector $\lambda(t) \in \mathbb{R}^n$ such that the following conditions hold:

Hamiltonian Function :

The Hamiltonian is defined as:

$$H(x, u, \lambda, t) = L(x, u, t) + \lambda^\top f(x, u, t) \tag{2.3}$$

Optimality Condition :

The optimal control minimizes the Hamiltonian:

$$u^*(t) = \arg\min_{u \in \mathcal{U}} H(x^*(t), u, \lambda(t), t) \tag{2.4}$$

Transversality Condition :

In optimal control theory, the transversality condition provides boundary conditions for the adjoint (costate) variables at the final time t_f. These conditions are essential for solving the system of state and costate differential equations [25, 27].

In our work, we consider the case where the final time t_f is fixed, while the final state $x(t_f)$ is free. This corresponds to the classical setting in which the control objective depends on the terminal state but no constraint is imposed on its value.

Under this condition, the transversality condition states that the costate vector $\lambda(t)$ at the final time must satisfy:

$$\lambda(t_f) = \left.\frac{\partial h}{\partial x}\right|_{t=t_f} \tag{2.5}$$

where:

- $\lambda(t_f)$ is the adjoint vector (or costate) evaluated at the final time t_f,
- h is the terminal cost function,
- x represents the state vector.

This relation expresses the fact that the marginal variation of the cost function with respect to each state variable at the final time determines the corresponding boundary value of the adjoint variables.

2.2.1 Example: Minimum-Energy Control Problem :

Consider a simple optimal control problem where the objective is to minimize the control effort required to move a system from an initial state to rest.

Problem Statement :

Minimize the cost functional:

$$\xi = \int_0^1 u^2(t)\, dt \tag{2.6}$$

subject to the system dynamics:

$$\dot{x}(t) = u(t), \quad x(0) = 1$$

with the terminal constraint:

$$x(1) = 0$$

Solution Using PMP :

1. Define the Hamiltonian:
$$H(x, u, \lambda, t) = u^2 + \lambda u \tag{2.7}$$
2. State equation:
$$\dot{x} = \frac{\partial H}{\partial \lambda} = u \tag{2.8}$$
3. Costate equation:
$$\dot{\lambda} = -\frac{\partial H}{\partial x} = 0 \quad \Rightarrow \quad \lambda(t) = \text{const}$$

4. Optimality condition:

$$\frac{\partial H}{\partial u} = 2u + \lambda = 0 \quad \Rightarrow \quad u^*(t) = -\frac{\lambda}{2}$$

5. Substitute into state equation:

$$\dot{x}(t) = -\frac{\lambda}{2} \quad \Rightarrow \quad x(t) = -\frac{\lambda}{2}t + C$$

Apply the initial condition $x(0) = 1 \Rightarrow C = 1$, so:

$$x(t) = -\frac{\lambda}{2}t + 1$$

Apply the terminal condition $x(1) = 0$:

$$0 = -\frac{\lambda}{2} \cdot 1 + 1 \Rightarrow \lambda = 2$$

6. Final result:

$$u^*(t) = -1, \quad x^*(t) = 1 - t$$

2.2.2 State Constraints :

In addition to control constraints, many real-world control problems impose explicit state constraints, which require the state vector $x(t)$ to remain within a certain admissible set $\mathcal{X}$ for all $t \in [0, t_f]$. These constraints can take the form:

$$g(x(t)) \leq 0 \tag{2.9}$$

or more generally,

$$m(x(t), u(t)) \leq 0 \tag{2.10}$$

To incorporate state constraints into the Pontryagin framework, the Hamiltonian is extended by introducing Lagrange multipliers $\mu(t) \geq 0$, yielding:

$$\tilde{H}(x, u, \lambda, \mu, t) = L(x, u, t) + \lambda^\top f(x, u, t) + \mu^\top g(x(t)) \tag{2.11}$$

The multipliers $\mu(t)$ must satisfy the complementary slackness condition:

$$\mu_i(t) \cdot g_i(x(t)) = 0, \quad \mu_i(t) \geq 0, \quad \forall i \tag{2.12}$$

These additional terms ensure that the PMP still provides necessary conditions for optimality even in the presence of inequality constraints on the states. When a constraint is active, the corresponding multiplier is positive, enforcing the constraint. When inactive, the multiplier is zero, and the constraint does not affect the Hamiltonian.

Terminal state constraints, such as $x(t_f) = x_f$, can also be handled using similar transversality conditions or by fixing the terminal state directly.

2.2.3 Shooting Method

The shooting method is a widely used numerical technique for solving two-point boundary value problems (TPBVPs), which naturally arise in the application of Pontryagin's Minimum Principle (PMP). In PMP, the optimal control problem is reduced to solving a system of ordinary differential equations (ODEs) involving both the state and costate (adjoint) variables, subject to boundary conditions at the initial and terminal times.

The main idea behind the shooting method is to convert the boundary value problem into an initial value problem by guessing the unknown terminal conditions (typically the initial costate), integrating the system forward in time, and iteratively refining the guess until the terminal boundary conditions are satisfied.

This approach is particularly effective for problems with smooth solutions and relatively low dimensions. However, its success depends heavily on a good initial guess and the smoothness of the Hamiltonian system. The method may fail or converge slowly when the system is highly nonlinear, exhibits discontinuities, or is sensitive to initial conditions.

The following algorithm outlines the basic single shooting method:

Algorithm 1 Single Shooting Method for Solving PMP

1: **Input:** Initial state $x(0)$, terminal constraint $x(T)$, time horizon $[0, T]$
2: **Initialize:** Guess for initial costate $\lambda(0)$
3: **repeat**
4: Integrate the state and costate ODEs forward in time:

$$\dot{x}(t) = \frac{\partial H}{\partial \lambda}, \quad \dot{\lambda}(t) = -\frac{\partial H}{\partial x}$$

5: Evaluate mismatch at terminal time: $e = x(T) - x_{\text{target}}$
6: Update $\lambda(0)$ using a root-finding method (e.g., Newton–Raphson)
7: **until** $\|e\| < \epsilon$
8: **Output:** Optimal state $x(t)$, costate $\lambda(t)$, and control $u(t)$

2.3 Dynamic Programming Approach

Dynamic Programming (DP) is a powerful method in optimal control theory for solving problems that can be decomposed into smaller, overlapping subproblems. It is especially effective for systems exhibiting optimal substructure—where optimal solutions to subproblems can be composed into a global solution [25, 28].

Optimal Control Problem Formulation

Consider a general optimal control problem (OCP) with the goal of minimizing the cost functional:

$$J = \int_0^T L(x(t), u(t), t)\, dt + \Phi(x(T)) \tag{2.13}$$

subject to the system dynamics:

$$\dot{x}(t) = f(x(t), u(t), t) \tag{2.14}$$

where $x(t)$ is the system state, $u(t)$ the control input, L the running cost, and Φ the terminal cost.

Bellman's Principle of Optimality and the HJB Equation

DP is based on Bellman's Principle of Optimality: *"An optimal policy has the property that whatever the initial state and initial decisions are, the remaining decisions must constitute an optimal policy from that point onward."*
This principle leads to the Hamilton–Jacobi–Bellman (HJB) equation:

$$V(x) = \min_u \left[L(x, u, t) + \frac{\partial V}{\partial x} f(x, u, t) \right] \tag{2.15}$$

where $V(x)$ is the value function representing the minimal cost-to-go from state x.

Discretization of Time and State

In practical applications, solving the optimal control problem (OCP) analytically is often intractable due to nonlinearities, constraints, or the absence of closed-form solutions. Hence, a common approach is to discretize both the **time horizon** and the **state space**, transforming the continuous-time OCP into a discrete-time, grid-based problem that can be solved numerically using dynamic programming.

The discrete form of the value function becomes:

$$V(x_t) = \min_{u_t \in \mathcal{U}} [L(x_t, u_t) + V(x_{t+1})] \tag{2.16}$$

where the next state is computed as:

$$x_{t+1} = f(x_t, u_t, t)$$

This formulation enables a **backward recursion** starting from the terminal condition $V(x_T) = \Phi(x_T)$, allowing us to compute the optimal value function and corresponding control law for all possible discrete states.

The following algorithm outlines the basic backward dynamic programming recursion for a finite time horizon:

Algorithm 2 Dynamic Programming Backward Recursion

Input: Discretized time horizon $t = 0, 1, \ldots, N$
Input: Discretized state space $\mathcal{X}$ and control space $\mathcal{U}$
Initialize: Terminal cost $V_N(x) = \Phi(x)$ for all $x \in \mathcal{X}$
for $t = N - 1$ down to 0 **do**
    for each state $x \in \mathcal{X}$ **do**
        $V_t(x) \leftarrow \infty$
        for each control $u \in \mathcal{U}$ **do**
            Compute next state $x_{\text{next}} = f(x, u, t)$
            Compute cost $J = L(x, u) + V_{t+1}(x_{\text{next}})$
            if $J < V_t(x)$ **then**
                $V_t(x) \leftarrow J$
                Store optimal control $u^*(x, t) \leftarrow u$
            end if
        end for
    end for
end for
Output: Optimal value function $V_0(x)$ and control policy $u^*(x, t)$

Illustrative Example: Minimum-Energy Control of a Double Integrator

Consider the discrete-time version of a double integrator system, which is often used as a benchmark problem in optimal control literature. The state vector is $x_t = \begin{bmatrix} x_{1,t} \\ x_{2,t} \end{bmatrix}$, where $x_{1,t}$ is the position and $x_{2,t}$ is the velocity. The control input u_t is a bounded force applied to the system.

System Dynamics:

$$x_{t+1} = Ax_t + Bu_t, \quad \text{where} \quad A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \Delta t \end{bmatrix}$$

Objective: Minimize the total control effort over a finite horizon of N steps:

$$J = \sum_{t=0}^{N-1} u_t^2 \tag{2.17}$$

subject to: - initial condition $x_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, - terminal condition $x_N = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, - control constraint $u_t \in [-1, 1]$.

Discretization: We define a uniform time grid with step size $\Delta t = 0.1$, and discretize the state and control spaces uniformly.

DP Implementation: The cost-to-go function is initialized at the final time step:

$$V_N(x) = \begin{cases} 0, & \text{if } x = x_N \\ \infty, & \text{otherwise} \end{cases}$$

Then, for each $t = N - 1$ down to 0, we solve:

$$V_t(x_t) = \min_{u_t \in \mathcal{U}} \left[u_t^2 + V_{t+1}(Ax_t + Bu_t) \right] \tag{2.18}$$

This recursion yields the optimal control law that minimizes total energy while steering the system to the origin.

Interpretation: This example demonstrates how DP solves a constrained quadratic minimization problem with nonlinear cost-to-go propagation. Although the solution could be obtained analytically via Riccati recursion for the unconstrained case, DP provides flexibility in handling bounds on u_t and sparse discretized domains.

2.3.1 Parallelized Backward Dynamic Programming

To improve computational performance, especially when the state space is large, the inner loop of the dynamic programming recursion (over discrete states) can be **parallelized**, as each state update at a given time step is independent.

The parallelization is typically applied to the loop over all possible current states at a fixed time, distributing the workload across multiple processors or CPU cores using task-level parallelism. This is especially effective in environments such as MATLAB (with `parfor`) or Python (with multiprocessing).

The structure of the parallel implementation is shown below:

Algorithm 3 Parallelized Dynamic Programming Evaluation

Input: Time horizon $t = 0, 1, \ldots, N$; state space $\mathcal{X}$; control space $\mathcal{U}$
Initialize: $V_N(x) = \Phi(x)$ for all $x \in \mathcal{X}$
for $t = N - 1$ down to 0 **do**
    Parallel over each $x \in \mathcal{X}$ **do**
    $V_t(x) \leftarrow \infty$
    for each control $u \in \mathcal{U}$ **do**
        Compute next state $x_{\text{next}} = f(x, u, t)$
        Compute cost $J = L(x, u) + V_{t+1}(x_{\text{next}})$
        if $J < V_t(x)$ **then**
            $V_t(x) \leftarrow J$
            Store optimal control $u^*(x, t) \leftarrow u$
        end if
    end for
    End parallel block
end for
Output: Optimal value function $V_0(x)$ and control policy $u^*(x, t)$

This significantly reduces computation time while preserving correctness. Further optimizations included vectorization and memory preallocation.

Advantages and Limitations

Advantages:

- Guarantees global optimality (given complete discretization).
- Applicable to nonlinear systems and constrained dynamics.

Limitations:

- Suffers from the curse of dimensionality: computational cost increases exponentially with state/control dimensions.
- Accuracy depends on grid resolution; finer grids imply higher memory/time cost.

2.4 Conclusion

In conclusion, both Pontryagin's Minimum Principle (PMP) and Dynamic Programming (DP) are foundational tools in optimal control theory, each offering distinct advantages and limitations depending on the problem structure.

In summary, Pontryagin's Minimum Principle (PMP) is well-suited for optimal control problems with smooth dynamics and clearly defined boundary conditions. It transforms the problem into a two-point boundary value problem and provides valuable structural insights into the optimal control law. PMP is computationally efficient and ideal for low-dimensional systems, but it often yields only locally optimal solutions and can be sensitive to initial guesses, especially in the presence of constraints or discontinuities.

Dynamic Programming (DP), on the other hand, offers a more flexible and robust approach, capable of handling nonlinearities, hard constraints, and high control complexity. It guarantees global optimality by recursively solving the Hamilton–Jacobi–Bellman equation, but at the cost of high memory and computational demand due to the curse of dimensionality. The choice between PMP and DP depends on the problem's structure, dimensionality, and available computational resources.

Chapter 3

Optimal Control of e-Racing Vehicle

3.1 Introduction :

Introduction

This chapter presents the application of the optimal control methods discussed in Chapter 2—Pontryagin's Minimum Principle (PMP) and Dynamic Programming (DP)—to the energy management of an electric racing vehicle (EV). The control objective is to minimize the total power losses across the drivetrain components while ensuring efficient power distribution between the battery, supercapacitor, and electric motor. Alternatively, this objective is expressed as the maximization of the final total energy stored in the battery and supercapacitor, under the nonlinear dynamics already defined in the system model.

Based on the previously established state-space model, PMP is applied to derive the optimal control law in continuous time by solving a boundary value problem involving the state and costate trajectories. In parallel, DP is implemented in discrete time through backward recursion over the discretized state and time grids, ensuring global optimality. The two approaches are compared in terms of energy performance, computational effort, and applicability to real-time control. This comparison highlights the strengths and trade-offs of each method when applied to complex, nonlinear energy management systems in electric racing applications.

3.2 Simulation Setup and System Configuration :

3.2.1 Independent Subsystem Analysis :

Figure 3.1 shows an isolated subsystem from the complete energy system model. This subsystem receives two inputs :

- $i_{\text{ch\_bat}}$: Battery charging current, which is the control variable
- P_v: mechanical power, calculated as $P_v = T_e \cdot \omega_m$, which is a time-varying input.

with :

- T_e : Motor torque [Nm].
- ω_m : Motor angular velocity [rad/s].

The System returns two key State Variables (outputs) :

- SoC: State of charge of the battery, wich takes values between 0 and 1. (0 full depletion and 1 for full charge)
- U_{sc}: Voltage across the supercapacitor.

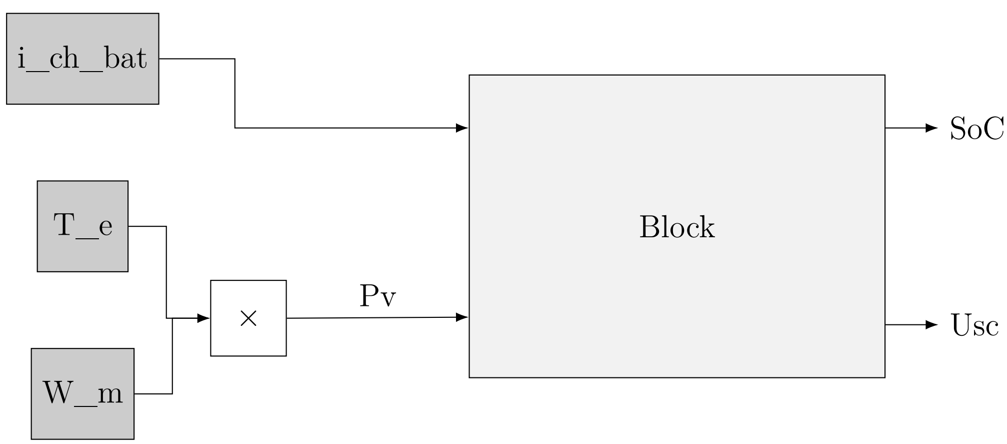


Figure 3.1: Isolated Subsystem with Inputs and State Outputs

3.2.2 Internal Dynamics And Functional Implementation :

The internal implementation of this subsystem (Block) is shown in Figure 3.2. It consists of a MATLAB Function block that computes the time derivatives of the system states.
These derivatives, $\dot{\text{SoC}}$ and $\dot{U}_{\text{sc}}$, are integrated to obtain the actual values of SoC and U_{sc}, the battery is charged 100% and the initial voltage of supercapasitor is 144 (V) which are also fed back into the function block to close the loop.

The function takes in the current SoC, supercapacitor voltage U_{sc}, battery charging current, and mechanical power, then outputs the time derivatives $\dot{\text{SoC}}$ and $\dot{U}_{\text{sc}}$:

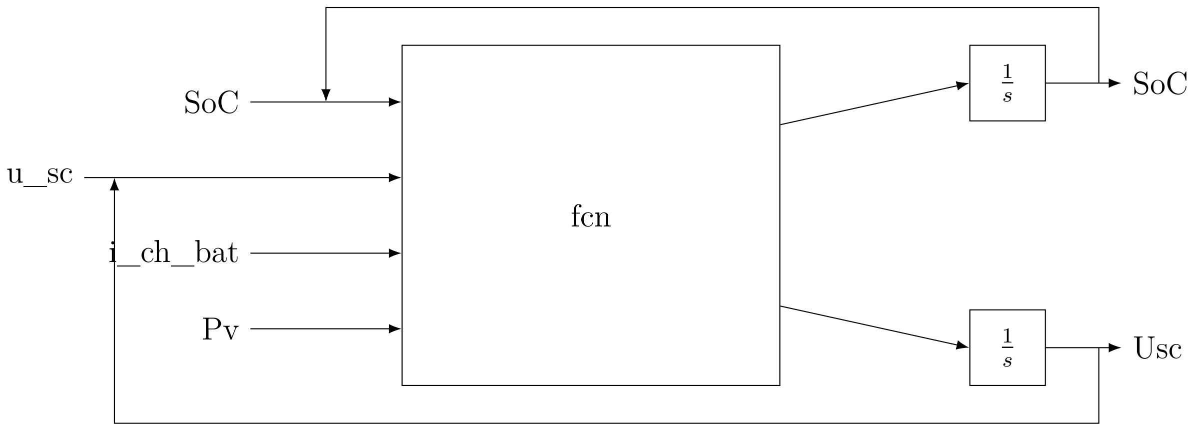


Figure 3.2: Internal Dynamics Computation Using a MATLAB Function

3.2.3 Battery Data and System

A) Battery Data

In battery modeling, the open-circuit voltage (OCV) is typically a nonlinear function of the state of charge (SoC). Because an analytical expression for this relationship is rarely available, the OCV–SoC characteristic is usually obtained from experimental data or manufacturer-provided discharge curves.

In this work, we use a look-up table approach in MATLAB, where the SoC and OCV values are stored as vectors. These vectors form a discrete representation of the OCV–SoC relationship and are used to interpolate the battery's open-circuit voltage during simulation.

```
SoC_vector = [0        0.0251    0.0685    0.1376    0.1686    0.2031    0.2403    0.2766    0.3076 ...
              0.3439    0.3829    0.4095    0.4476    0.4759    0.5140    0.5521    0.6114    0.6530 ...
              0.6893    0.7239    0.7584    0.7974    0.8399    0.8727    0.8891    0.9063    0.9294 ...
              0.9418    0.9612    0.9798    0.9984    1.0082    1.0170    1.0241    1.0281    1.0303 ...
              1.0316];


OCV_vector = n_bat_seri*[4.3151    4.2796    4.2263    4.1435    4.1095    4.0695    4.0311    3.9926    3.9660 ...
                         3.9334    3.8994    3.8757    3.8447    3.8240    3.8018    3.7825    3.7618    3.7500 ...
                         3.7411    3.7352    3.7278    3.7145    3.6953    3.6701    3.6538    3.6464    3.6257 ...
                         3.6050    3.5577    3.5059    3.4098    3.3269    3.2204    3.1021    2.9778    2.8506 ...
                         2.7589];

OCV_vector = flip(OCV_vector);

soe_vector     = [0     0.0190    0.0532    0.1099    0.1363    0.1668    0.2007    0.2345    0.2641    0.2992 ...
             0.3373    0.3634    0.4009    0.4289    0.4668    0.5049    0.5645    0.6064    0.6430    0.6780 ...
             0.7130    0.7527    0.7963    0.8300    0.8470    0.8649    0.8892    0.9023    0.9230    0.9430 ...
             0.9631    0.9738    0.9835    0.9914    0.9960    0.9985    1.0000];
```

Figure 3.3: Part of the script for computing internal battery dynamics

B) Defining the Data

The vectors `SoC_vector` and `OCV_vector` define discrete data points sampled from a battery's discharge profile:

- **SoC_vector:** Contains SoC values ranging from 0 (fully discharged) to 1 (fully charged).
- **OCV_vector:** Contains the corresponding open-circuit voltage values (scaled if necessary to the full battery voltage).

As shown in the script (Figure 3.3), the vector is reversed with:

```
OCV_Vector = flip(OCV_Vector);
```

This ensures that the SoC and OCV vectors are consistently ordered for interpolation.

C) Interpolation

To estimate the OCV at any arbitrary SoC value, linear interpolation is used:

`OCV = interp1(SoC_vector, OCV_vector, x(1))` where $x(1)$ represents the current SoC.

This approach allows accurate and efficient voltage estimation without the need for fitting complex nonlinear functions. The same interpolation method is also used to estimate the state of energy (SOE):

```
SOE = interp1(SoC_vector, soe_vector, x(1))
```

D) Numerical Derivation of dOCV/dSoC

In some control strategies, such as Pontryagin's Minimum Principle, the derivative of the OCV with respect to SoC is required. Since OCV is only available as discrete data, its derivative must be approximated numerically.

To do this, we compute the finite difference of the OCV vector with respect to the SoC vector:

```
dOCV_vector = diff(OCV_vector) ./ diff(SoC_vector);
```

This gives the numerical derivative $\frac{d\,\mathrm{OCV}}{d\,\mathrm{SoC}}$ evaluated between each pair of consecutive SoC points. To use this derivative in simulation or control, we interpolate it just like OCV:

```
dOCV = interp1(SoC_vector_mid, dOCV_vector, x(1));
```

Here, `SoC_vector_mid` is the midpoint of each interval used for differentiation. This technique allows accurate estimation of the OCV slope at any SoC value and enables the implementation of control laws that depend on the battery's voltage sensitivity to charge.

3.3 PMP Control of E-Racing Vehicle (EV)

State Dynamics

The system is described by a nonlinear state-space model composed of two state variables: the battery's state of charge (SoC) and the supercapacitor voltage U_{sc}. The control input is the reference charging current $i_{\text{ch\_bat}}$.

$$x = \begin{pmatrix} \text{SoC} \\ v_{sc} \end{pmatrix}, \quad u = \left(i_{\text{ch\_bat}}\right)$$

Battery SoC Dynamics:

$$\dot{\text{SoC}} = A \cdot \frac{u}{\text{OCV}(x_1) - r \cdot u}, \quad A = -\frac{U_{\text{dc}} \cdot \eta_{\text{bat}}}{3600 \cdot Q_{\text{rated}} \cdot n_{\text{bat\_par}}} \tag{3.1}$$

Supercapacitor Voltage Dynamics:

$$\dot{x}_2 = \frac{B_1 \cdot u}{x_2} + \frac{B_2}{x_2} \tag{3.2}$$

$$\text{with} \quad B_1 = \frac{n_{\text{sc}} \cdot U_{\text{dc}}}{C_{\text{rated}}}, \quad B_2 = -\frac{n_{\text{sc}} \cdot T_e \cdot W_m}{C_{\text{rated}}}$$

Parameters:

- r: Equivalent series resistance (battery + switches + interconnects)
- $\text{OCV}(x_1)$: Battery open-circuit voltage, function of SoC
- U_{dc}: DC bus voltage [V]
- η_{bat}, η_{sc}: Battery and supercapacitor efficiencies
- $n_{\text{bat\_par}}$: Number of parallel battery cells

Final-State Objective Function

The control objective is to maximize the total stored energy in the battery and supercapacitor at final time t_f, expressed as:

$$\mathcal{J} = E_{\text{bat}}(x_1(t_f)) + \frac{1}{2} C \cdot x_2(t_f)^2 \tag{3.3}$$

Here, $E_{\text{bat}}(x_1)$ is defined via a numerical look-up table as a function of SoC, and the supercapacitor energy is modeled analytically.

Because the final time is fixed and the final state is free, the transversality conditions yield the terminal values of the costates (adjoint variables):

$$\lambda_1(t_f) = \frac{dE_{\text{bat}}}{dx_1}, \tag{3.4}$$

$$\lambda_2(t_f) = \frac{d}{dx_2}\left(\frac{1}{2} C x_2^2\right) = C x_2(t_f) \tag{3.5}$$

3.3.1 Hamiltonian Function

The Hamiltonian is defined as:

$$\mathcal{H}(x, u, \lambda) = \lambda_1 \cdot \dot{x}_1 + \lambda_2 \cdot \dot{x}_2 \tag{3.6}$$

Substituting the system dynamics:

$$\mathcal{H}(x, u, \lambda) = \lambda_1 \cdot A \cdot \frac{u}{\mathrm{OCV}(x_1) - ru} + \lambda_2 \cdot \left(\frac{B_1 u + B_2}{x_2} \right) \tag{3.7}$$

Or equivalently

$$\mathcal{H}(x, u, \lambda) = \left(\frac{\lambda_1 A}{\mathrm{OCV}(x_1) - ru} + \frac{\lambda_2 B_1}{x_2} \right) u + \frac{\lambda_2 B_2}{x_2} \tag{3.8}$$

3.3.2 State-Costate System

The state and costate equations are coupled and described by:

$$\begin{cases} \dot{x}_1 = A \cdot \dfrac{u}{\mathrm{OCV}(x_1) - r \cdot u} \\ \dot{x}_2 = \dfrac{B_1 u + B_2}{x_2} \\ \dot{\lambda}_1 = \lambda_1 \cdot A \cdot \dfrac{u \cdot \frac{d\,\mathrm{OCV}(x_1)}{dx_1}}{(\mathrm{OCV}(x_1) - ru)^2} \\ \dot{\lambda}_2 = \lambda_2 \cdot \dfrac{-(B_1 u + B_2)}{x_2^2} \end{cases}$$

3.3.3 Control and State Constraints

In addition to the state and costate differential equations, the optimal control problem is subject to bounds on both the control input and the state variables, ensuring safe and physically realistic operation of the electric racing vehicle.

- **Control constraint:**

  $$u = i_{\mathrm{ch\_bat}} \in [-100, 100] \text{ A}$$

  This constraint ensures the charging/discharging current commanded to the battery remains within its rated operational limits.

- **State constraints:**

  $$\mathrm{SoC} \in [0.1, 1], \quad U_{\mathrm{sc}} \in [50, 144] \text{ V}$$

  These bounds enforce safe energy levels in both the battery and the supercapacitor. The minimum SoC of 0.1 prevents over-discharge, while the upper limit reflects full charge. Similarly, the supercapacitor voltage is constrained to remain within its permissible range.

These constraints must be respected throughout the optimization process. In the PMP framework, they are typically handled either via constraint-aware control law selection or through numerical clipping during the simulation of the optimal trajectory. Incorporating these bounds is essential to ensure that the resulting control policy is both optimal and implementable in practice.

Handling the OCV Derivative

To compute $\frac{d\,\mathrm{OCV}}{dx_1}$, we use a numerical derivative of the OCV–SoC map provided as experimental data. Given discrete vectors `SoC_vector` and `OCV_vector`, the derivative is estimated using:

```
dOCV_vector = diff(OCV_vector) ./ diff(SoC_vector);
SoC_mid = (SoC_vector(1:end-1) + SoC_vector(2:end)) / 2;
dOCV = interp1(SoC_mid, dOCV_vector, x(1));
```

This interpolated derivative $dOCV$ is used in $\dot{\lambda}_1$, as required by the PMP formulation. This step is critical to incorporate the nonlinearity of the battery OCV curve into the optimal control computation.

Simulation Parameters

Symbol	Description	Value
Q_{rated}	Battery capacity	2.1 Ah
U_{dc}	DC bus voltage	330 V
η_{bat}	Battery efficiency	0.995
η_{sc}	Supercapacitor efficiency	0.997
$n_{\text{bat\_par}}$	Number of battery cells in parallel	30
$n_{\text{bat\_seri}}$	Number of battery cells in series	60
C_{rated}	Supercapacitor capacitance	58 F
r	Equivalent system resistance	0.05 Ω

Table 3.1: Model parameters used in the PMP simulation

3.4 Simulation Results 1: Limitations of PMP in an Unconstrained Scenario

To evaluate the behavior of Pontryagin's Minimum Principle (PMP) in solving the energy management problem, we begin with a simplified configuration where all constraints are removed, and the power demand is zero (i.e., both speed and torque are set to zero). This allows us to isolate the intrinsic behavior of the optimal control structure predicted by PMP.

3.4.1 Shooting Method and Transversality Conditions

The two-point boundary value problem derived from the PMP formulation was solved using the shooting method. The initial state is specified, while the final state is free. Therefore, the transversality condition requires that the final costate values satisfy:

$$\lambda_1(t_f) = \frac{dE_{\text{bat}}}{dx_1}, \qquad \lambda_2(t_f) = Cx_2(t_f)$$

A root-finding algorithm was used to iteratively adjust the initial values of $\lambda_1(0)$ and $\lambda_2(0)$ such that these terminal conditions are met. Despite convergence of the costate trajectory to the correct terminal values, the resulting control policy consistently yielded the maximum admissible charging current $u = u_{\max}$ over the entire horizon.

3.4.2 Optimal Control Trajectory

Figure 3.4 shows the evolution of the optimal control $u(t)$ computed using PMP. As expected in the unconstrained and power-free scenario, the Hamiltonian minimization results in a bang-bang strategy. In this case, the control saturates at its upper limit:

$$u^*(t) = u_{\max}, \quad \forall t \in [0, T]$$

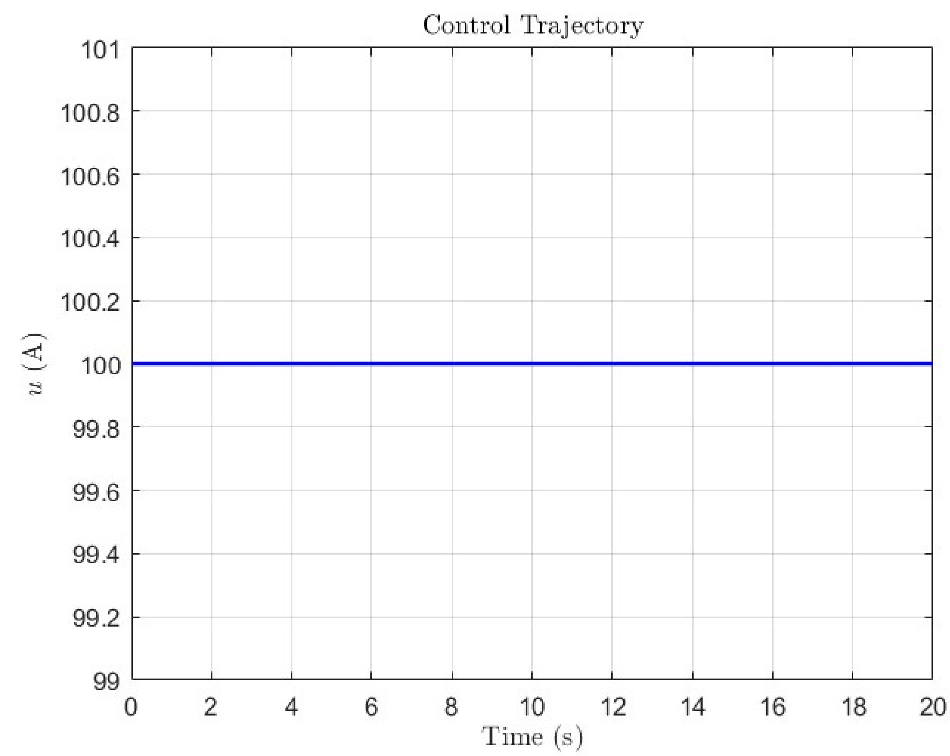
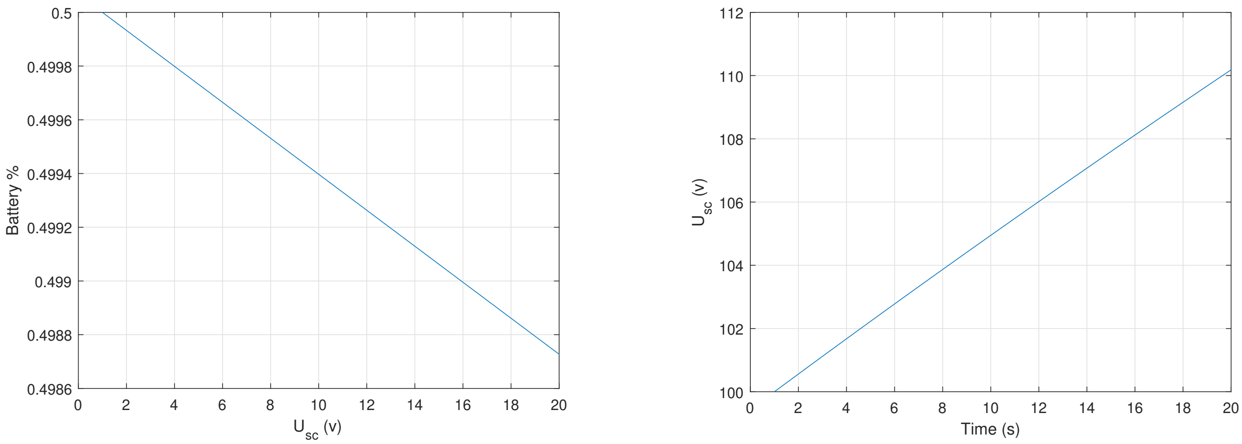


Figure 3.4: Control trajectory obtained using PMP in the unconstrained scenario.

3.4.3 State Evolution

Figure 3.5 shows the evolution of the state variables under the PMP solution: the battery's State of Charge (SoC) and the supercapacitor voltage U_{sc}. Due to the constant maximum control, Both SoC and the supercapacitor voltage follow a linear profiles, discharge for the battery, and a charge of the Supercapacitor.



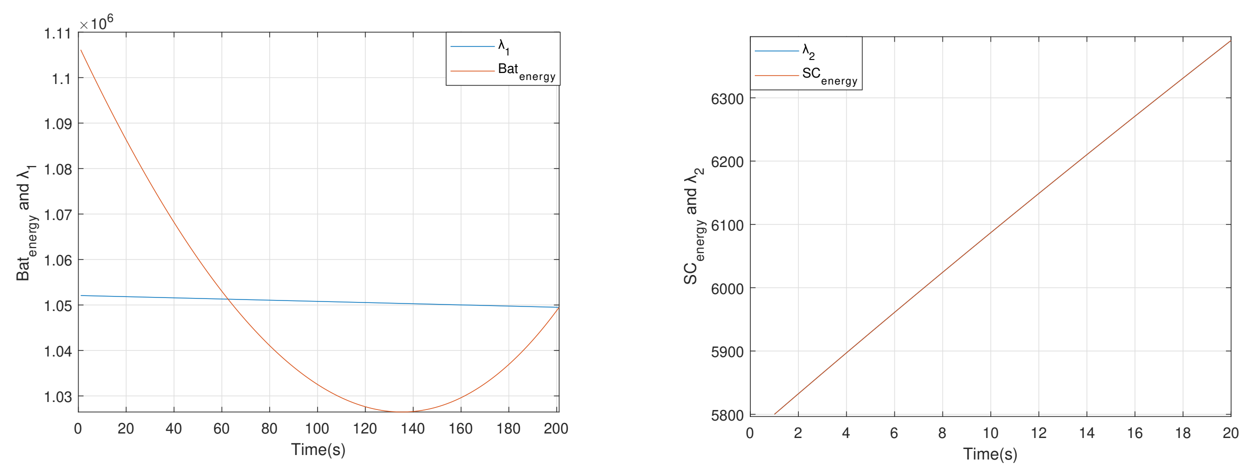
(a) SoC trajectory

(b) Supercapacitor voltage

Figure 3.5: State trajectories under PMP: SoC and U_{sc}.

3.4.4 Costate Evolution

Figure 3.6 displays the evolution of the costate variables $\lambda_1(t)$ and $\lambda_2(t)$. The plots show that the final values match the theoretical transversality conditions, confirming the correctness of the shooting method used to initialize the adjoint system.



(a) $\lambda_1(t)$ with terminal value $\lambda_1(t_f) = \frac{dE_{\text{bat}}}{dx_1}$

(b) $\lambda_2(t)$ with terminal value $\lambda_2(t_f) = Cx_2(t_f)$

Figure 3.6: Costate trajectories under PMP with verification of transversality conditions.

3.4.5 Interpretation of Results

Although the PMP-based solution satisfies the necessary conditions for optimality, the resulting control remains constant at $u = u_{\max}$ throughout the simulation. This outcome is a direct consequence of the Hamiltonian being linear in the control input, leading PMP to select a bang-bang solution. However, this result is not globally optimal in the context of this unconstrained, no-load scenario.

In fact, the truly optimal control in this trivial case is $u(t) = 0$ for all $t \in [0, T]$. Applying zero current avoids any energy transfer between the battery and supercapacitor, thereby maintaining the total stored energy constant over time. Any nonzero control, such as full charging or discharging, induces energy conversion through the powertrain components, which inherently involve resistive and conversion losses. As a result, energy is irreversibly dissipated, reducing the final stored energy—contrary to the control objective.

This highlights a key limitation of PMP: while it provides locally optimal control laws based on necessary conditions, it may fail to identify the true global optimum, especially in degenerate or low-gradient cases where the Hamiltonian structure promotes extreme control actions that are energetically inefficient.

3.5 Simulation Results 2: DP Results Across Representative Scenarios

This section presents the application of Dynamic Programming (DP) to the optimal energy management problem of the electric racing vehicle. Unlike PMP, which relies on local necessary conditions, DP computes a globally optimal solution over the discretized state and control spaces.

Three scenarios are studied:

- A trivial no-load case with zero torque and speed,
- The ARTEMIS standard drive cycle,
- The UDDS urban drive cycle.

For each case, the results are visualized through a figure containing three subplots: the optimal control input, battery SoC, and supercapacitor voltage U_{sc}. A dedicated interpretation is provided after each scenario.

3.5.1 Scenario 1: Trivial No-Load Case

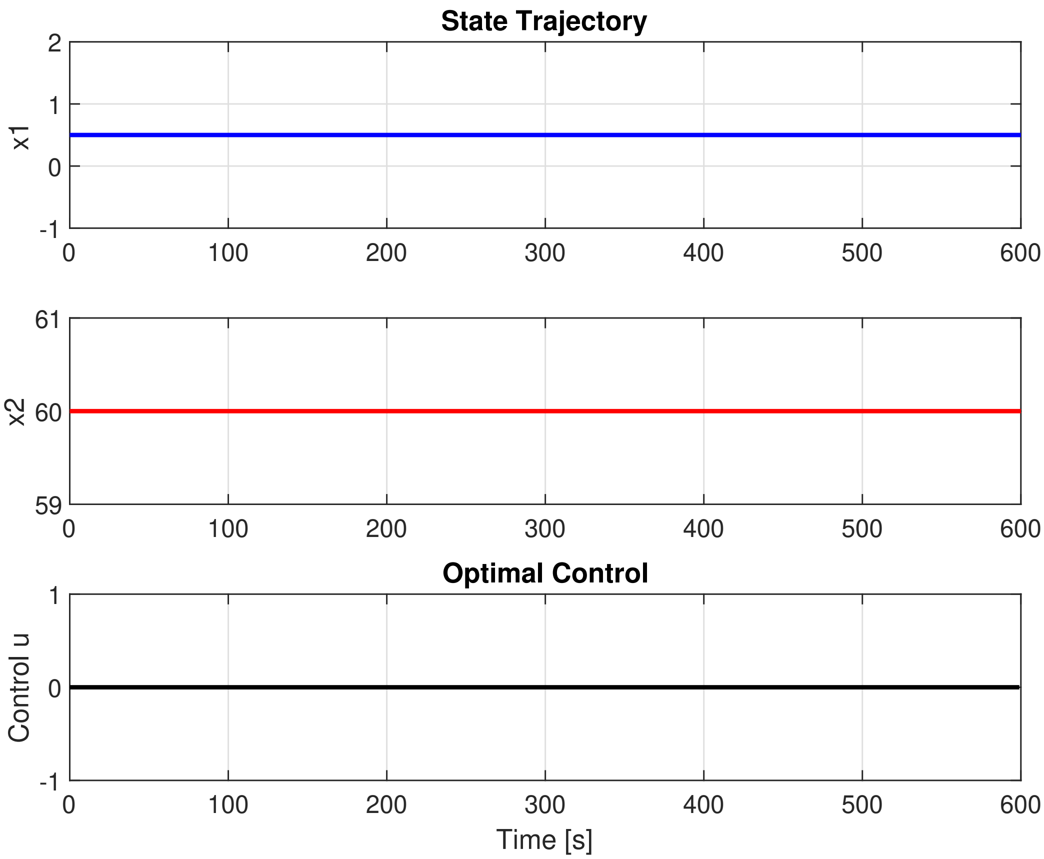


Figure 3.7: DP results for the trivial no-load scenario: (top) control input, (middle) battery SoC, (bottom) supercapacitor voltage.

Interpretation

As shown in Figure 3.7, DP identifies the globally optimal control strategy as keeping the control input at zero throughout the horizon. This maintains both the SoC and supercapacitor voltage constant, preserving the total stored energy. Any nonzero control would cause power transfer and introduce losses due to inefficiencies in the powertrain. This confirms that DP correctly avoids unnecessary energy dissipation, unlike the PMP solution which applied maximum charging and led to suboptimal outcomes.

3.5.2 Scenario 2: ARTEMIS Drive Cycle

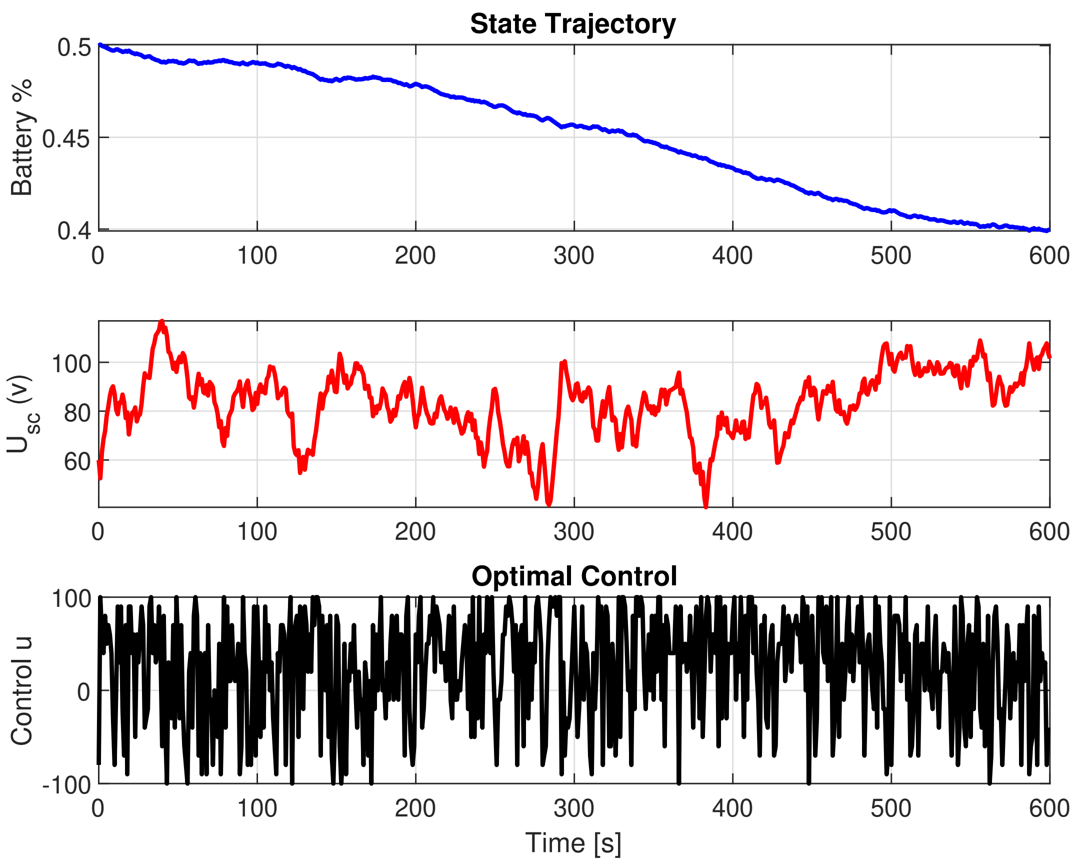


Figure 3.8: DP results under the ARTEMIS driving cycle: (top) control input, (middle) battery SoC, (bottom) supercapacitor voltage.

Interpretation

In the ARTEMIS scenario (Figure 3.8), the DP solution shows a smooth and non-aggressive control trajectory. The control input responds adaptively to power demand fluctuations while maintaining SoC and supercapacitor voltage within operational bounds. The battery is used efficiently during steady phases, while the supercapacitor handles short-term variations. This behavior aligns with the energy-saving goal and showcases DP's ability to manage hybrid storage systems effectively under realistic driving conditions.

3.5.3 Scenario 3: UDDS Cycle

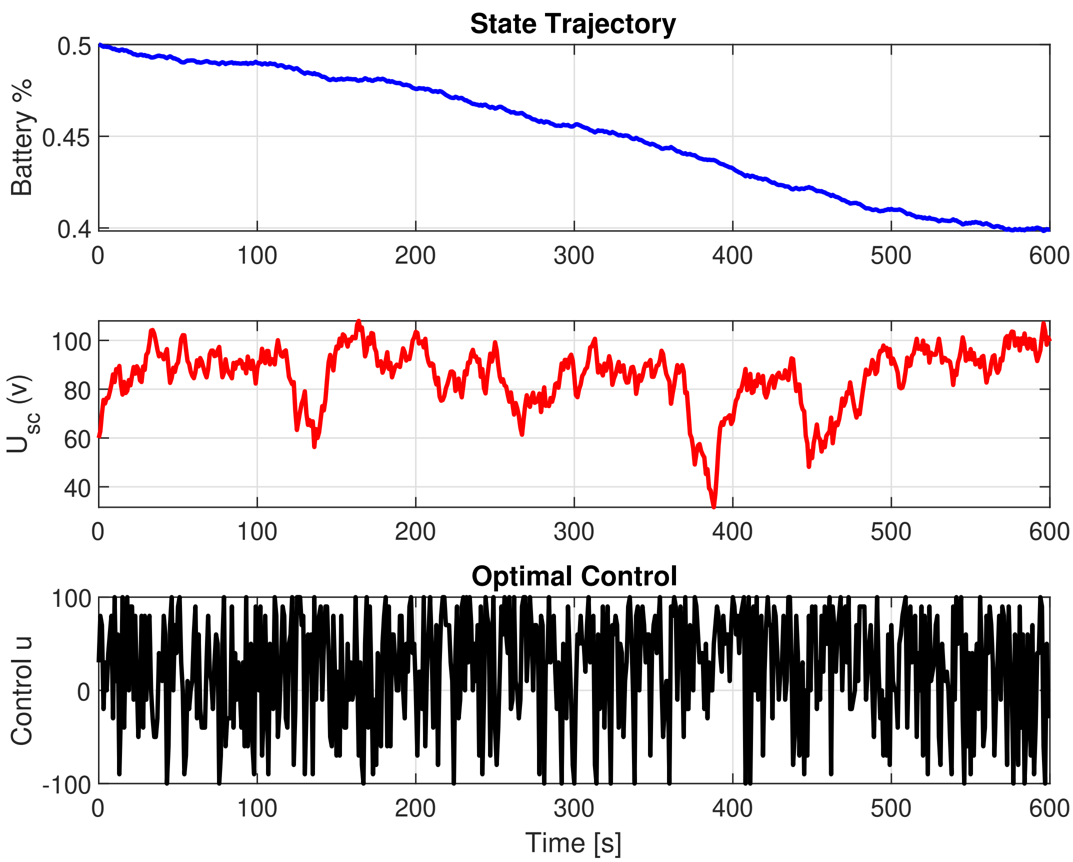


Figure 3.9: DP results under the UDDS driving cycle: (top) control input, (middle) battery SoC, (bottom) supercapacitor voltage.

Interpretation

The UDDS cycle (Figure 3.9) represents a more aggressive urban profile with frequent stops and sharp accelerations. The DP controller generates more dynamic control actions, relying heavily on the supercapacitor to absorb fast changes in power demand. The battery discharge is regulated more conservatively to extend its usable energy and reduce losses. This scenario further illustrates DP's flexibility in optimizing energy usage while maintaining constraints across highly variable profiles.

3.6 Conclusion :

Conclusion

This chapter presented the application of two optimal control strategies—Pontryagin's Minimum Principle (PMP) and Dynamic Programming (DP)—to the energy management of an electric racing vehicle. The control objective was to maximize the total stored energy at the final time by optimally managing the power flow between the battery, supercapacitor, and electric motor.

The PMP approach, though computationally efficient and analytically insightful, was shown to be limited in practice. In the trivial no-load scenario, PMP converged to a bang-bang solution that satisfied the necessary conditions but failed to achieve global optimality, highlighting its sensitivity to the Hamiltonian structure and lack of constraint handling.

In contrast, the DP method demonstrated robustness and flexibility. Across all test scenarios—trivial, ARTEMIS, and UDDS—it successfully generated globally optimal control policies that respected system constraints and adapted to varying load profiles. The results validated DP's superiority in handling complex, nonlinear systems, albeit at the cost of increased computational demand.

Overall, this comparative study confirmed that while PMP can be useful for low-dimensional or well-structured problems, DP remains the preferred choice for achieving global optimality in real-world, constrained energy management tasks.

General Conclusion :

General Conclusion

This thesis addressed the problem of optimal energy management in electric racing vehicles, focusing on the efficient use of a hybrid energy storage system composed of a battery and a supercapacitor. The study aimed to develop, analyze, and compare two optimal control strategies—Pontryagin's Minimum Principle (PMP) and Dynamic Programming (DP)—in terms of their ability to minimize powertrain losses and maximize final stored energy.

Chapter 1 established a detailed and realistic model of the EV powertrain, capturing the nonlinear dynamics and energy conversion processes of key components, including the battery, supercapacitor, inverter, and internal permanent magnet motor. The system model served as the foundation for subsequent control analyses.

Chapter 2 introduced the theoretical foundations of PMP and DP. Through simplified examples, the essential principles of each method were illustrated, including the derivation of the Hamiltonian, costate dynamics, transversality conditions, and recursive Bellman equations. This chapter provided the necessary mathematical tools and conceptual clarity to apply the two approaches effectively.

In Chapter 3, both PMP and DP were implemented on the modeled EV system under different scenarios. The simulation results highlighted the limitations of PMP, especially in unconstrained or degenerate cases where it failed to find the global optimum. In contrast, DP consistently produced globally optimal control policies, effectively managing energy flow in both simple and realistic driving cycles (ARTEMIS and UDDS). While PMP remains valuable for low-dimensional systems with smooth solutions, DP proved more robust and reliable for real-world energy management.

In conclusion, the work demonstrates the importance of selecting an appropriate control strategy based on the problem structure. It also underscores the practical challenges of optimal control in complex EV systems and provides a pathway for future research into more efficient numerical methods and real-time implementable control architectures.

Bibliography :

[1] C. Depature et al., “IEEE VTS Motor Vehicles Challenge 2017 – Energy management of a fuel cell/battery vehicle,” in *Proc. IEEE VPPC*, 2016, pp. 1–6.

[2] C. Depature et al., “IEEE VTS Motor Vehicles Challenge 2018 – Energy management of a range extender electric vehicle,” in *Proc. IEEE VPPC*, 2017, pp. 1–6.

[3] W. Lhomme et al., “IEEE VTS Motor Vehicles Challenge 2019 – Energy management of a dual-mode locomotive,” in *Proc. IEEE VPPC*, 2018, pp. 1–6.

[4] J. Solano et al., “IEEE VTS Motor Vehicles Challenge 2020 – Energy management of a fuel cell/ultracapacitor/lead-acid battery hybrid electric vehicle,” in *Proc. IEEE VPPC*, 2019, pp. 1–6.

[5] B.-H. Nguyen et al., “IEEE VTS Motor Vehicles Challenge 2021 – Energy management of a dual-motor all-wheel drive electric vehicle,” in *Proc. IEEE VPPC*, 2020, pp. 1–6.

[6] T. Vo-Duy et al., “IEEE VTS Motor Vehicles Challenge 2022 – Sizing and energy management of hybrid dual-energy storage system for a commercial electric vehicle,” in *Proc. IEEE VPPC*, 2021, pp. 1–6.

[7] K. Li, P. Evans, and M. Johnson, “SiC/GaN power semiconductor devices: A theoretical comparison and experimental evaluation under different switching conditions,” *IET Electr. Syst. Transp.*, vol. 8, no. 1, pp. 3–11, 2018.

[8] C. Liu et al., “A critical review of advanced electric machines and control strategies for electric vehicles,” *Proc. IEEE*, vol. 109, no. 6, pp. 1004–1028, 2021.

[9] T. Vo-Duy and M. C. Ta, “Fundamental design of electric motor control systems,” in *Encyclopedia of Electrical and Electronic Power Engineering*, J. García, Ed. Oxford: Elsevier, 2023, pp. 428–453.

[10] T. D. V. Thang et al., “Wide-bandgap power semiconductors for electric vehicle systems: Challenges and trends,” *IEEE Veh. Technol. Mag.*, vol. 16, no. 4, pp. 2–11, 2021.

[11] K. H. Nam, *AC Motor Control and Electrical Vehicle Applications*, 2nd ed. Taylor & Francis, 2019.

[12] A. Bouscayrol et al., “Energetic macroscopic representation and inversion-based control illustrated on a wind-energy-conversion system using hardware-in-the-loop simulation,” *IEEE Trans. Ind. Electron.*, vol. 56, no. 12, pp. 4826–4835, 2009.

[13] A. Jacome et al., "A benchmark of different starting modes of a passive fuel cell/ultracapacitor hybrid source for an electric vehicle application," *J. Energy Storage*, vol. 35, p. 102280, 2021.

[14] H.-L. T. Nguyen et al., "A comparative study of adaptive filtering strategies for hybrid energy storage systems in electric vehicles," *Energies*, vol. 14, no. 12, 2021.

[15] D. Ramsey et al., "Comparison of energy recovery solutions on a suburban DC railway system," *IEEE Trans. Transp. Electrific.*, vol. 7, no. 3, pp. 1849–1857, 2021.

[16] C. Mayet et al., "Comparison of different EMR-based models of traction power substations for energetic studies of subway lines," *IEEE Trans. Veh. Technol.*, vol. 65, no. 3, pp. 1021–1029, 2016.

[17] C. Mayet et al., "EMR-based simulation tool of a multi-train subway system," in *Proc. IEEE VPPC*, 2016, pp. 1–6.

[18] K. S. Agbli et al., "EMR modeling of a hydrogen-based electrical energy storage," *Eur. Phys. J. Appl. Phys.*, vol. 54, no. 2, p. 23404, 2011.

[19] R. Krishnan, *Permanent Magnet Synchronous and Brushless DC Motor Drives*. CRC Press, 2010.

[20] J. Pyrhönen, T. Jokinen, and V. Hrabovcova, *Design of Rotating Electrical Machines*. Wiley, 2008.

[21] U.S. Environmental Protection Agency, "Federal Test Procedure (FTP): Urban Dynamometer Driving Schedule (UDDS)," [Online]. Available: https://www.epa.gov/vehicle-and-fuel-emissions-testing/dynamometer-driving-schedules.

[22] United Nations Economic Commission for Europe (UNECE), "Worldwide harmonized Light vehicles Test Procedure (WLTP)," [Online]. Available: https://unece.org/transport/vehicle-regulations/wltp.

[23] C. André and J. Joumard, "Modelling of vehicle emissions based on the ARTEMIS European driving cycles," INRETS, Lyon, France, Tech. Rep. LTE0511, 2005.

[24] U.S. EPA, UNECE, and INRETS, "Standard driving cycles for vehicle testing: UDDS, WLTC, and ARTEMIS," Technical Summary from EPA, UNECE Regulations, and INRETS Report LTE0511, 2005–2023.

[25] A. Sciarretta, L. Guzzella, and M. Back, "A Real-Time Optimal Control Strategy for Parallel Hybrid Vehicles With CVT," *IEEE Transactions on Control Systems Technology*, vol. 9, no. 3, pp. 460–473, 2001.

[26] A. E. Bryson and Y.-C. Ho, *Applied Optimal Control: Optimization, Estimation and Control*. Hemisphere Publishing, 1975.

[27] P. Elbert, B. Schürmann, and L. Guzzella, “Pontryagin’s Minimum Principle for the Control of Energy Storage Systems in Hybrid Electric Vehicles,” *Oil & Gas Science and Technology – Rev. IFP Energies nouvelles*, vol. 68, no. 1, pp. 109–118, 2013.

[28] J. Rousseau, D. Di Domenico, and T. Vo-Duy, “Optimization-Based Energy Management for Hybrid Electric Vehicles: Pontryagin’s Minimum Principle and Real-Time Algorithms,” in *Handbook of Intelligent Vehicles*, Springer, 2012, pp. 1261–1284.

[29] R. E. Bellman, *Dynamic Programming*. Princeton University Press, 1957.