

A COMPARATIVE ANALYSIS OF ENERGY  
MANAGEMENT STRATEGIES FOR HYBRID  
ELECTRIC VEHICLES

DISSERTATION

Presented in Partial Fulfillment of the Requirements for  
the Degree Doctor of Philosophy in the  
Graduate School of The Ohio State University

By

Lorenzo Serrao, M.S.

\* \* \* \* \*

The Ohio State University

2009

Dissertation Committee:

Giorgio Rizzoni, Adviser

Yann G. Guezennec

Steve Yurkovich

Junmin Wang

© Copyright by  
Lorenzo Serrao  
2009

## **Abstract**

The dissertation offers an overview of the energy management problem in hybrid electric vehicles. Several control strategies described in literature are presented and formalized in a coherent framework. A detailed vehicle model used for energy flow analysis and vehicle performance simulation is presented. Three of the strategies (dynamic programming, Pontryagin's minimum principle, and equivalent consumption minimization strategy, also known as ECMS) are analyzed in detail and compared from a theoretical point of view, showing the underlying similarities. Simulation results are also provided to demonstrate the application of the strategies.

To the people who care about me, and look after me.  
To the people who admire me, and look up to me.  
To the people who loved me, and now look at me from the sky.  
To my family.

## ACKNOWLEDGMENTS

I am deeply grateful to my advisor, prof. Giorgio Rizzoni, for the guidance and support during the past four years, and for being of example in both academic and personal life. I feel incredibly fortunate for the opportunity to work with him. I also feel fortunate for having met during my school years many people that had a profound impact on my life: my elementary school teacher, Francesca Messineo, who gave an injection of confidence to a shy kid; my math teacher in middle school, Gino Strano, who was the first to make me appreciate the beauty of mathematics and science; my Italian and Latin teacher in high school, Elio D'Agostino, who made me a rational person and transmitted me his love for knowledge; and my master's thesis advisor, prof. Mauro Velardocchia of Politecnico di Torino, who has always believed in me, and without whom I would not even be here.

I wish to thank my committee members for the suggestions and ideas they provided, as well as Prof. Vadim Utkin, whose help in the initial phase of this dissertation was extremely important to let me understand Pontryagin's minimum principle. I am honored for working with him.

I am greatly indebted to two people from whom I learned a lot: Chris Hubert, who taught me most of what I know about modeling, and has always been there to answer my questions; and CG Cantemir, who is always happy to share his all-around engineering knowledge.

I am also grateful to the many bright students and researchers I met at CAR, for the interesting discussions about hybrids, batteries, or just cars... and I would like to say thanks to all my friends for their presence in my life, especially important when one lives far away from home. From the moment they came to pick me up at the airport the first time I came to Columbus (remember, Marcello?), and

then during all the great times that followed, I have always enjoyed my stay in Columbus thanks to the wonderful company.

And wonderful has been sharing the office with my officemate and dear friend Simona Onori. It is difficult to express in words how thankful I am for the great ideas, precious help and guidance she offered me during the writing of this dissertation. Not to mention her moral support during the tough days before the defense...

Finally, I would like to thank my family for being always so close to me despite the geographical distance. I could not have made it without their support. I feel they deserve a Ph.D. as well, since I talked about it so much!

## VITA

November 2, 1978 ..... Born - Taurianova, Italy

2003 ..... M.S. Mechanical Engineering, Politecnico di Torino (Italy)

2005-present ..... Graduate Research Associate, Center for Automotive Research at the Ohio State University.

## PUBLICATIONS

### Research Publications

G. Rizzoni, J. Josephson, A. Soliman, C. Hubert, C. Cantemir, N. Dembski, P. Pisu, D. Mikesell, L. Serrao, and J. Russell. Modeling, simulation, and concept design for hybrid-electric medium-size military trucks. *Proceedings of SPIE*, 5805, 2005.

L. Serrao, Z. Chehab, Y. Guezennec, and G. Rizzoni. An aging model of Ni-MH batteries for hybrid electric vehicles. *Proceedings of the 2005 IEEE Vehicle Power and Propulsion Conference (VPP05)*, pages 78–85, 2005.

C. Cantemir, G. Ursescu, L. Serrao, G. Rizzoni, J. Bechtel, T. Udvardy, and M. Letherwood. Concept design of a new generation military vehicle. *Proceedings of SPIE*, 6201, 2006.

C.-G. Cantemir, G. Ursescu, L. Serrao, G. Rizzoni, J. Bechtel, T. Udvardy, and M. Letherwood. Island concept evt. *SAE Paper 06FFL-250*, 2006.

Z. Chehab, L. Serrao, Y. Guezennec, and G. Rizzoni. Aging characterization of nickel – metal hydride batteries using electrochemical impedance spectroscopy. *Proceedings of the 2006 ASME International Mechanical Engineering Congress and Exposition*, 2006.

P. Pisu, C. Cantemir, N. Dembski, G. Rizzoni, L. Serrao, J. Josephson, and J. Russell. Evaluation of powertrain solutions for future tactical truck vehicle systems. *Proceedings of SPIE*, 6228, 2006.

P. Pisu, L. Serrao, C. Cantemir, and G. Rizzoni. Hybrid-electric powertrain design evaluation for future tactical truck vehicle systems. *Proceedings of the 2006 ASME International Mechanical Engineering Congress and Exposition*, 2006.

L. Serrao, P. Pisu, and G. Rizzoni. Analysis and evaluation of a two engine configuration in a series hybrid electric vehicle. *Proceedings of the 2006 ASME International Mechanical Engineering Congress and Exposition*, 2006.

T. Donateo, L. Serrao, and G. Rizzoni. Multi-objective optimization of a heavy duty hybrid electric vehicle. *Workshop on Electric, Hybrid and Solar Vehicles, University of Salerno (Italy)*, 2007.

L. Serrao, C. Hubert, and G. Rizzoni. Dynamic modeling of heavy-duty hybrid electric vehicles. *Proceedings of the 2007 ASME International Mechanical Engineering Congress and Exposition*, 2007.

T. Donateo, L. Serrao, and G. Rizzoni. A two-step optimisation method for the preliminary design of a hybrid electric vehicle. *International Journal of Electric and Hybrid Vehicles*, 1(2), 2008.

L. Serrao and G. Rizzoni. Optimal control of power split for a hybrid electric refuse vehicle. *Proceedings of the 2008 American Control Conference*, 2008.

L. Serrao, S. Onori and G. Rizzoni. Equivalent Consumption Minimization Strategy as a realization of Pontryagin's minimum principle for HEV control. *Proceedings of the 2009 American Control Conference*, 2009.

L. Serrao, S. Onori, G. Rizzoni and Y. Guezennec. A Model Based Strategy for Estimation of the Residual Life of Automotive Batteries. *2009 IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes (SAFEPROCESS 09)*, 2009.

## FIELDS OF STUDY

Major Field: Mechanical Engineering

# Contents

	<b>Page</b>
Abstract . . . . .	ii
Dedication . . . . .	iii
Acknowledgments . . . . .	iv
Vita . . . . .	vi
List of Tables . . . . .	xii
List of Figures . . . . .	xiii
List of Symbols . . . . .	xxi
Chapters:	
1. Introduction . . . . .	1
1.1 Motivation . . . . .	1
1.2 Introduction to hybrid electric vehicles . . . . .	2
1.2.1 The importance of driving cycles . . . . .	7
1.3 The energy management problem in HEVs . . . . .	9
1.3.1 Numerical global optimization . . . . .	12
1.3.2 Analytical optimal control techniques . . . . .	13
1.3.3 Instantaneous optimization . . . . .	14
1.3.4 Heuristic control techniques . . . . .	15
1.4 Powertrain modeling for energy management . . . . .	16
1.5 Organization and contributions of the dissertation . . . . .	16
1.6 Case studies . . . . .	18
1.6.1 Case study 1: AHHPS project and experimental vehicle . . . . .	18
1.6.2 Case study 2: EcoCAR Challenge . . . . .	21

2.	Powertrain modeling . . . . .	22
2.1	An energetic approach to hybrid electric vehicles . . . . .	22
2.1.1	Vehicle energy balance . . . . .	23
2.1.2	Powertrain losses . . . . .	26
2.1.3	Modeling approaches . . . . .	27
2.1.4	Approach used in this work . . . . .	30
2.2	Physical modeling tools . . . . .	31
2.3	Simulator implementation . . . . .	38
2.4	Powertrain components . . . . .	39
2.4.1	Internal combustion engine . . . . .	39
2.4.2	Torque converter . . . . .	43
2.4.3	Gearings and differential . . . . .	47
2.4.4	Gearbox and transmission . . . . .	48
2.4.5	Wheels, brakes, and tires . . . . .	48
2.4.6	Electric machines . . . . .	52
2.4.7	Energy storage systems . . . . .	53
2.4.8	Batteries . . . . .	53
2.4.9	Capacitors . . . . .	57
2.4.10	Power electronics and electric bus . . . . .	58
2.4.11	Engine accessories . . . . .	59
2.4.12	Auxiliary loads . . . . .	59
2.4.13	Vehicle dynamics . . . . .	61
2.4.14	Driver . . . . .	64
2.5	Model validation: refuse collection vehicle . . . . .	65
2.5.1	Vehicle dynamics and road load . . . . .	68
2.5.2	Supercapacitors . . . . .	68
2.5.3	Traction motors . . . . .	71
2.5.4	Engine, generator and accessories . . . . .	73
2.6	Conclusion . . . . .	76
3.	Optimal control strategies for hybrid vehicles . . . . .	77
3.1	Energy management of hybrid electric vehicles . . . . .	77
3.1.1	Definition of the optimal control problem for hybrid vehicles	79
3.1.2	Classification of energy management strategies . . . . .	82
3.2	Pontryagin's minimum principle . . . . .	83
3.2.1	Minimum principle for problems with no constraints on the state . . . . .	84
3.2.2	Minimum principle for problems with constraints on the state . . . . .	85
3.2.3	Notes on the minimum principle . . . . .	86
3.3	Dynamic programming . . . . .	87

3.3.1	General concepts . . . . .	87
3.3.2	Application to HEVs . . . . .	90
3.4	Stochastic Dynamic Programming . . . . .	94
3.4.1	Generalities . . . . .	94
3.4.2	Markov chains . . . . .	95
3.4.3	Stochastic dynamic programming for HEV energy management . . . . .	98
3.4.4	Stochastic driving cycle models . . . . .	100
3.4.5	Problem formulation . . . . .	101
3.5	Equivalent Consumption Minimization Strategy . . . . .	103
3.5.1	Basic concepts . . . . .	103
3.5.2	Equivalence factor and charge-sustainability . . . . .	105
3.5.3	Adaptive ECMS . . . . .	107
3.5.4	Implementation issues . . . . .	109
3.5.5	Chattering issues . . . . .	111
3.6	From the minimum Principle to the ECMS . . . . .	111
3.7	Model predictive control . . . . .	114
3.7.1	Overview . . . . .	114
3.7.2	Receding Horizon Technique . . . . .	114
3.7.3	Application to energy management of HEVs . . . . .	117
3.8	Rule-based control strategies . . . . .	118
3.9	Implementation issues common to all energy management strategies . . . . .	121
3.9.1	Stability . . . . .	121
3.9.2	Interface with low-level controllers . . . . .	121
3.9.3	Power demand . . . . .	123
3.9.4	Regenerative braking . . . . .	124
3.9.5	Actuator bandwidth . . . . .	125
3.9.6	Battery characterization . . . . .	125
3.10	Conclusion . . . . .	125
4.	Application of the strategies . . . . .	126
4.1	General problem formulation . . . . .	127
4.2	Definition of the control problem for refuse collection truck (case study 1) . . . . .	128
4.2.1	Power flow diagram . . . . .	129
4.2.2	System dynamics . . . . .	132
4.2.3	State constraints . . . . .	137
4.2.4	Control constraints . . . . .	137
4.2.5	Fuel consumption . . . . .	139
4.3	Definition of the control problem for case study 2 . . . . .	148
4.3.1	Power flow diagram . . . . .	150

4.3.2	System dynamics . . . . .	152
4.3.3	State constraints . . . . .	152
4.3.4	Control constraints . . . . .	155
4.3.5	Fuel consumption . . . . .	155
4.4	Parallel between the two case studies . . . . .	155
4.5	Simulation setup . . . . .	156
4.6	Driving cycles . . . . .	157
4.6.1	Refuse truck . . . . .	158
4.6.2	EcoCAR . . . . .	161
4.6.3	Fuel consumption correction . . . . .	164
4.7	Dynamic programming . . . . .	164
4.7.1	State discretization . . . . .	166
4.7.2	Arc cost and cost-to-go . . . . .	167
4.7.3	Implementation issues . . . . .	169
4.7.4	Simulation results, refuse truck . . . . .	171
4.7.5	Simulation results, EcoCAR, charge-sustaining . . . . .	173
4.8	Pontryagin's minimum principle . . . . .	176
4.8.1	Simulation results, refuse truck . . . . .	182
4.8.2	Simulation results, EcoCAR, charge-sustaining . . . . .	187
4.9	ECMS . . . . .	187
4.9.1	Basic formulation . . . . .	187
4.9.2	Admissible control set . . . . .	191
4.9.3	Fuel consumption . . . . .	192
4.9.4	Penalty function and difference between charge-sustaining and charge-depleting case . . . . .	192
4.9.5	Equivalence factors . . . . .	193
4.9.6	Simulation results, refuse truck . . . . .	194
4.9.7	Simulation results, EcoCAR, charge-sustaining . . . . .	197
4.10	Strategy comparison . . . . .	201
4.11	ECMS as an implementable quasi-optimal strategy: i-ECMS . . . . .	207
4.12	On the stability of ECMS and PMP . . . . .	210
5.	Conclusion . . . . .	217
	Bibliography . . . . .	221

## List of Tables

Table	Page
1.1 Vehicle weight classification based on GVWR (Gross vehicle weight rating, i.e. the maximum allowable total weight of a road vehicle when loaded - i.e including the weight of the vehicle itself plus fuel, passengers, cargo, and trailer tongue weight. . . . .	19
4.1 Supercapacitor characteristics . . . . .	133
4.2 Fitting coefficients for the Willans line model of the engine (4.39) (power in kW, speed in rad/s) . . . . .	147
4.3 Fitting coefficients for the Willans line model of the generator . . . .	147
4.4 Fitting coefficients for the engine fuel consumption along maximum efficiency line ( $\dot{m}_f = m_0 + m_1 P_{ice}$ , with $P_{ice}$ in W and $\dot{m}_f$ in g/s) . . .	147
4.5 EcoCAR battery characteristics . . . . .	154
4.6 Fuel consumption and best equivalence factors for case study 1 . . .	195
4.7 Fuel consumption and best equivalence factors for case study 2 . . .	203
4.8 Fuel consumption for the three strategies, case study EcoCAR. All values are normalized with respect to the DP solution . . . . .	205
4.9 Fuel consumption for the three strategies, case study refuse truck. All values are normalized with respect to the DP solution . . . . .	205

## List of Figures

Figure	Page
1.1 Typical architectures of hybrid electric vehicles. The arrows indicate admissible power flow. . . . .	6
1.2 Velocity profiles of the U.S. regulatory cycles [1] . . . . .	10
1.3 Example of integration of models with different accuracy . . . . .	17
2.1 Information flow in a forward simulator [2] . . . . .	28
2.2 Information flow in a backward simulator [2] . . . . .	29
2.3 Elementary model of a spring . . . . .	34
2.4 An example of a mechanical system whose structure changes with time . . . . .	35
2.5 Implementation of the model of the system shown in Figure 2.4 using Simulink . . . . .	37
2.6 Implementation of the model of the system shown in Figure 2.4 using SimDriveline . . . . .	37
2.7 Engine model . . . . .	40
2.8 Schematic representation of a torque converter . . . . .	44
2.9 Torque converter model . . . . .	45
2.10 Torque converter map . . . . .	46
2.11 Wheel and tire model . . . . .	49

2.12	Electric machine model . . . . .	53
2.13	General model of energy storage system . . . . .	54
2.14	Battery circuit model . . . . .	56
2.15	Open circuit voltage vs. state of charge for a Ni-MH battery [3]. The top and bottom curves correspond to charge and discharge (at 0.1C), the middle one is an average. . . . .	56
2.16	Circuit model of supercapacitor pack. . . . .	59
2.17	Hydraulic pump model . . . . .	60
2.18	Efficiency map of the hydraulic pump (Eaton 062 ADU [4]) . . . . .	61
2.19	Flow speed characteristic of the hydraulic pump [4] . . . . .	61
2.20	Standby torque curves [4] . . . . .	62
2.21	Architecture of the series hybrid electric prototype . . . . .	66
2.22	The five driving cycles used for model validation . . . . .	67
2.23	Powertrain architecture with model causality . . . . .	69
2.24	Scheme of the vehicle dynamics validation procedure . . . . .	70
2.25	Comparison between calculated and measured vehicle speed ( <i>Approach</i> cycle). The agreement between simulation and experiment is apparently good, except for the acceleration around 800 s, where the vehicle speed is very high and the model appears to slightly underestimate the resistances. . . . .	70
2.26	Comparison between calculated and measured vehicle speed ( <i>Approach</i> cycle, detail) . . . . .	71
2.27	Validation scheme of capacitor model . . . . .	71

2.28	Comparison between calculated and measured voltage at the terminals of the supercapacitor pack (cycle <i>Approach</i> ). The agreement is very good, despite the use of a simple first-order circuit model for the capacitors. The parameters of the circuit model (capacity, resistance) have been optimized using curve fitting of experimental voltage measurements (obviously, the one shown here is a validation data set; the calibration was done using a different experiment) . . .	72
2.29	Validation scheme of electric machine model . . . . .	73
2.30	Comparison between calculated and measured electric power at one traction motor (detail of cycle <i>Route 2</i> ). This curve represents the good quality of the efficiency map of the machine, since the electric power is computed directly using (2.50), based on torque and speed measurements. . . . .	74
2.31	Validation scheme of engine model . . . . .	74
2.32	Comparison between calculated and experimental engine torque (detail of <i>Route 2</i> ). This torque is the result of modeling the PTO accessories, the secondary accessories, and the generator. The models need to be validated together because there is no measurement of the torque request for each component. Considering this limitation, the agreement between experiment and simulation is acceptable. . .	75
2.33	Comparison of engine operating points on efficiency map (cycle <i>Route 2</i> ) . . . . .	75
3.1	The role of energy management control in a hybrid electric vehicle .	78
3.2	Shortest path problem [5, 6] . . . . .	89
3.3	Dynamic programming in HEVs: sequence of feasible power splits .	91
3.4	Example of application of dynamic programming to a hybrid electric vehicle . . . . .	92
3.5	Simple example of a Markov chain . . . . .	96
3.6	Energy path during charge and discharge in a parallel HEV [7] . . . .	104
3.7	The two factors in the ECMS correction term [8] . . . . .	106

3.8	Control diagram of adaptive ECMS with online optimization [9] . . .	108
3.9	Control diagram of adaptive ECMS with driving pattern recognition [10] . . . . .	109
3.10	Examples of reference trajectory, actual trajectory, and control sequence . . . . .	115
3.11	Basic structure of MPC . . . . .	116
3.12	Applying MPC is like driving a car: the drivers decide what to do predicting the consequences of their actions [11]. . . . .	117
3.13	An example of rule-based control [12] . . . . .	120
3.14	Relation between power demand and accelerator pedal position in a conventional vehicle . . . . .	124
4.1	Series hybrid electric architecture . . . . .	130
4.2	Circuit model of supercapacitor pack. . . . .	134
4.3	Charge-effectiveness factor for the ultracapacitor pack of the refuse truck . . . . .	136
4.4	Willans line fit of the generator map (constant $a_0, a_1$ ) . . . . .	141
4.5	Willans line fit of the engine map in the power-speed plane (left figure: speed dependency of the maps is neglected; right: speed dependency is considered) . . . . .	142
4.6	Willans line fit of the engine map in the torque-speed plane (left figure: speed dependency of the maps is neglected; right: speed dependency is considered) . . . . .	142
4.7	Engine fuel consumption map with optimal operating line . . . . .	144
4.8	Engine efficiency map with optimal operating line and iso-power lines	145
4.9	Engine fuel consumption along maximum efficiency line . . . . .	145
4.10	Powertrain architecture for case study 2 . . . . .	151

4.11	Power flow diagram for case study 2 . . . . .	151
4.12	Charge-effectiveness factor for the battery pack of the EcoCAR . . . . .	153
4.13	Open circuit voltage of the battery as a function of state of charge (data referred to a single cell) . . . . .	154
4.14	Fuel consumption of the genset as a function of the electrical power output . . . . .	156
4.15	Velocity profile and power requests of cycle <i>Approach</i> . . . . .	159
4.16	Velocity profile and power requests of cycle <i>Route 1</i> . . . . .	159
4.17	Velocity profile and power requests of cycle <i>Route 2</i> . . . . .	160
4.18	Velocity profile and power requests of cycle <i>Route 3</i> . . . . .	160
4.19	Velocity profile and power requests of cycle <i>Return</i> . . . . .	161
4.20	Velocity profile and power requests of cycle <i>UDDS</i> . . . . .	162
4.21	Velocity profile and power requests of cycle <i>US 06</i> . . . . .	162
4.22	Velocity profile and power requests of cycle <i>FTP highway</i> . . . . .	163
4.23	Velocity profile and power requests of cycle <i>FTP highway</i> . . . . .	163
4.24	SOE discretization for dynamic programming . . . . .	167
4.25	Example of optimal SOE sequence . . . . .	168
4.26	Dynamic programming solution for refuse truck, cycle <i>Approach</i> . . . . .	173
4.27	Dynamic programming solution for refuse truck, cycle <i>Approach</i> (de- tail of the first 180 s) . . . . .	174
4.28	Energy flow diagram corresponding to the dynamic programming solution for refuse truck, cycle <i>Approach</i> , compared to the results obtained in the same cycle by the conventional version of the same vehicle. . . . .	175
4.29	Dynamic programming solution for EcoCAR, cycle <i>UDDS</i> . . . . .	176

4.30	Dynamic programming solution for EcoCAR, cycle <i>UDDS</i> (detail of the first 180 s) . . . . .	177
4.31	Derivative of $\varepsilon_{cap}(\zeta, P_{cap})$ with respect to $\zeta$ , for case study 1 . . . . .	179
4.32	Derivative of $\varepsilon_{batt}(\zeta, P_{batt})$ with respect to $\zeta$ , for case study 2 . . . . .	180
4.33	Flow chart of iterative solution for Pontryagin's minimum principle .	181
4.34	Application of the minimum principle to the refuse truck: effect of $\lambda_0$ on fuel consumption (cycle <i>Approach</i> ) . . . . .	183
4.35	Application of the minimum principle to the refuse truck: effect of $\lambda_0$ on SOE profile (cycle <i>Approach</i> ) . . . . .	184
4.36	Application of the minimum principle to the refuse truck: effect of $\lambda_0$ on power split (cycle <i>Approach</i> ) . . . . .	185
4.37	Visualization of the Hamiltonian at one instant ( $t = 628$ s in driving cycle <i>Approach</i> ) . . . . .	186
4.38	Application of the minimum principle to the EcoCAR: effect of $\lambda_0$ on fuel consumption (cycle <i>UDDS</i> ) . . . . .	188
4.39	Application of the minimum principle to the EcoCAR: effect of $\lambda_0$ on SOE variation, defined as $\Delta SOE = SOE(t_f) - SOE(t_0)$ (cycle <i>UDDS</i> )	188
4.40	Application of the minimum principle to the EcoCAR: effect of $\lambda_0$ on SOE profile (cycle <i>UDDS</i> ) . . . . .	189
4.41	Application of the minimum principle to the EcoCAR: effect of $\lambda_0$ on power split (cycle <i>UDDS</i> ) . . . . .	190
4.42	Representation of (4.82) in the case in which the penalty is symmetrical, i.e. $\zeta_{ref} = \frac{1}{2}(\zeta_{min} + \zeta_{max})$ and $n_{\zeta 1} = n_{\zeta 2} = n_{\zeta}$ . . . . .	193
4.43	Application of ECMS to the refuse truck: SOE profiles obtained for four combinations of equivalence factors (cycle <i>Approach</i> ) . . . . .	195

4.44	Effect of equivalence factors on fuel consumption for the Refuse truck, cycle <i>Approach</i> . Fuel consumption values are corrected including the effect of SOE variation, and normalized with respect to the optimal solution obtained with dynamic programming. The points correspond to the cases shown in Figure 4.43 . . . . .	196
4.45	Effect of equivalence factors on SOE variation (positive values indicate an increasing trend, negative values a decreasing one) (cycle <i>Approach</i> ) . . . . .	196
4.46	Velocity profile and power requests of <i>UDDS</i> cycle . . . . .	198
4.47	SOE profile in <i>UDDS</i> cycle, for various combinations of equivalence factors (each letter represent a pair $s_{dis}, s_{chg}$ and the respective values are shown in Figure 4.48) . . . . .	199
4.48	Effect of equivalence factors on fuel consumption (fuel consumption is corrected to account for SOE variation and the result is normalized with respect to the optimal solution obtained with dynamic programming) . . . . .	200
4.49	Effect of equivalence factors on SOE variation (positive values indicate an increasing trend, negative values a decreasing one) . . . . .	200
4.50	Results of ECMS implementation on <i>UDDS</i> cycle ( $s_{dis} = 1.75, s_{chg} = 4.25$ ) . . . . .	201
4.51	Results of ECMS implementation on <i>UDDS</i> cycle, detail of Figure 4.50	202
4.52	The terms of the ECMS instantaneous cost function at one instant . . . . .	202
4.53	Comparison of the SOE profile obtained with the three strategies for the EcoCAR case, cycle <i>UDDS</i> . . . . .	204
4.54	Comparison of the RESS power obtained with the three strategies for the EcoCAR case, cycle <i>UDDS</i> , detail #1. . . . .	204
4.55	Comparison of the RESS power obtained with the three strategies for the EcoCAR case, cycle <i>UDDS</i> , detail #2. . . . .	205
4.56	Comparison of the SOE profile obtained with the three strategies for the EcoCAR case, cycle <i>UDDS</i> . . . . .	209

4.57	Comparison of the RESS power obtained with the three strategies for the EcoCAR case, cycle UDDS, detail #1. . . . .	209
4.58	Comparison of the RESS power obtained with the three strategies for the EcoCAR case, cycle UDDS, detail #2. . . . .	210
4.59	Behavior of ECMS under a constant power request $P_{req} = 30$ kW . . .	215
4.60	ECMS cost function for the three cases of Figure 4.59 . . . . .	216

## List of Symbols

*	(Relative to the optimal solution)
$a$	Distance of CG from front axle
$b$	Distance of CG from rear axle
$A_f$	Frontal area of the vehicle
$\alpha$	Accelerator pedal position (normalized)
$B_p$	Piston bore (diameter)
$\beta$	Brake pedal position (normalized)
BSFC	Brake specific fuel consumption
$C$	Electrical capacitance
$C_d$	Coefficient of aerodynamic drag
$c_m$	Mean piston speed
CG	Center of gravity
$E$	Energy
$E_{aero}$	Energy dissipated in aerodynamic resistance
$E_{kin}$	Kinetic energy
$E_{pot}$	Potential energy
$E_{roll}$	Energy dissipated in rolling resistance

$E_{trac}$	Energy delivered at the wheels by the powertrain
$\varepsilon$	Charge-effectiveness of energy storage device
$\varepsilon_{ress}$	Charge-effectiveness of RESS
$\eta$	Efficiency
$F$	Force
$F_{aero}$	Aerodynamic resistance
$F_{grade}$	Grade force (due to slope)
$F_{inertia}$	Inertial force (due to acceleration)
$F_{roll}$	Rolling resistance
$F_{trac}$	Total tractive force at the wheel-road interface
$G(x, t)$	State constraints
$H(\cdot)$	Hamiltonian function
$h_{CG}$	Height of the center of gravity
$I$	Current
$J$	Cost function of optimal control problem
$J$	Mass moment of inertia
$K$	Capacity factor (in torque converter)
$k$	Time index
$L$	Instantaneous cost
$\lambda$	Co-state variable
$\dot{m}_{elec}$	Instantaneous virtual fuel consumption corresponding to the use of electrical power

$\dot{m}_{eqv}$	Instantaneous equivalent fuel consumption
$\dot{m}_f$	Instantaneous fuel consumption (fuel mass flow rate)
$M_{veh}$	Vehicle mass
$MR$	Multiplication ratio or torque ratio (in torque converter)
$\omega$	Rotational speed
$\omega_{em}$	Rotational speed of the electric machine <i>em</i>
$\omega_{ice}$	Rotational speed of the internal combustion engine
$\Omega_x$	Set of admissible states
$P$	Power
$P_{acc}$	Mechanical power for secondary accessories
$P_{gen,e}$	Electrical power at the generator
$P_{gen,m}$	Mechanical power at the generator
$P_{hyd}$	Hydraulic power
$p_{hyd}$	Hydraulic pressure
$P_{ice}$	Mechanical power generated by the internal combustion engine
$p_{ma}$	Mean available pressure
$p_{me}$	Mean effective pressure
$P_{pto}$	Mechanical power for PTO (power take off) accessories
$\phi(x_f, t_f)$	Terminal cost of optimal control problem
$\pi$	Control policy (in dynamic programming)
$\psi(x_f, t_f)$	Terminal constraints of optimal control problem

$Q_{batt}$	Charge stored in a battery [Ah]
$Q_{cap}$	Charge capacity of a capacitor
$Q_{hyd}$	Hydraulic flow
$Q_{lhr}$	Fuel lower heating value
$Q_{ress}$	Charge capacity of a generic RESS
$R$	Electrical resistance
$r_0$	Rolling resistance coefficient
$r_1$	Rolling resistance coefficient
RESS	Rechargeable energy storage system
$\rho_{air}$	Air density
RPM	Rotational speed expressed in revolutions per minute
$s_p$	Piston stroke
SOC, $\xi$	State of charge
SOE, $\zeta$	State of energy
SR	Speed ratio (in torque converter)
$T$	Torque
$t$	Time
$t_0$	Initial time of the optimization horizon
$t_f$	Final time of the optimization horizon
$\theta$	Temperature
$\mathcal{U}$	Admissible control set

$u$	Control variable(s)
$V$	Voltage
$V_{circ}$	Voltage drop in a circuit
$V_d$	Engine displacement
$V_L$	Load voltage
$V_{oc}$	Open circuit voltage
$V_{veh}$	Vehicle speed
$\zeta, SOC$	State of charge
$x$	Dynamic state(s) of a system
$\zeta, SOE$	State of energy

# Chapter 1

## INTRODUCTION

### 1.1 Motivation

Increasing concerns about environmental issues, such as global warming and greenhouse gas emissions, as well as the predicted scarcity of oil supplies (and geopolitical issues related to oil suppliers) have made energy efficiency and reduced emissions a primary selling point for automobiles, and a concern for many governments. Because of this, hybrid electric vehicles have become extremely popular (even though not very widespread) and represent an icon of “good” technology. Indeed, hybrid electric vehicles offer some important advantages in comparison to conventional (engine-only) vehicles, despite additional components, greater complexity and increased cost: hybridization can reduce fuel consumption by very significant percentages, and can also help reducing polluting emissions. Hybrid vehicles derive part of their advantages from the fact that the total power request is split among the fuel and the electrical energy buffer: this fact poses some interesting challenges from the control standpoint, which this dissertation attempts to formalize and describe thoroughly.

Plug-in hybrid vehicles, i.e. hybrid electric vehicles with oversized batteries that can also be recharged using electric power from the grid, have recently become a hot topic in the automotive community, on both the industrial and the academic side, for the undoubted advantages in terms of emissions and fuel consumption deriving from the possibility to be driven for a relatively extended driving range using only electricity. However, these vehicles present some additional challenges

for control and optimization, due to the necessity of accounting for the cost, energy depletion and pollution due to the use of electrical energy in place of fuel.

This dissertation provides methodologies for modeling energy flow and fuel consumption in hybrid vehicles (Chapter 2) and presents an analytical formalization of the optimal control problem associated with the repartition of power request between the energy sources on board of the vehicle, as well as an organic review of the techniques proposed in the literature (Chapter 3). A detailed analysis of some of these techniques is proposed in Chapter 4, with application to two case studies of practical relevance, including a heavy-duty vehicle and a plug-in SUV.

## **1.2 Introduction to hybrid electric vehicles**

A brief overview is given here, focusing on the energy flow characteristics of hybrid vehicles. A more complete introduction to hybrid electric vehicles can be found in textbooks [13, 14] and lecture notes [15]. Several other theses and dissertations [16, 17, 18, 19, 20, 21, 22, 23] also deal with the problem and provide details about some of the aspects that are not studied in detail in this work, such as mechanical design, drivability issues etc.

To understand why hybrid electric vehicles are beneficial from the efficiency point of view, it is necessary to think about the way in which fuel (i.e., energy) is used in a vehicle. This is done in detail in Chapter 2, but a preliminary analysis is useful. Consider the case of a conventional vehicle: the combustion engine, which converts the chemical energy in the fuel to mechanical energy, generates all the power needed during a trip. The mechanical power generated by the engine is used for moving all driveline components, driving accessories (power steering, alternator, air conditioning...), and, of course, moving the vehicle. Given the driver's input (accelerator and brake pedals) and the driving conditions (speed, road surface, etc.), the mechanical power that the engine must deliver is determined. In a hybrid electric vehicle, instead, the total power demand is satisfied by summing together the outputs of the engine (thermal path, or fuel path) and of the battery or other storage device (electric path). The ratio of the power flows generated by

each path can be chosen freely<sup>1</sup> and constitutes a degree of freedom that allows to change the operating conditions of the engine with respect to the conventional case, thus giving the potential to increase its average efficiency.

The force needed to propel a vehicle along a given route is the sum of rolling resistance, aerodynamic resistance, grade (road slope) and inertia force (acceleration):

$$F_{trac} = F_{roll} + F_{aero} + F_{grade} + F_{inertia} \quad (1.1)$$

While the first two terms are dissipative (always tend to slow down the vehicle), the grade and inertia represent conservative forces, whose effect is to modify respectively the potential and kinetic energy of the vehicle. This means that some energy is stored in the vehicle when its speed or altitude are increased using energy coming from the engine. On the other hand, when the vehicle is decelerating to stop, or is being driven downhill, its energy content is decreasing, and some kinetic and/or potential energy must be dissipated: this is usually done by using the mechanical brakes, in addition to the rolling and aerodynamic resistances, which are always present and tend to slow down the vehicle (dissipating some of the energy). It can be concluded that, during a trip, the vehicle energy is dissipated only through the rolling resistance, the aerodynamic resistance, and the brakes.

An idea that makes electric vehicles very attractive is the fact that the electric motors driving the wheels are reversible and can produce negative torque. This means that they can replace the mechanical brakes as a mean to decelerate the vehicle, with the benefit of acting like generators and producing electrical energy, which is then stored in batteries on board of the vehicle for later use. This operation, known as regenerative braking, in principle could recover all the energy that is normally dissipated in the conventional brakes. In practice, only a fraction of it can be regenerated: in part for the limitations of the machines (in terms of peak power or torque), in part for their efficiency (the electrical power generated is smaller than the mechanical power entering the machine). Even with these limits,

<sup>1</sup>within some constraints, as shown later.

regenerative braking provides a significant improvement in the overall vehicle efficiency. The devices needed for implementing regenerative braking are reversible motors and reversible energy storage systems, which can store and deliver energy on command. These can be of any kind: electrical (batteries, capacitors), hydraulic (pressure accumulators), or mechanical (flywheels, springs). The system fuel tank + internal combustion engine is not reversible, because there is no way to generate fuel from the mechanical power entering the engine; however, its energy density is much higher than all known reversible energy storage systems and this is the reason why all mainstream vehicles have always used fuel as energy source.

*Hybrid* vehicles are so defined because they have two energy sources, complementing each other: in *hybrid electric* vehicles, the two sources are the fuel tank and an electrical energy storage system, usually batteries (but supercapacitors are possible as well). The idea is to have a high-capacity (i.e., high energy-density) source coupled with a reversible one, for storing energy coming from regenerative braking. Other kinds of hybrid vehicles are possible, that couple the engine to a hydraulic accumulator (hydraulic energy storage) or to a flywheel (mechanical energy storage), but only hybrid electric vehicles have reached the mass market.

Once the rechargeable energy source is added to the engine, it can be used for more than just regenerative braking: in fact, it can act as an energy buffer for the engine, which can instantaneously deliver an amount of power different than what is required by the vehicle load. This flexibility in engine management results in the ability to operate the engine more often in conditions where it is more efficient, or less polluting.

Other benefits offered by hybridization are the possibility to shut down the engine when it is not needed (such as at a stop or at very low speed), and the downsizing of the engine: since the peak power can be reached by summing the output from the engine and from the electrical storage, the former can be downsized (which typically implies higher efficiency).

Several kinds of hybrid electric vehicles have been conceived, usually distinguished by their architecture, which is related to the path that the power flow follows from the energy sources to the wheels. They are (see Figure 1.1):

- *series* hybrid electric vehicles, in which the engine drives a generator whose electrical power output is summed to the power coming from the electrical storage, then transmitted, via an electric bus, to the electric motors driving the wheels;
- *parallel* hybrid electric vehicles, in which the power summation is mechanical rather than electrical: the engine and the electric machines (one or more) are connected with a gear set, a chain, or a belt, so that their torque is summed and then transmitted to the wheels using a conventional driveshaft and possibly a differential;
- *power-split* hybrids, in which two electric machines can either add or subtract torque at the engine shaft; the vehicle thus behaves as a parallel or series hybrid, depending on the control actions;
- *series/parallel* hybrids, in which the engagement/disengagement of one or two clutches allows to change the powertrain configuration from series to parallel and vice-versa, thus allowing the use of the configuration best suited to the current operating conditions.

The series architecture (Figure 1.1.a) has the advantage of presenting only electrical connections between the main power transformation devices. This simplifies vehicle packaging and design, since each component can be placed independently from the others. Also, having the engine completely disconnected from the wheels gives great freedom in choosing its load and speed, thus making it operate at the highest possible efficiency. On the other hand, there are always two energy conversions (mechanical to electrical in the generator, and electrical to mechanical in the motor), which introduce losses, even in cases when a mechanical connection of the engine to the wheels would actually be overall more efficient. For this reason, there are conditions in which a series hybrid vehicle consumes more fuel than its conventional counterpart: for example, highway driving.

The parallel architecture (Figure 1.1.b) does not have this problem; however, unless significantly over-designed, the electric motors are less powerful than in

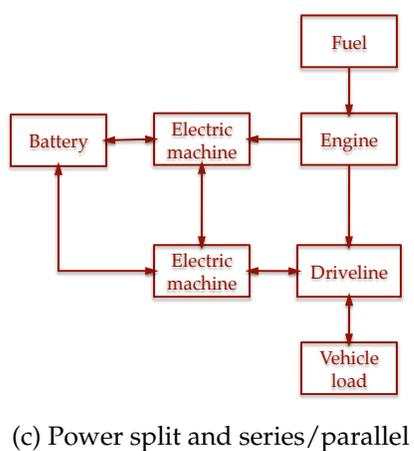
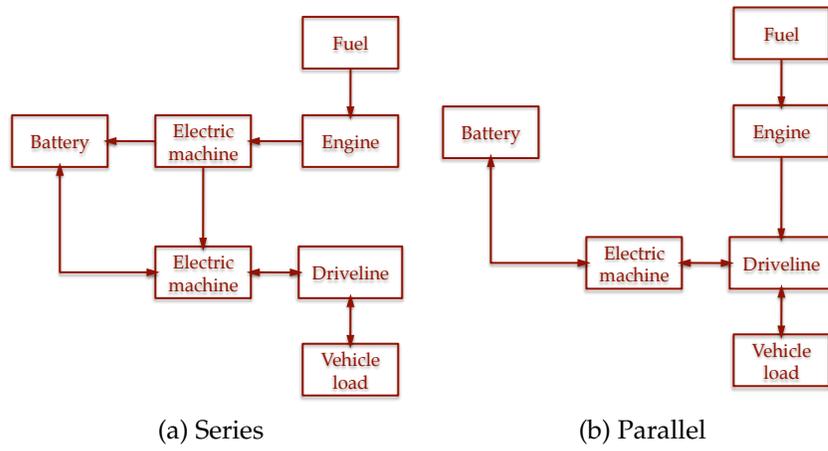


Figure 1.1: Typical architectures of hybrid electric vehicles. The arrows indicate admissible power flow.

a series hybrid (because not all the mechanical power flows through them), thus reducing the potential for regenerative braking; also, the engine operating conditions cannot be determined as freely as in a series hybrid architecture, being its speed related (via the transmission) to the vehicle velocity.

Power split or series/parallel architectures (which can be realized in different ways, but are in general characterized by the power flow shown in Figure 1.1.c) are the most flexible, and give a higher degree of control of the operating conditions of the engine than the parallel architecture. A typical embodiment of a power split architecture uses a planetary gear set with three shafts: one connected to the engine, the others to two electric machines. Depending on the relative speeds of the three shafts, the speed and torque ratios can change and power can flow from the engine to either of the two electric machines.

### 1.2.1 The importance of driving cycles

As implied in the previous section, the advantages of hybrid vehicles depend on how the vehicle is used. In particular, the hybridization advantages consist essentially in recovering potential and kinetic energy that would otherwise be dissipated in the brakes, and in operating the engine in its highest-efficiency region. If the engine had a constant efficiency and the vehicle drove at constant speed on a flat road, there would be no advantage in a hybrid electric configuration. The characteristics of the driving cycle will be considered in this section, while a discussion on the engine characteristics will be presented later.

A *driving cycle* represents the way the vehicle is driven during a trip, and the road characteristics. In the simplest case, it is defined as a sequence of vehicle speed (and therefore acceleration) and road grade. Together with some vehicle characteristics, this completely defines the road load, i.e., the force that the vehicle needs to exchange with the road during the driving cycle. The road load is, in fact, the sum of several terms:

- inertia, i.e. force needed to accelerate the vehicle;
- grade force, needed to overcome the slope of the road;

- rolling resistance, due to tire/road interaction, bearing losses etc.;
- aerodynamic drag.

It is important to point out that each term is a function of both the driving cycle (speed, acceleration, grade) and the vehicle (mass, frontal area, coefficients of aerodynamic and rolling resistance). For this reason, the fuel consumption of a vehicle must always be specified in reference to a specific driving cycle. On the other hand, given a driving cycle, the absolute value of the road load and also the *relative* magnitude of its components depend on the vehicle characteristics.

The necessity for a standard method to evaluate fuel consumption of all vehicles on the market, and to provide a reliable basis for their comparison, led to the introduction of a small number of regulatory driving cycles: any vehicle sold in a country has to be tested, according to detailed procedures, using one or more of these standard cycles.

In the US, the EPA (Environmental Protection Agency) is responsible for defining the regulatory driving cycles and the rules according to which the vehicles should be tested [24]. Several regulatory cycles are defined by EPA [1], and shown in Figure 1.2:

- the Urban Dynamometer Driving Schedule (UDDS) represents city driving conditions, and is used for light duty vehicle testing;
- the Federal Test Procedure (FTP), also called EPA75, is composed of the UDDS followed by the first 505 seconds of the UDDS;
- the Highway Fuel Economy Driving Schedule (HWFET) represents highway driving conditions under 60 mph;
- the New York City Cycle (NYCC) features low speed stop-and-go traffic conditions;
- the US06 cycle is a high acceleration aggressive driving schedule that is often identified as the “Supplemental FTP” driving schedule;

- the Heavy Duty Urban Dynamometer Driving Schedule (H-UDDS) is for heavy duty vehicle testing.

These driving cycles are designed to be representative of urban and extra-urban driving conditions, and reproduce measures of vehicle speed in real roads. Some of them and the test procedures have been recently updated to better suit modern vehicles, following criticism towards the previous regulation. In fact, because of acceleration levels far below the capabilities of any modern car and no use of air conditioning (now ubiquitous), the official values of fuel consumption obtained by testing vehicles according to previous EPA standards were much lower than in real-world driving conditions. The situation is not different in Japan and in Europe, where the regulatory cycles are synthetic, and represent rather optimistic approximations of real driving conditions.

Even with the current improvements, the regulatory cycles should be considered a comparison tool rather than a prediction tool. In fact, it is not possible to predict how a vehicle will be driven, since each vehicle has a different usage pattern and each driver his or her own driving style. In order to obtain more realistic estimations of real-world fuel consumption for a specific vehicle, vehicle manufacturers may develop their own testing cycles. In the case of hybrid vehicles, estimating the actual driving cycles becomes an even more important task, because the actual fuel consumption is affected by the supervisory control strategy implemented, which is tuned using simulations based on the estimated driving cycles, as shown in Chapter 4.

### **1.3 The energy management problem in HEVs**

In general, the main reason for using a hybrid electric architecture is the additional degree of freedom due to the presence of an additional energy source besides the fuel tank; this implies that, at each instant of time, the power needed by the vehicle can be provided by either one of these sources, or by a combination of the two.

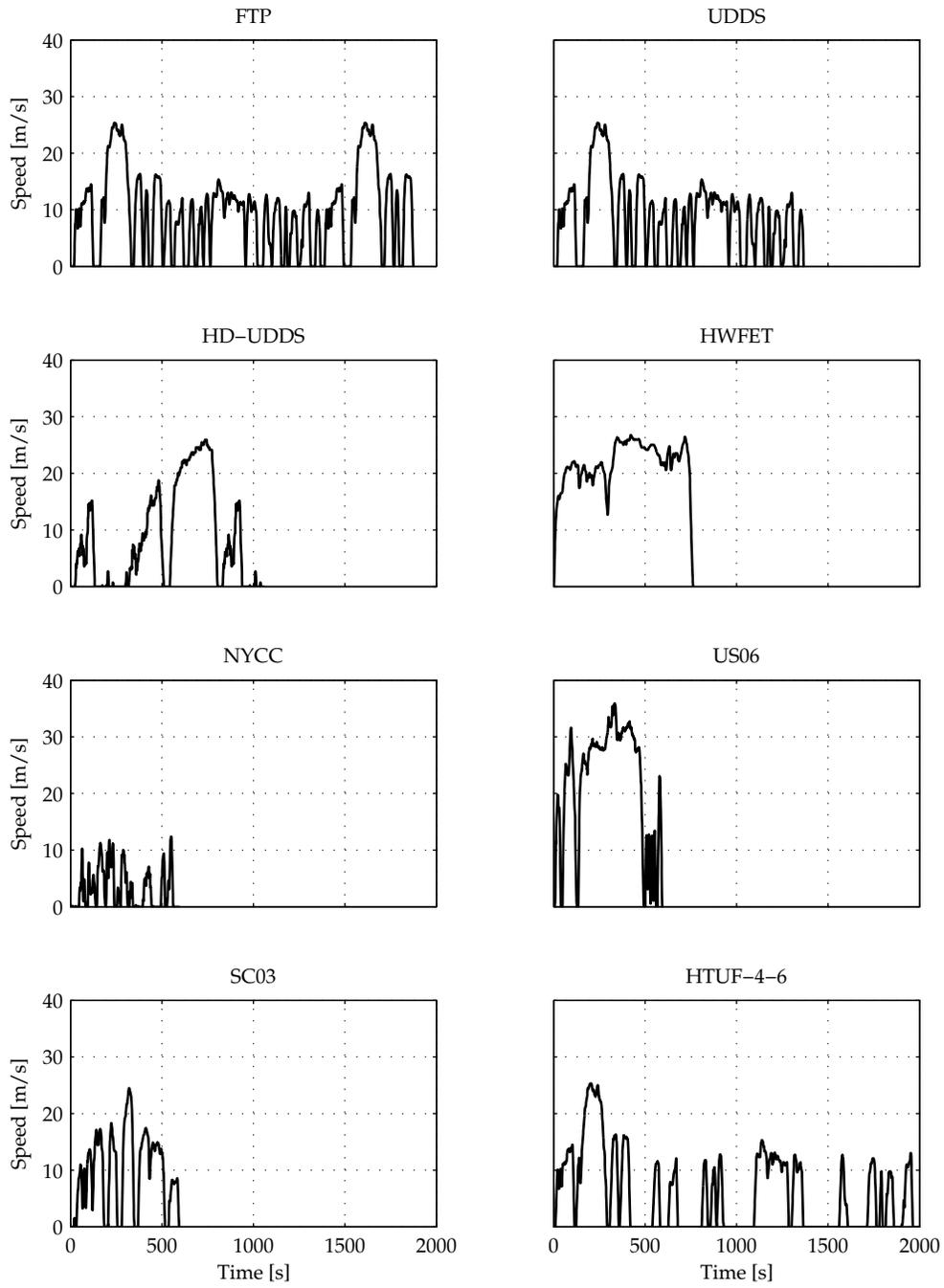


Figure 1.2: Velocity profiles of the U.S. regulatory cycles [1]

Choosing the correct combination is usually a complex problem. If the vehicle is decelerating, it is obvious that the electric accumulator should receive as much of the braking energy as possible. But, if the vehicle is accelerating, is it more advantageous to use the engine and leave the battery charged for later use, or to use some of the energy stored in the battery instead of running the engine? In general, the answer to this question depends on several variables.

The first aspect to consider is the actual objective of hybridization. Hybrid vehicles are mostly being developed for reducing fuel consumption, but they can also provide other advantages, such as reduction of pollutant emissions (due to the higher flexibility in controlling engine operation in comparison to conventional vehicles). In general, it is possible to define the objective of hybridization as the minimization of a given cost function, representing fuel consumption, emissions, or a sum of both. The minimization should ideally take place over the entire life cycle of the vehicle, but in practical cases the optimization horizon is finite and usually coincides with a short trip or section of a trip, with duration of several minutes or a few hours.

The other important issue to be taken into account is the typology of hybrid vehicle to deal with. In particular, a *charge-sustaining* vehicle will be characterized by the fact that the state of charge of the electric buffer (e.g. battery) at the end of the optimization horizon should be the same as it was at the beginning, or at least very close. In this case, the entire energy needed for completing the trip derives from the fuel. A *charge-depleting*, or *plug-in*, hybrid vehicle instead uses its batteries much more, and the state of charge can decrease sensibly at the end of a trip, because the vehicle can be plugged in the electrical grid to be recharged. In this case, a substantial amount of the total energy needed for a trip is deriving from the battery (and ultimately from the electric grid), not the fuel. The differences between the two cases, in terms of the control problem, can be seen as different boundary conditions and different optimization objectives. Boundary conditions are different because the state of charge variation is zero for charge-sustaining hybrids, but can be arbitrary or pre-determined for plug-in hybrids (depending on the case). The difference in optimization objectives is that, while fuel consumption

is generally the minimization objective in a charge-sustaining hybrid, the total energy consumption, or the total expense, may be a more significant cost function for a plug-in vehicle[25].

In general, once a suitable optimization horizon and cost function have been decided, the control problem in hybrid vehicles consists in minimizing the total cost (an integral function) using a sequence of instantaneous actions. This is a typical optimal control problem, and several methods can be used for its solution. Chapter 3 gives a formal definition of this problem and describes some of the techniques used for its solution. In this section, we attempt at an informal description of the possible approaches. These can be subdivided into four categories: numerical optimization, analytical optimal control theory, instantaneous optimization, and heuristic control techniques. In the first two cases, the problem is considered in its entirety, i.e. taking into account at each instant information related to past, present, and future time; in the latter two, the solution at each time is calculated based only on present (and possibly past) information.

### 1.3.1 Numerical global optimization

In general, the optimal solution to the problem is only achievable if the entire horizon is considered at once, i.e., if the driving cycle is well defined and known in advance. This is clearly not possible in a real vehicle, because it is impossible to know exactly its future driving conditions (speed, road slope etc.), or even the duration of the trip<sup>2</sup>. Despite this, it is interesting to consider the ideal case in which perfect information on the entire trip is available. Even if not directly applicable, the optimal solution obtained in simulation can be used as a comparative benchmark for implementable strategies, and to gain insights into the behavior of the system. The method most widely used for obtaining the optimal solution in case of perfect and complete information is dynamic programming [28, 29, 26, 9], which is a numerical technique for solving the optimal control problem backwards

<sup>2</sup>Approximated information may be available if the route is known (for example if the driver has pre-programmed the trip on the GPS navigation system) and has been successfully used as an auxiliary resource for actual algorithms [26, 27]. However, an exact prediction of velocity profile on a public road is impossible, because of traffic and other disturbances.

in time, i.e. starting from the final instant of the driving cycle and proceeding backwards, ending at the initial time. It is based on Bellman's principle of optimality [30], stating that, given an optimal control sequence for a problem, the optimal sequence from any of its intermediate steps to the end corresponds to the terminal part of the overall optimal sequence. Thanks to this principle, the optimal solution can be calculated step by step starting at the final time and minimizing the *cost-to-go* at each step, i.e. the cost incurred in moving from that step to the end. From a practical standpoint, dynamic programming gives the same results that would be obtained using an enumerative solution (i.e. considering all the possible combinations of control sequences and choosing the one with the lowest total cost), but in a fraction of the computational time, because the number of combinations to be evaluated is greatly reduced. In fact, at each time step the optimal path to the end is found and stored, discarding all the other combinations, because the optimality principle guarantees that the solution from there to the end will not be affected by the previous control actions.

### 1.3.2 Analytical optimal control techniques

Traditional optimal control theory (whose origins can be dated back to the 17th century [31]) provides a mathematical framework for addressing the dynamic optimization problem. Unfortunately, the energy management problem in hybrid vehicles is rather complex and must be simplified and abstracted significantly in order to be completely solved using these techniques. Nonetheless, applying optimal control theory to the abstracted problem allows for its better understanding and can lead to improvements of practically implementable solutions. One of the most powerful results in optimal control theory is Pontryagin's minimum principle [32, 33, 34], which gives necessary condition that the optimal solution must satisfy. Despite offering only necessary (not sufficient) optimality conditions, the principle (which is, in fact, a theorem) is extremely useful because applicable to any problem, since it does not impose any restrictive hypothesis on the analytical properties of the mathematical functions involved in the problem formulation. In practice, Pontryagin's principle can be used to generate solution candidates; if the

optimal control problem admits one solution and the necessary conditions are satisfied by a single candidate, the solution obtained with this principle is the optimal solution.

In the field of HEV optimization, Pontryagin's principle has been used by several authors [35, 36, 37, 38, 39] to find the optimal power split given the driving cycle. It can be a valid alternative to dynamic programming if the power flows in the powertrain can be described with simple analytical functions and offers very significant insights into the problem, but it cannot be applied in practice without a-priori knowledge of the cycle.

### **1.3.3 Instantaneous optimization**

A third family of control strategies includes those that modify the global optimal control problem into a sequence of local (instantaneous) problems, thus calculating the solution as a sequence of local minima. This approach works well if the local minimization is well defined. The equivalent consumption minimization strategy (ECMS), first introduced by Paganelli et al. [7, 40] and developed at the Ohio State University [41, 10, 9], is the most well-known of these strategies. ECMS is based on the concept that, in charge-sustaining vehicles, the difference between the initial and final state of charge of the battery is very small, negligible with respect to the total energy used. This means that the electrical energy storage is used only as an energy buffer. Since all the energy ultimately comes from fuel, the battery can be seen as an auxiliary, reversible fuel tank. The electricity used during a battery discharge phase must be replenished at a later phase using the fuel from the engine (either directly or indirectly through a regenerative path).

Two cases are possible at a given operating point:

1. the battery power is positive (discharge case): a recharge with the engine will require some additional fuel consumption in the future;
2. the battery power is negative (charge case): the stored electrical energy will be used to reduce the engine load, which implies a fuel saving.

The instantaneous cost that is minimized at each instant is called *equivalent fuel consumption* and is obtained by adding a term to the actual engine fuel consumption. This term is positive in case 1 above and negative in case 2; it represents the virtual fuel consumption associated with the use of the battery, and – if suitably defined – allows to obtain results close to the optimal solution, while maintaining the battery state of charge at the desired level. The big advantage of this approach is that, being based on instantaneous minimization, is easily implementable in real time. As mentioned, a proper definition of the equivalent fuel consumption is necessary to achieve quasi-optimal results, but this requires optimization of the tuning parameters which is only possible if the driving cycle is known in advance. However, good results have been achieved with adaptive ECMS based on driving pattern recognition [10], a more refined strategy that can recognize the type of driving conditions in which the vehicle is being used (e.g. city, highway, suburban roads etc.) and dynamically adapt the definition of virtual fuel consumption in order to find the best match to each situation.

#### **1.3.4 Heuristic control techniques**

Heuristic control techniques are not based on minimization or optimization, but rather on a pre-defined set of rules. The rules generate the control action (i.e., the value of power delivered from each energy source) based on the instantaneous values of several significant vehicle parameters (vehicle speed, power demand, battery state of charge, etc.). Many times, rules are derived using engineering judgment and a substantial amount of testing for tuning their parameters [12, 42, 43, 44]; the technique can be made robust and suitable for production vehicles, but the results may not be optimal, since they are not based on formal optimization techniques. In some cases rules can be extracted from the optimal solution found using dynamic programming, thus representing a method to implement (at least approximately) the optimal solution. For example, it may be possible to create a set of rules that try to mimic the optimal vehicle behavior based on the observation of external inputs and the state of the system [45, 46, 47, 48].

## 1.4 Powertrain modeling for energy management

As mentioned previously, most of the energy management techniques based on global optimization (either numerical or analytical) rely on mathematical models of the vehicle in order to calculate the fuel consumption starting from the driving cycle. For an accurate estimate of the fuel consumption, it is not necessary to capture all the details in the dynamic behavior of the powertrain, but it is important to take into account all losses and all the interactions between components. A low-order dynamic model of the powertrain (including only the inertia of the vehicle and engine), accounting for losses in all the major powertrain components, is sufficient to capture almost all the energy flows in the vehicle [49]; this kind of modeling approach is detailed in Chapter 2.

Oversimplified models, on the other hand, may lead to erroneous results of the energy management strategy if this is not able to correctly discern between operating points with different efficiency characteristics. To understand the consequences of oversimplification, consider as an example the case in which the control strategy should choose the transmission ratio to minimize the engine fuel consumption. If the transmission is modeled using a constant efficiency, the strategy looks just at the engine map and determines that the engine fuel consumption decreases by 2 % if the fourth gear is engaged rather than the third, and therefore shifts from third to fourth gear. However, in the real vehicle, the transmission losses are higher in fourth gear than they are in third by 5 %. This means that the fuel consumption actually increases. Obviously, the only way to ensure that the strategy chooses the optimal gear ratio is to use a transmission model that accounts for the fact that the transmission efficiency changes with the gear selected.

## 1.5 Organization and contributions of the dissertation

Following this introduction, Chapter 2 presents a formalization of an energetic approach for modeling the power flows in hybrid electric vehicles, and describes computational methods for implementing such models.

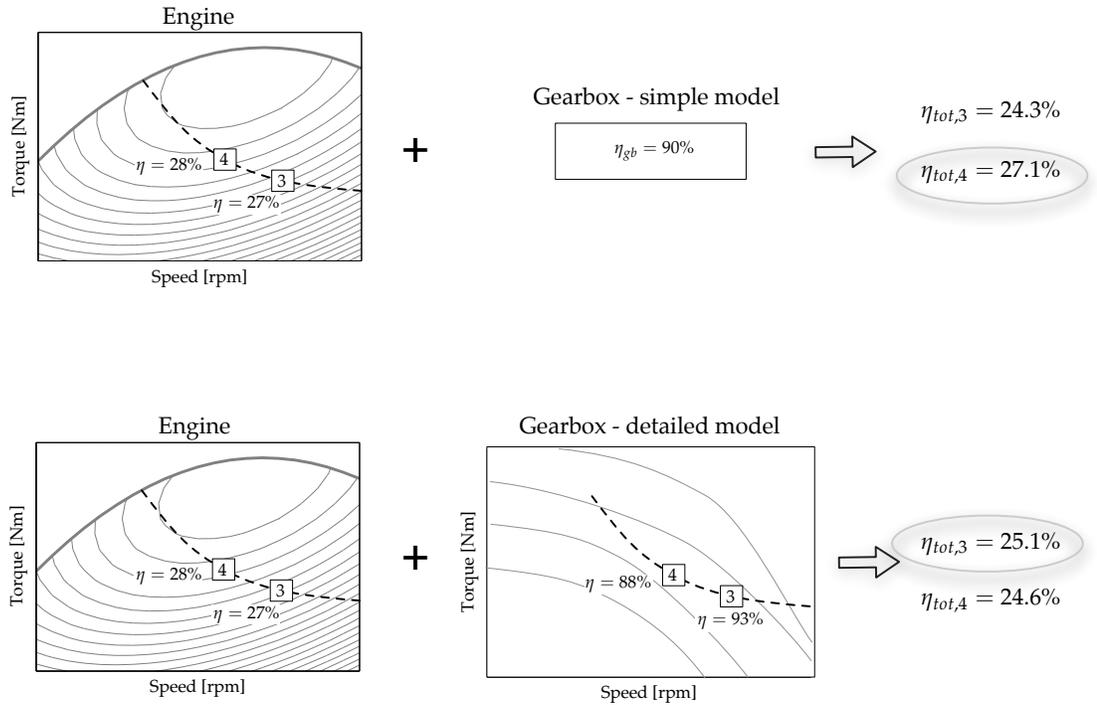


Figure 1.3: Example of integration of models with different accuracy

The formal definition of the energy management problem and an organic review of strategies described in literature is proposed in Chapter 3; Chapter 4 illustrates the results of three approaches to energy management using comparative simulations in two case studies.

The contributions of this dissertation are the following:

- Formalization of the optimal control problem
- Review of several energy management methods
- Development of detailed behavioral model
- Development of control-oriented model
- Application of three energy management strategies: analysis and comparison

## 1.6 Case studies

Two case studies are used to demonstrate the application of the strategies in Chapter 4. They represent two different vehicles, a truck and a mid-size SUV, characterized both by a series hybrid architecture, but using different energy storage systems (capacitors in the first case, high capacity Li-Ion batteries in the second). Despite its simplicity, the series hybrid architecture is of interest because less work has been published on series HEVs in comparison to other architectures (especially parallel), more common among light-duty vehicles; furthermore, the series architecture can be used in plug-in HEVs, especially those in which the internal combustion engine (or fuel cell) is used as an auxiliary power unit (APU).

### 1.6.1 Case study 1: AH2PS project and experimental vehicle

To increase the interest of manufacturers in developing hybrid electric trucks, promoting the creation of the necessary technology, in 2002 the U.S. Department of Energy (DOE) and the National Renewable Energy Laboratories (NREL) started the Advanced Heavy Hybrid Propulsion System (AH2PS) project, a government/industry cost-shared research and development project for advanced, next-generation heavy hybrid propulsion components and heavy hybrid vehicle systems [50]. Four sub-projects have been funded: two are related to vehicle system integration for a Class 4-6 hybrid delivery truck and a Class 7-8 hybrid refuse collection truck, the other two consist in a hybrid electric bus and in waste energy recovery systems. All these projects are related to vehicle families representing applications of hybridization potentially very effective, because they are driven in and around the city, with frequent stops.

The Ohio State University partnered with Oshkosh corp., the assignee of the second sub-contract, providing tools for system-level modeling and supervisory control. The project was composed by two phases. The first phase saw the definition of the vehicle architecture, the development of driving cycles representative of typical operating conditions, and a comprehensive analysis of energy flows to

Class	Weight range
1	0-6000 lbs. (0-2721.5 kg)
2	6001-10000 lbs. (2722-4536 kg)
3	10001-14000 lbs. (4536.5-6350 kg)
4	14001-16000 lbs. (6351-7257.5 kg)
5	16001-19500 lbs. (7258-8845 kg)
6	19501-26000 lbs. (8845.5-11793.5 kg)
7	26001-33000 lbs. (11794-14968.5 kg)
8	33001 lbs. and over (14969 kg and over)

Table 1.1: Vehicle weight classification based on GVWR (Gross vehicle weight rating, i.e. the maximum allowable total weight of a road vehicle when loaded - i.e including the weight of the vehicle itself plus fuel, passengers, cargo, and trailer tongue weight).

identify the sources of losses and the areas of possible improvements. In the second phase, the design was optimized at the component level, to obtain a prototype vehicle serving as technology demonstrator.

As one of the project outcomes, the present dissertation describes the development of a longitudinal, low-order dynamic simulator of the vehicle, and introduces an optimal control strategy for the minimization of the fuel consumption.

### **Driving cycles**

In order to start designing the propulsion system for any vehicle, especially a hybrid electric vehicle, it is very important to analyze its loads, intended as driving cycles. This is definitely a difficult task, since each individual vehicle produce will follow a different life pattern. However, it is somewhat easier to do for a truck than it is for a passenger vehicle, because of the fact that most trucks have a rather specialized use. Therefore, an important part of the first phase of the project was to determine the characteristics of the driving cycles of typical refuse vehicles in the United States. This was done by following several of these trucks during

their service in various cities and states. The work is presented in [51] and briefly summarized here. The cities taken into consideration were:

- Blaine, MN (for extreme cold);
- Fort Walton Beach, FL (for extreme humidity);
- Fort Worth, TX (for extreme heat);
- Chandler, AZ (for extreme heat);
- Ogden, UT (for extreme grade).

In each city, a truck was instrumented and followed for a week, measuring the speed, the mass before and after each dump (i.e. once or twice per day), the road elevation (for estimation of the grade), the engine load, and the power required to lift, pack, and dump the refuse hauled by the truck. This allowed the creation of a valuable database with driving and loading cycles representative of real-world conditions, which was also used to generate a few synthetic test cycles capable of capturing the most important aspects of the measured cycles. The synthetic cycles were created using statistical analysis of all the data collected. Each trip was subdivided into elementary sequences representing the trip from one house to the next or, in general, the driving conditions between two subsequent stops. Several “operating modes” can be identified based on the typical operation of these vehicles. Usually, the truck leaves the deposit early in the morning and travels on highway, unloaded, to reach the neighborhoods where the refuse collection operations take place: this high-speed trip, whose average duration is about one hour, is called *Approach*. Once arrived in the city, the truck starts the operation of refuse collection, which consist in driving from one house to the next, stopping each time for collecting a refuse bin. This phase, called *Route*, lasts a few hours, and both the average and the peak speed are very low. When the truck is full, it travels to the dump site for dumping the refuse: the trip involves a highway section taking place with full load, and is called *Return*.

### 1.6.2 Case study 2: EcoCAR Challenge

EcoCAR Challenge [52] is a competition among seventeen north american universities, sponsored by the United States Department of Energy (DoE) and General Motors (GM). The student teams are asked to re-design the powertrain of a production vehicle (the Saturn VUE) in order to minimize fuel consumption and emission without reducing performance and consumer acceptability. The Ohio State University team chose to compete with a range-extended electric powertrain, in which the wheels are propelled by electric motors and power is supplied by Li-Ion batteries and by an auxiliary power unit (APU), a spark-ignited engine propelled either by gasoline or by E-85 (a mixture composed by 85% of ethanol and 15% of gasoline). The actual vehicle architecture is more complex and also allows to connect the engine to the front wheels with a clutch, in order to reduce losses at highway speed; this mode of operation is not considered in this dissertation.

## Chapter 2

### POWERTRAIN MODELING

Most engineering students have heard at some point that *all models are wrong, but some are useful*<sup>1</sup>. The expression is used to point out that every mathematical model can only be an approximation of physical reality. In order to create efficient models, it is necessary to understand what level of approximation is acceptable for the application at hand (i.e. what physical phenomena must be taken into account) and then to identify suitable mathematical models for these phenomena, capable of delivering the right compromise between accuracy and computational time. Sometimes, modeling is described as an “art”, because the ability to achieve these objectives derives not only from engineering knowledge and ingenuity, but also from experience and intuition. A good model is still wrong, but it can be useful if its limits and assumptions are known and appropriate for its application.

#### 2.1 An energetic approach to hybrid electric vehicles

In order to evaluate fuel consumption, it is important to correctly understand the energy flows in the powertrain and identify the areas in which savings can be introduced – for example by hybridization. In this chapter, we first describe the energy flows in a vehicle powertrain, identifying the source of losses and describing the relevance of driving cycles in the energy balance; then present a brief overview of the methods normally used to evaluate fuel consumption in conventional and hybrid vehicles, and finally describe the models used in this work and their implementation and validation against experimental data.

<sup>1</sup>George Box, statistician

### 2.1.1 Vehicle energy balance

If a vehicle is considered as a mass-point, its equilibrium equation can be written as:

$$M_{veh} \frac{dV_{veh}}{dt} = F_{inertia} = F_{trac} - F_{roll} - F_{aero} - F_{grade} \quad (2.1)$$

where  $M_{veh}$  is the total vehicle mass,  $V_{veh}$  is the vehicle velocity,  $F_{inertia}$  is the inertial force,  $F_{trac}$  is the tractive force generated by the powertrain at the wheels<sup>2</sup>,  $F_{roll}$  is the rolling resistance (friction due to tire deformation),  $F_{aero}$  the aerodynamic resistance,  $F_{grade}$  the force due to road slope.

The aerodynamic resistance can be expressed as

$$F_{aero} = \frac{1}{2} \rho_{air} A_f C_d V_{veh}^2 \quad (2.2)$$

where  $\rho_{air}$  is the air density,  $A_f$  the vehicle frontal area,  $C_d$  the aerodynamic drag coefficient.

The rolling resistance force is usually modeled as [14]

$$F_{roll} = c_{roll}(V_{veh}, p_{tire}, \dots) M_{veh} g \cos \alpha \quad (2.3)$$

where  $g$  is the gravity acceleration,  $\alpha$  the road slope angle (so that  $M_{veh} g \cos \alpha$  is the vertical component of the vehicle weight), and  $c_{roll}$  is a rolling resistance coefficient which is, in principle, a function of vehicle speed, tire pressure  $p_{tire}$ , external temperature etc. In most cases,  $c_{roll}$  is assumed to be constant, or to be an affine function of the vehicle speed. The order of magnitude of  $c_{roll}$  is 0.01-0.02, which means that the rolling resistance is roughly 1-2 % of the vehicle weight.

The grade force is the horizontal component of the vehicle weight:

$$F_{grade} = M_{veh} g \sin \alpha. \quad (2.4)$$

Eq. (2.1) can be rearranged to calculate the tractive force that the powertrain needs to produce can be derived from, given the inertia:

<sup>2</sup>i.e., it is the net torque acting on the rims of the traction wheels divided by the effective wheel radius

$$F_{trac} = F_{inertia} + F_{grade} + F_{roll} + F_{aero}. \quad (2.5)$$

While (2.1) and (2.5) are the same equation, their different form is a representation of two modeling approaches: in (2.1), the vehicle acceleration  $\frac{dV_{veh}}{dt}$  is calculated as a consequence of the tractive force generated by the powertrain (and obviously the external resistance terms), and the speed is then obtained by integration of the acceleration; this is the “forward” approach, which reproduces the physical causality of the system. On the other hand, in (2.5), the tractive force is calculated starting from the inertia force: in this case, it is assumed that the vehicle is following a prescribed velocity (and acceleration) cycle, and  $F_{trac}$  represents the corresponding force that the powertrain must supply; this is called “backward” approach (force follows velocity). A more in-depth discussion of these two approaches is given in Section 2.1.3.

The inertial force  $F_{inertia}$  is positive when the vehicle is accelerating, and negative during deceleration; the grade force  $F_{grade}$  is positive when the vehicle is driven uphill and negative when it is going downhill; the rolling and aerodynamic resistances are always positive. Depending on the net value of  $F_{trac}$ , three operating modes can be identified:

1. traction, if  $F_{trac} > 0$  ;
2. braking, if  $F_{trac} < 0$ ;
3. coasting, if  $F_{trac} = 0$ .

Note that, even in traction mode ( $F_{trac} > 0$ ), the vehicle can decelerate if the traction force is smaller than sum of rolling and aerodynamic resistance and grade force. On the other hand, in coasting or braking modes, the vehicle decelerates unless the road slope is positive (downhill) enough to exceed the resistances and the braking forces.

The forces  $F_{roll}$  and  $F_{aero}$  are dissipative, since they always oppose the motion of the vehicle, while the inertial and grade force are conservative, being only dependent on the vehicle state (respectively velocity and altitude). Thus, part of the tractive force generated by the powertrain increases the kinematic and potential energy of the vehicle (by accelerating it and moving it uphill), and part is dissipated in rolling and aerodynamic resistances. When the vehicle decelerates or drives downhill, its potential and kinetic energy must be dissipated: again, rolling and aerodynamic resistances contribute to dissipating part of the vehicle energy, but for faster deceleration the mechanical brakes must be used. Thus, ultimately, all the energy that the powertrain produces is dissipated in these three forms: rolling resistance, aerodynamic resistance, and mechanical brakes. The kinetic and potential energy, on the other hand, are never dissipated: the net variation of kinetic energy is always zero between two stops (since initial speed and final speed are both zero), and the variation of potential energy only depends on the difference of altitude between the initial and ending point of the trip considered.

Multiplying all terms of (2.5) by the vehicle speed the following balance of power is obtained:

$$P_{trac} = P_{inertia} + P_{grade} + P_{roll} + P_{aero}. \quad (2.6)$$

The term  $P_{trac}$  represents the tractive power at the wheels, both positive and negative. The powertrain provides all the positive values of  $P_{trac}$ , while the negative values are generated partially by the brakes and partially by the powertrain. In conventional vehicles, the amount of negative power that the powertrain can absorb is rather limited: it consists in friction losses in the various components and pumping losses in the engine. In hybrid electric vehicles, the amount of negative power is much higher, since the electric traction machines are reversible and can be used for deceleration as well as acceleration.

The term  $P_{inertia} = M_{veh} \dot{V}_{veh} V_{veh}$  represents the amount of power needed just to accelerate the vehicle (without considering the losses); the terms  $P_{roll} = F_{roll} V_{veh}$  and  $P_{aero} = F_{aero} V_{veh}$  are the amount of power needed to overcome the rolling and aerodynamic resistances respectively; and  $P_{grade} = F_{grade} V_{veh}$  is the power that

goes into overcoming a slope (or, if the slope is negative and the vehicle is going downhill, is the power that the powertrain and/or the brakes must dissipate to prevent undesired acceleration).

The vehicle energy balance is obtained by integration of (2.6) over the duration of a trip:

$$E_{trac} = E_{kin} + E_{pot} + E_{roll} + E_{aero}, \quad (2.7)$$

where each term represents the variation of each form of energy (corresponding to the integral of the respective power) between the initial and final instant of the trip. Note that the integral of the inertial power  $P_{inertia}$  is the kinetic energy  $E_{kin}$ , and the integral of the grade power  $P_{grade}$  is the potential energy  $E_{pot}$ . As mentioned earlier, assuming that the speed at the beginning and the end of the trip is zero and that there is no variation in altitude between the initial and final position, the variation of kinetic and potential energy is zero:

$$E_{kin} + E_{pot} = E_{trac} - E_{roll} - E_{aero} = 0. \quad (2.8)$$

The relative amount of rolling resistance, aerodynamic resistance, and brake energy defines the characteristics of a driving cycle. In particular, the potential for energy recovery using regenerative braking is equal to the amount of kinetic and potential energy that needs to be dissipated, minus the quantity that is dissipated because of rolling and aerodynamic resistance. Thus, a urban driving cycle with frequent accelerations and decelerations at low speed (where the resistances are lower) presents more potential for energy recovery than a highway cycle in which the speed is more or less constant and the losses due to aerodynamic resistance represent the major component of the power request by the vehicle.

### 2.1.2 Powertrain losses

Because of the losses in the powertrain, the net amount of energy produced at the wheels is smaller than the amount of energy introduced into the vehicle from external sources (e.g. fuel). Conversion losses take place each time power is transformed into a different form (e.g., chemical into mechanical, mechanical into

electrical etc.). Similarly, each time that the power flow passes through a connection device, friction losses and other inefficiencies reduce the amount of power flowing out of the device with respect to the power flowing into it. In order to generate a powertrain model capable of estimating the fuel consumption during a given driving cycle, these losses must be taken into account using appropriate component models.

### **2.1.3 Modeling approaches**

Three approaches can be used to estimate the fuel consumption of a vehicle given the prescribed driving cycle [14]: the average operating point approach, the quasi-static approach, and the dynamic approach.

The average operating point approach consists in calculating one single operating point of the engine, to be assumed as representative of its average efficiency during the driving cycle. The average operating point is calculated starting from the average value of the power request at the wheels during the tractive section of the cycle, and working backwards through the powertrain components. Each component can be represented using its average efficiency.

The quasi-static approach retains the sequential nature of the driving cycle and does not lump all the cycle into a single operating point, but it is based on the assumption that the prescribed driving cycle is followed exactly by the vehicle. The driving cycle is subdivided in small time intervals (typically of 1 s), during which the average operating point approach is applied, assuming that speed, torque, and acceleration remain constant. Each powertrain component is modeled using an efficiency map, a power loss map, or a fuel consumption map: these give a relation between the losses in the component and the present operating conditions, averaged during the desired time interval.

The dynamic approach is based on a first-principles description of each powertrain component, with dynamic equations describing the evolution of its state. There is no limit to the degree of modeling detail, which depends on the time scale and the nature of the phenomena that the model should predict.

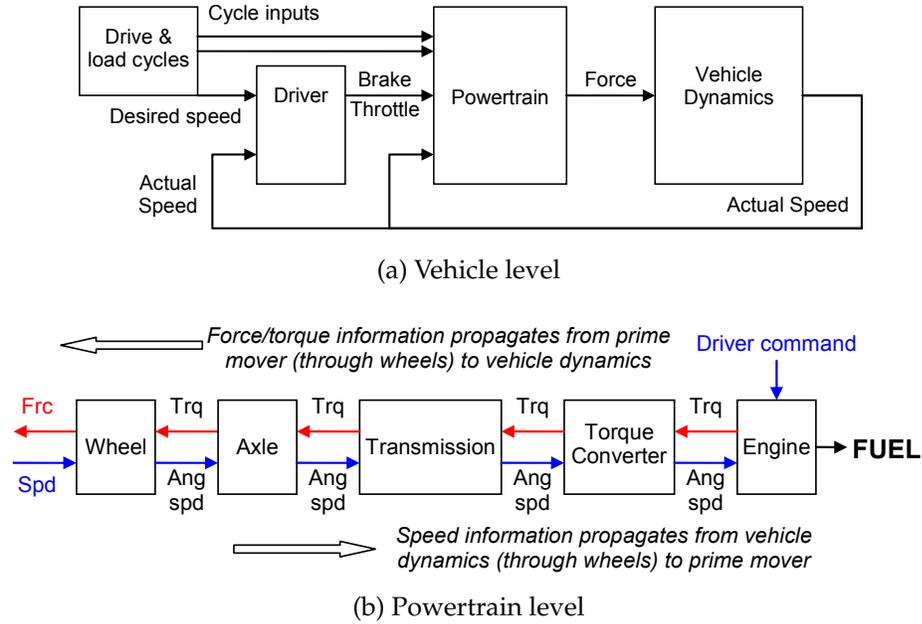


Figure 2.1: Information flow in a forward simulator [2]

The quasi-static and dynamic approach just described represent the two views of the vehicle equilibrium equation expressed by (2.5) and (2.1) respectively. As mentioned in Section 2.1.1, the two approaches are sometimes categorized as *backward* and *forward*, focusing on the modeling philosophy, i.e. information flow, more than on the level of detail.

The forward approach is the option typically chosen in most simulators; it is characterized by the information flow as shown in Figure 2.1. The desired speed (from the cycle inputs) is compared to the actual vehicle speed, and braking or throttle commands are generated using a driver model (typically a PID speed controller) in order to follow the imposed vehicle profile. This driver command is an input to the engine and the rest of the powertrain components, which ultimately produce a tractive force. Finally, the force is applied to the vehicle dynamics model, where the acceleration is determined taking into account the road load information [2].

In a backward simulator, instead (see Figure 2.2), no driver model is necessary, since the desired speed is a direct input to the simulator, while the engine torque

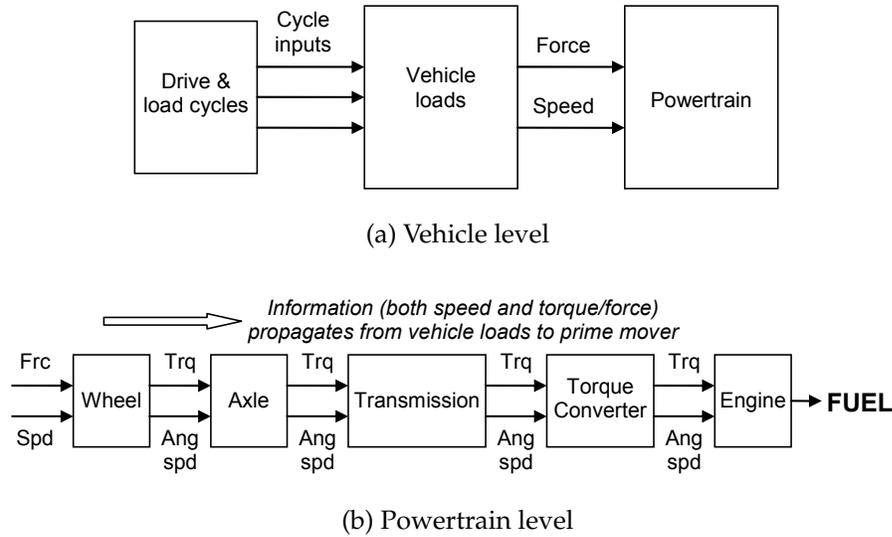


Figure 2.2: Information flow in a backward simulator [2]

and fuel consumption are outputs. The simulator determines the net tractive force to be applied based on the velocity, payload, and grade profiles, along with the vehicle characteristics. Based on this information, the torque that the traction motors should apply is calculated, and then the torque/speed characteristics of the various powertrain components are taken into account in order to determine the engine operating conditions and, finally, the fuel consumption.

Both the forward and backward simulation approaches have their relative strengths and weaknesses. Fuel economy simulations are typically conducted over predetermined driving cycles, and therefore using a backward simulator ensures that each different simulation exactly follows this profile. By contrast, a forward simulator will generally not exactly follow the trace, and a small error between the actual and the desired signal will generally exist. Proper tuning of the driver block can reduce the differences, at the price of some extra time and effort, whereas the backward version keeps the error at zero without any effort. However, backward simulator cannot easily capture the powertrain limits. There is no guarantee that a given vehicle/powertrain will actually be able to meet the desired trace. In a backward

simulator, however, it must be assumed that the trace can be met, and information about the actual limits does not become available until the end of the calculations, when the operating conditions of the prime movers are evaluated. A forward simulator takes these limits into account, since the force information originates at the prime mover (where it can be limited to the component maximum), and then is transmitted to the vehicle (allowing other constraints to be imposed). Similarly forward simulators are more suited to acceleration tests, by simply forcing the driver to give a full throttle command. From the control development point of view, forward simulators are preferred because they maintain the physical causality of the real system and allow using the same controller inputs/outputs in the simulator as well as in the real system (assuming that the modeling detail is adequate).

#### **2.1.4 Approach used in this work**

In this dissertation, the modeling objective is to predict vehicle fuel consumption and performance, which are the most noticeable consequences of hybridization and are directly affected by the sizing of powertrain components and the tuning of supervisory control strategy. Also, the model developed is intended to be used for a wide variety of vehicles, and for each of them it should allow to:

- predict vehicle performance
- predict energy flows and fuel consumption
- develop supervisory control strategies

In light of this, it was decided to use the forward approach described in Section 2.1.3 to create a dynamic simulator of the powertrain, intended to describe the transient response of the vehicle to the driver's power demand. Since fuel consumption and energy flows are the main concern, the simulator must be fast enough to allow evaluation of relatively long driving cycles (up to several hours of simulated driving). Accurate performance prediction requires the evaluation of phenomena that are characterized by time constants of 0.1 s and lower (10-20

Hz); on the other hand, fuel consumption is evaluated over quite long driving cycles (from 15 minutes to several hours). The contrast between the two objectives was solved by developing a modular simulator, in which each component is represented by a block that keeps the same external interface independently from the complexity of the model it contains. In the simplest version, each block contains lumped-parameters models that are either static or have one single degree of freedom. In most cases, losses in the components are calculated using efficiency maps, following the typical approach described in literature [53, 54] for this kind of applications.

Thus, the simulator is composed of simple models so that long driving cycle evaluation can be conducted at reasonable speed, but, whenever a more accurate simulation of powertrain dynamics is needed (for example for evaluating gear shifting strategies), the modeling detail can be changed by using different versions of the same blocks. The simulation will be more accurate albeit slower, which is acceptable since dynamic simulations usually cover a much shorter time frame than a complete driving cycle.

## 2.2 Physical modeling tools

Numerical simulation of the behavior of dynamic systems has been an important engineering topic since the introduction of computers. In the past two decades, numerous software tools have been introduced for this purpose. Some of them became part of university curricula and are known to almost any recent engineering graduate: the most notable examples are The Mathworks<sup>®</sup> products, *Matlab*<sup>®</sup> and *Simulink*<sup>®</sup>. Others were short-lived, or found a very narrow range of applications.

There are two main families of engineering modeling tools, which could be defined as *math-based* or *physics-based* approaches. Math-based tools are essentially interfaces for writing differential equations in explicit form, but using a graphical user interface (GUI) that makes it easy for the users to define equations, parameters, and to run simulations. Simulink is the best known and the most full-featured of these tools.

Physics-based tools serve a different purpose: instead of providing an environment in which the users write their own equations, they provide a library of elements that represent physical objects, and can be connected to each other to form a system that mimics the interaction that these objects would have in reality. The term *physical modeling* is sometimes used to describe this modeling approach. Several software tools allow for doing this, in different domains and using different approaches.

*Easy5* [55], initially developed by Boeing in 1974 [56], is now a commercial product developed by MSC software. It was among the first modeling tools to be provided with a graphical user interface, and was initially aimed at modeling air-aircraft interaction and developing control flight dynamics; it now a comprehensive modeling tool that includes libraries for hydraulics, aerospace dynamics, powertrain dynamics, engines, etc. The approach followed is, in general, to model the 1-D flow of power and the balance of effort between components. For example, a hydraulic system will be approximated by a set of lines and components, each point of which represents the average fluid properties over the section considered. The software comes with several libraries of components that the user can choose from, and also allows for defining customized models.

The same description fits other modeling tools, such as *AMESim* [57], developed by the French company Imagine (now owned by LMS International) and Ansoft *Simplorer*<sup>®</sup>. *AMESim* started out as a tool for 1-D flow simulation applied to hydraulic systems; after rapidly becoming very popular in that field, it was expanded with additional libraries and marketed as a comprehensive physical modeling tool. The main reason for *AMESim* success, apart from the nice interface and ease of use (which is common to *Easy5* and others), is the fact that it provides very effective numerical integration methods for the stiff dynamic equations that characterize hydraulic systems.

All these packages are based on standard programming languages (C, C++), and usually require the user to code in that language (or to tweak existing code) in order to modify object templates or create new ones (GUI that facilitate this operation are usually provided).

A different approach to a-causal modeling is provided by *Modelica* [58], a special programming language initially developed by H. Elmqvist in his PhD dissertation [59] and well described in [60]. In this case, the user only needs to define one form for each equation, and the special compiler will then manipulate the equations at the moment of combining them with those of the other components. The most successful commercial implementation of the Modelica language (which, by itself, is open-source) is Dynasim *Dymola*.

In recent years, Simulink has also been expanded with physical modeling tools that introduce the ability to use blocks representing physical components and physical connections rather than mathematical operators and signals. One of these, *SimDriveline*<sup>™</sup>, is especially designed for modeling powertrain dynamics. Others are *SimHydraulics*<sup>®</sup> (for hydraulic systems), *SimPowerSystems*<sup>™</sup> (for electric systems), *SimMechanics*<sup>®</sup> (for dynamics and kinematics of mechanisms). All these tools are organized within a common framework called *SimScape*<sup>®</sup>, which is the Mathworks' alternative to Modelica, in the sense that it introduces the ability for the users to define their own physics-based models and to integrate blocks from the other Simulink-based physical modeling tools.

In general, the physical, object-based modeling approach is very attractive because it makes extremely easy to create models of even very complicated systems: apparently, all it takes is to put the pieces together. This is the most visible characteristic of what is sometimes called *a-causal* modeling: the term refers to the fact that, while usually one writes equations in explicit form and has to decide which variable is an input and which is an output, in this case there is no need to define explicitly the inputs and outputs of the model, i.e. its *causality*. The software will do this for the user, by setting up a correct set of equations based on the way the components are connected. In the tools mentioned so far, this flexibility is achieved by embedding in each component model all the possible forms<sup>3</sup> of the system dynamic equations, i.e. considering all combinations of inputs and outputs.

<sup>3</sup>note that, if each component represents an elementary building block as it usually happens, there are usually just a few possible ways of writing its equations.

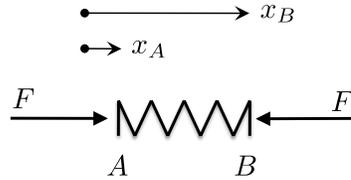


Figure 2.3: Elementary model of a spring

For example, consider the ideal spring shown in Figure 2.3. The input (known) variables can be either the force  $F$  or the displacement of one of the terminal points,  $x_A$  or  $x_B$ ; the output (unknown) variables will be the corresponding missing information. There are then three forms for the system equation:

$$F = K_s(x_B - x_A - L_0) \quad (2.9)$$

$$x_B = \frac{F}{K_s} + x_A + L_0 \quad (2.10)$$

$$x_A = x_B - L_0 - \frac{F}{K_s} \quad (2.11)$$

where  $L_0$  is the length of the spring at rest and  $K_s$  is its stiffness. The software should be able to determine which form of the equation is more suited to the system being considered, taking into account the information available from other components and the expected outputs. The user can also force the use of a specific form, if needed.

Independently from the underlying technology and the implementation templates, the advantage of all these a-causal modeling tools is the ability to compose complex systems simply by connecting their components. This is not a trivial advantage, or something limited to user interface issues: in fact, it is a very powerful method for creating *modular* simulators, in which components can be connected in various ways to change the topology of the system. The usefulness of this is

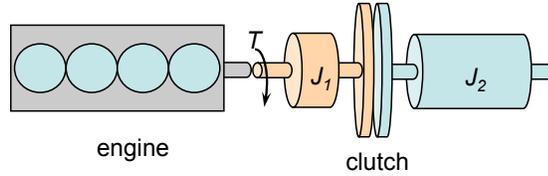


Figure 2.4: An example of a mechanical system whose structure changes with time

especially evident in powertrain modeling. For example, to create a detailed dynamic model of an automatic transmission, one should account for the fact that the system changes depending on which control actions are taken, which in a conventional (causal) modeling language means that a different set of equations should be created for each system configuration. Since these models may have more than ten degrees of freedom, writing several sets of equations can be a complex task, and running them all, switching appropriately between them, can be computationally expensive and give rise to issues with the numerical solution because of the fact that the system equations change with time. An elementary example can help to understand these issues: consider the physical system shown in Figure 2.4, composed by two inertias driven by an engine (in this case just a source of torque) and connected via a dry clutch. Three cases are possible:

1. When the clutch is engaged, it functions as a rigid connection: the result is equivalent to a single degree-of-freedom system, with inertia  $J_{tot} = J_1 + J_2$ . The system equation in this case is

$$(J_1 + J_2)\dot{\omega} = T \quad (2.12)$$

2. When the clutch is slipping, the system still has two degrees of freedom, but there is frictional force between the two plates of the clutch, which implies an exchange of torque between the inertias:

$$\begin{cases} J_1\dot{\omega}_1 = T - T_{cl} \\ J_2\dot{\omega}_2 = T_{cl} \end{cases} \quad (2.13)$$

3. When the clutch is disengaged, the two inertias rotate independently and each of them follows its own dynamics. No torque is transmitted between the two inertias. The system equation is:

$$\begin{cases} J_1\dot{\omega}_1 = T \\ J_2\dot{\omega}_2 = 0 \end{cases} \quad (2.14)$$

Obviously, the system can be in any of these three states during its functioning. The traditional modeling approach consists in writing the equations as it has just been done, then using one of them depending on the clutch state, which is defined by the clutch slip  $s = \frac{\omega_1 - \omega_2}{\omega_1}$ . Noticing that case 3 is just a particular case of case 2 with  $T_{cl} = 0$ , the discriminant between case 1 and the others is whether there is slip between clutch plates or not, i.e. whether  $s = 0$  or  $s \neq 0$ . The value of the slip depends on the dynamics of the system, therefore the switching between cases happens as a consequence of the system evolution.

Using an a-causal modeling approach, the system has to be modeled only once, using icons that resemble the physical components. The implementation of this model in Simulink (Figure 2.5) shows the two separate cases and how the respective differential equations are represented. The implementation in SimDriveline (Figure 2.6) uses blocks representing the same components of the physical system and does not need to include two separate submodels. The results of the two model implementations are obviously identical; however, it is apparent how the model built using physical modeling tools is easier to visualize and to build. This also means that this kind of model is easier to understand for someone who did not create it, making the process of modeling more efficient and easier to share. In the case of more complex models (with tens of possible different states, for example an automatic transmission), the ability to model the entire system at once and to leave to the solver the task to determine the correct set of equations is a very important advantage, drastically reducing the time necessary to develop a model.

One disadvantage of all object-based modeling tools is the false sense of security that they give to users: the fact that there is a block representing a physical component may obscure the level of detail and accuracy of the model implemented, and may lead inexperienced users to overestimate the power of these

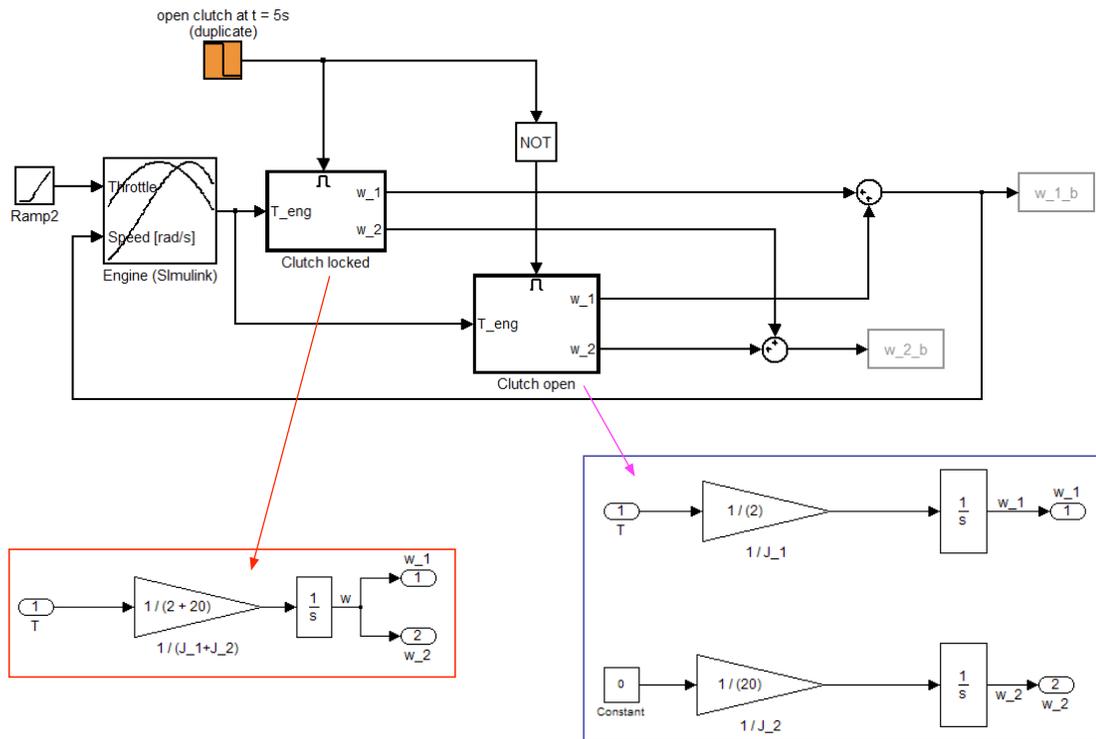


Figure 2.5: Implementation of the model of the system shown in Figure 2.4 using Simulink

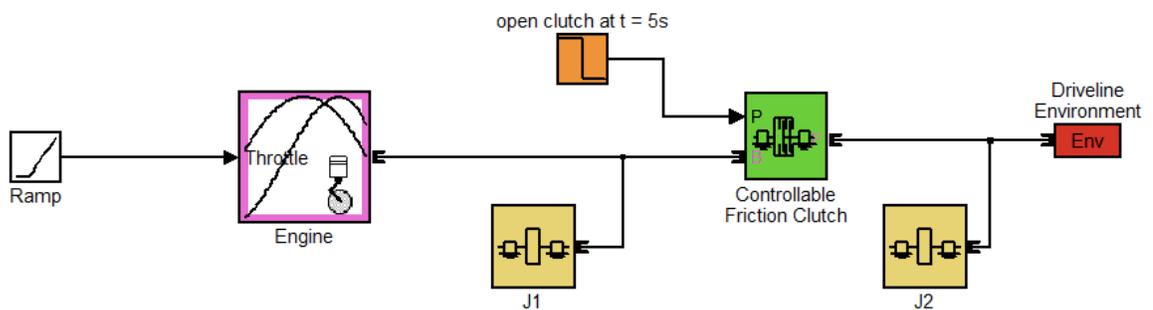


Figure 2.6: Implementation of the model of the system shown in Figure 2.4 using SimDriveline

tools. It is therefore important to consider the kind of model used in each block in order to understand which phenomena can be accurately modeled, and what simulation results should be expected.

### **2.3 Simulator implementation**

The model developed is purely longitudinal, in the sense that it does not account for any lateral or vertical motion; however, the simulator is compatible with typical vehicle handling models and has been implemented taking into account the possibility of future extension in this direction. For example, the wheels are implemented as individual blocks and not lumped together – this allows to easily extend the simulator to consider the lateral forces developed by each of them. Combining electric powertrain and traditional vehicle dynamics modeling can be interesting because the very fast response of electric traction machines (with respect to the dynamic range described) may allow to use them as a replacement for, or in addition to, safety devices such as anti-skid regulator, active differential, or advanced all-wheel-drive (AWD) strategies. Therefore, this can be seen as the longitudinal module of a complete vehicle dynamics simulator.

The software used for implementing the model is Simulink, and its specialized blockset SimScape is used as framework for the simulator, for the ability to introduce a-causal modeling templates that are very useful to create a more modular and open simulator, as mentioned in Section 2.2. The reason for choosing Simulink is due to the fact that allows for an easier implementation of the control strategy and for easier interface with the industry, being the de facto standard in the automotive sector.

All powertrain components in the simulator have been modeled using steady-state efficiency maps (the only efficiency information normally available). The blocks containing the models of the various components are mechanically connected by torque and speed information. The use of SimScape allows this information to be shared using only one connection line, representing a physical shaft between components. The practical consequence is that the overall powertrain

model can be assembled using self-contained blocks for each physical component, thus facilitating the reuse of the blocks for various powertrain configurations.

## 2.4 Powertrain components

### 2.4.1 Internal combustion engine

A conceptual sketch of the engine model is shown in Figure 2.7: it is a static model, which neglects crank-angle dynamics and torque oscillations due to the alternating inertia and combustion cycles.

The engine torque is applied to the crankshaft and flywheel, lumped together in a single rotational inertia, which is also subject to the load torque, coming from the rest of the powertrain. This constitutes the mechanical interface between the engine and the drivetrain. The torque that the engine generates is calculated using a table interpolation based on the maximum available torque at the current speed and the percentage of load desired  $\alpha$  (corresponding to throttle opening in traditional gasoline engines and to amount of injected fuel in Diesel engines). The fuel consumption is estimated using another table interpolation, as a function of torque and speed.

The torque is given by

$$T_{ice} = \alpha (T_{ice,max} - T_{ice,min}) + T_{min} \quad (2.15)$$

where  $T_{ice,max}(\omega)$  and  $T_{ice,min}(\omega)$  represent respectively the maximum torque and the friction torque, function of engine speed.

Given the throttle input and the measured crankshaft speed, the net torque is calculated and applied to the equivalent inertia, which represents the crankshaft and the flywheel. The output shaft is then used to connect the engine to the rest of the driveline components. Note that, being the torque map obtained with steady-state testing, it does not take into account the effect of the equivalent inertia, which justifies the torque being applied to it. This equivalent inertia is a rather abstract concept in the case of a piston engine: in fact, even at steady-state (constant nominal speed), the reciprocal motion involved in the engine functioning implies that

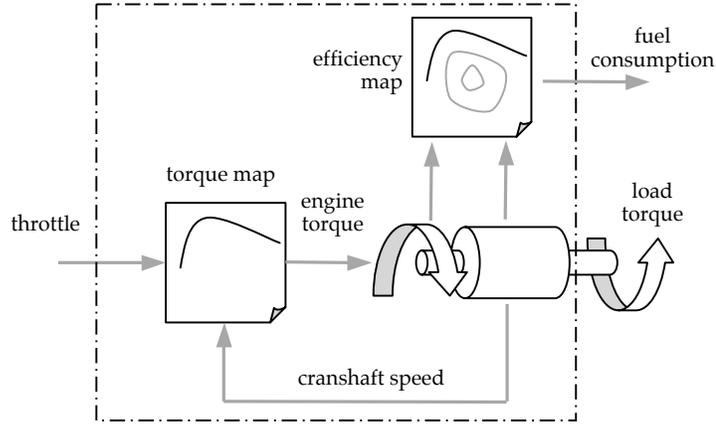


Figure 2.7: Engine model

some of the torque delivered by the fuel is used to accelerate the pistons from one cycle to the next. However, since the brake torque measured in a test bench is, in fact, the net torque delivered by the engine after moving its pistons, this model is in agreement with the way the maps are measured. The equivalent inertia is, therefore, a way of taking into account the delay associated with a significant change in the engine average speed.

### Scaling engine maps

Traditionally, internal combustion engines have been characterized using maps that relate torque, speed, and fuel consumption. Alternatively, the maps can be expressed using mean effective pressure, mean piston speed, and efficiency, which are rescaled measured of the former three [61].

The *mean effective pressure*  $p_{me}$  is defined as

$$p_{me} = \frac{4\pi}{V_d} T \quad (2.16)$$

where  $V_d$  is the total engine displacement and  $T$  is the engine torque.

The mean piston speed is

$$c_m = \frac{s_p}{\pi} \omega = \frac{s_p}{30} RPM \quad (2.17)$$

where  $s_p$  is the piston stroke,  $\omega$  the engine speed (in rad/s) and  $RPM$  the engine speed in rev/min.

The mechanical power is given by

$$P_{mech} = T\omega = \frac{V_d}{4\pi} p_{me} \pi \frac{c_m}{s_p} = \frac{V_d}{4s_p} c_m p_{me}. \quad (2.18)$$

The global engine efficiency is obviously the ratio of the output (mechanical) power to the input power, i.e. the chemical power of the fuel. This can be expressed as a function of the fuel mass flow rate  $\dot{m}_f$  and of the fuel lower heating value  $Q_{lhv}$ , which is the energy content per unit of mass of the fuel:

$$P_{fuel} = Q_{lhv} \dot{m}_f. \quad (2.19)$$

The efficiency is then

$$\eta = \frac{P_{mech}}{P_{fuel}} = \frac{T\omega}{Q_{lhv} \dot{m}_f}. \quad (2.20)$$

Just as the mechanical power can be expressed in terms of the mean effective pressure, the fuel power can be expressed using the concept of *mean available pressure*  $p_{ma}$ , which is proportional to the *available torque*, i.e. the torque that the engine would produce if its efficiency were unitary. The mean available pressure can be defined as follows:

$$p_{ma} = \frac{4\pi}{V_d} T_{avail} = \frac{4\pi}{V_d} \frac{P_{fuel}}{\omega} = \frac{4\pi s_p Q_{lhv} \dot{m}_f}{V_d \pi c_m} \quad (2.21)$$

and the efficiency can also be expressed as the ratio of  $p_{me}$  and  $p_{ma}$ :

$$\eta = \frac{P_{mech}}{P_{fuel}} = \frac{T\omega}{T_{avail}\omega} = \frac{p_{me}}{p_{ma}}. \quad (2.22)$$

Sometimes, instead of using the fuel flow rate  $\dot{m}_f$  or the overall efficiency  $\eta$ , the efficiency map of an engine is expressed in terms of the *brake specific fuel consumption* (BSFC), which is the amount of fuel consumed for a given amount of mechanical energy (work) produced, or the rate of fuel consumed for a given amount of mechanical power:

$$BSFC = \frac{m_f}{E_{mech}} = \frac{m_f}{P_{mech}t} = \frac{m_f/t}{P_{mech}} = \frac{\dot{m}_f}{P_{mech}} \quad (2.23)$$

*BSFC* and efficiency can be related as follows:

$$BSFC \cdot Q_{lhv} = \frac{Q_{lhv}\dot{m}_f}{P_{mech}} = \frac{1}{\eta} \quad (2.24)$$

and

$$\eta = \frac{1}{Q_{lhv}BSFC} \quad (2.25)$$

In other words, the brake specific fuel consumption is the reciprocal of the overall engine efficiency, scaled by the energy density of the fuel.

Engines produced by the same manufacturer and being part of the same family usually have the property that the relation between efficiency, mean effective pressure and mean piston speed are the same across all the members of the family. This means that their maps, when expressed in terms of these variables, look the same. The torque-speed-fuel consumption maps for each individual engine can be obtained from them, multiplying by the corresponding scaling factors (displacement, piston stroke, number of cylinders). For example, assume that the same engine module (piston, combustion chamber, head assembly) is used to produce a 4- and a 6-cylinder engine. If the maps of the smaller engine (subscript 4c) are known, the corresponding maps for the bigger one (6c) can be obtained as follows:

$$T_{6c} = p_{me} \frac{V_{d,6c}}{4\pi} = \frac{4\pi}{V_{d,4c}} T_{4c} \frac{V_{d,6c}}{4\pi} = \frac{V_{d,6c}}{V_{d,4c}} T_{4c} = \frac{6}{4} T_{4c} \quad (2.26)$$

$$\dot{m}_{f,6c}(c_m, p_{me}) = Q_{lhv}\eta(c_m, p_{me}). \quad (2.27)$$

This scalability property was applied better to older engines than it is for modern engines, where electronic control may introduce discontinuities in the maps, which do not scale according to these considerations. However, in cases when data for only one engine in the family is available, this is still the most accurate way of estimating unavailable maps (assuming the geometric scaling factors are known).

Additional relations that can be useful for rescaling engine maps are the following:

$$V_d = \frac{\pi}{4} n_{cyl} B_p^2 s_p = \frac{\pi}{4} n_{cyl} \frac{B_p^2}{s_p^2} s_p^3 = \frac{\pi}{4} n_{cyl} \left( \frac{B_p}{s_p} \right)^2 s_p^3 \quad (2.28)$$

where  $s_p$  is the piston stroke,  $B_p$  the bore (piston diameter),  $n_{cyl}$  the number of cylinders, and  $\frac{B_p}{s_p}$  is the bore-to-stroke ratio (a typical engine characteristic).

### Willans line model

The Willans line model<sup>4</sup> is an affine relationship between the mean effective pressure  $p_{me}$  and the mean available pressure  $p_{ma}$ , whose coefficients depend on the mean piston speed  $c_m$ :

$$p_{me} = p_{m0}(c_m) + \epsilon(c_m) p_{ma} \quad (2.29)$$

The advantage of such a model is that the coefficients  $p_{m0}(c_m)$  and  $\epsilon(c_m)$  are typically the same for engines of the same family, thus this model fits all of them. The terms  $p_{me}$  and  $p_{ma}$  are then scaled to obtain the torque and fuel consumption, using (2.16) and (2.21) respectively. Despite its simple form, this model is usually capable of fitting with good accuracy efficiency maps deriving from experimental data, and can also be used for electric machines [62]. The Willans model is useful when there is the need to express the efficiency map of an engine as an analytical function of the speed and torque, or to create a model that is completely size-independent and can be seamlessly rescaled to fit several engine sizes.

#### 2.4.2 Torque converter

The torque converter is a fluid coupling device that is used to transmit motion from the engine to the transmission input shaft. It is capable of multiplying the engine torque (acting as a reduction gear), and, unlike most other mechanical joints, provides extremely high damping capabilities, since all torque is transmitted through fluidodynamic forces rather than friction or pressure. It is traditionally

<sup>4</sup>named after the British engineer P.W. Willans, one of the pioneers in the early development of steam engines in the 19th century.

Figure 2.8: Schematic representation of a torque converter

used in vehicles with automatic transmissions to avoid the need of automatically engaging and disengaging a mechanical clutch (which is – or at least used to be – a very difficult control task, creating several drivability issues). In fact, a torque converter allows for large speed differences between its two shafts.

A torque converter (Figure 2.8) is composed by three elements: a pump, connected to the engine shaft, a turbine, connected to the transmission, and a stator which does move. The fluid in the torque converter is moved by the pump because of engine rotation, and drags the turbine and therefore transmits torque to the transmission. The torque at the turbine is generally higher than the torque at the pump (i.e. the engine torque), thanks to the presence of the stator. The torque difference is higher when the speed difference between the pump and the turbine is higher; at steady state, the two elements tend to rotate at the same speed and the torque difference tends to zero.

The torque converter model, which can be conceptually represented as in Figure 2.9, is based on a torque-speed map that for calculating the torques exerted by the fluid on the turbine and the pump.

In particular, torque characteristics are usually represented in graphical form, as graphs of torque ratio and capacity factor versus the speed ratio [63]. The speed ratio is

$$SR = \frac{\omega_t}{\omega_p}, \quad (2.30)$$

the torque ratio or multiplication ratio is

$$MR = \frac{T_t}{T_p} \quad (2.31)$$

and the capacity factor, which gives an idea of how much torque the torque converter can transmit, is defined as

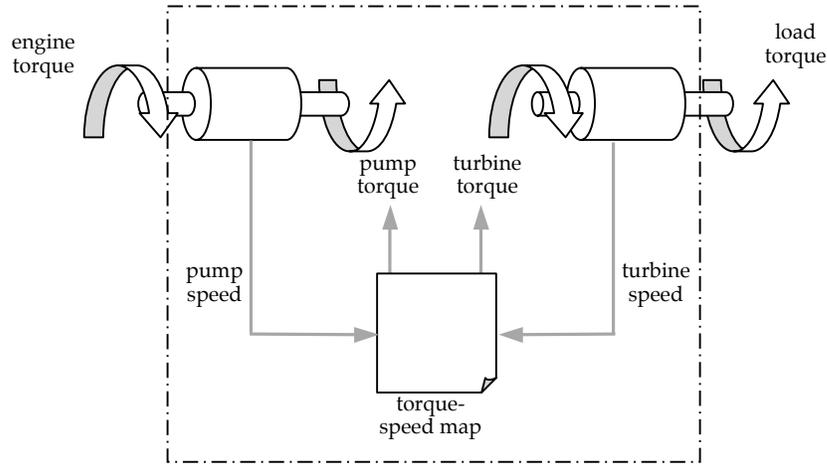


Figure 2.9: Torque converter model

$$K_{tc} = \frac{\omega_p}{\sqrt{T_p}} \quad (2.32)$$

In the vehicles modeled here, detailed data on the transmission components are not available; therefore, public-domain maps for torque converters of various size are used. The characteristic curves of a torque converter are shown in Figure 2.10.

The map can be easily replaced by an analytical model based on curve fitting, the Kotwicki model [64], described in the following.

The operation of the unlocked torque converter can be split into two phases: *multiplication mode* and *torque coupling mode*.

The multiplication mode occurs when the engine speed exceeds transmission input speed by a significant amount. As the name might suggest, the torque at the output shaft of the converter is actually larger than the input torque (i.e. the input torque is multiplied). The pump and turbine torques can be expressed as a quadratic fit of the speeds:

$$\begin{cases} T_p = T_{p,m} = t_{p1}\omega_p^2 + t_{p2}\omega_p\omega_t + t_{p3}\omega_t^2 \\ T_t = T_{t,m} = t_{t1}\omega_p^2 + t_{t2}\omega_p\omega_t + t_{t3}\omega_t^2 \end{cases}, \quad (2.33)$$

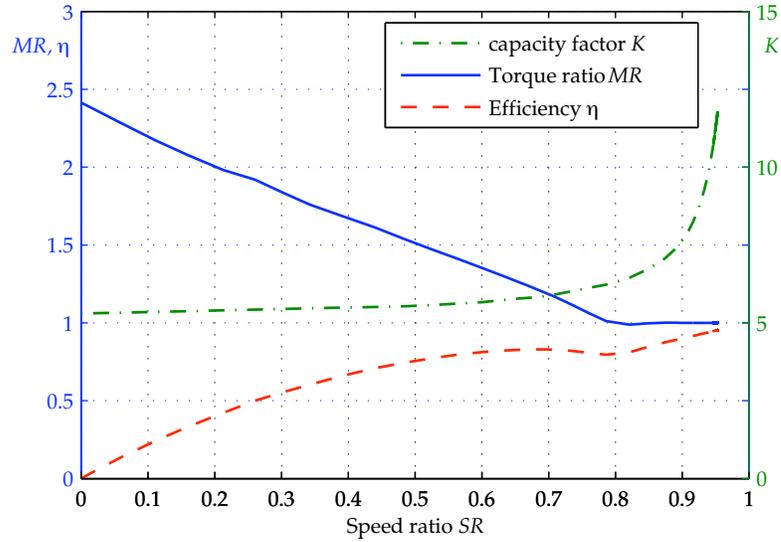


Figure 2.10: Torque converter map

where  $T$  indicates torque,  $\omega$  speed, the subscripts  $p$  and  $t$  refer respectively to the pump (sometimes called impeller) and the turbine, and  $t_{xi}$  are fitting coefficients obtained by fitting experimental data.

The coupling operating mode occurs when the engine speed and the transmission input speed are nearly equal (i.e., there is a small slip between the turbine and the pump). In this mode the input torque and output torque are equal to one another:

$$T_p = T_t = T_{cpl} = t_{tp1}\omega_p^2 + t_{tp2}\omega_p\omega_t + t_{tp3}\omega_t^2. \quad (2.34)$$

In a forward facing model (as shown in Figure 2.9), the turbine and impeller speed are given. A determination must be made as to which “mode” is active (coupling or multiplication), then the torques can be determined via the quadratic formulas given earlier. Coupling mode is detected by comparing the two speeds: coupling occurs when their difference is below a given threshold.

This basic model is fairly common and generally is accepted, but requires a total of 9 coefficients, which can be obtained using curve fit of tabulated data.

Kotwicki's model for the multiplication mode (2.33) can be rewritten in terms of the speed ratio  $SR$  as follows:

$$\begin{cases} T_p = T_{p,m} = \omega_p^2 \left( t_{p1} + t_{p2} \frac{\omega_t}{\omega_p} + t_{p3} \frac{\omega_t^2}{\omega_p^2} \right) \\ T_t = T_{t,m} = \omega_p^2 \left( t_{t1} + t_{t2} \frac{\omega_t}{\omega_p} + t_{t3} \frac{\omega_t^2}{\omega_p^2} \right) \end{cases} \quad (2.35)$$

and therefore the corresponding multiplication ratio and the efficiency are

$$MR = \frac{T_t}{T_p} = \frac{t_{t1} + t_{t2}SR + t_{t3}SR^2}{t_{p1} + t_{p2}SR + t_{p3}SR^2} \quad (2.36)$$

$$\eta = MR \cdot SR = \frac{t_{t1}SR + t_{t2}SR^2 + t_{t3}SR^3}{t_{p1} + t_{p2}SR + t_{p3}SR^2}. \quad (2.37)$$

In coupling mode, (2.34) becomes:

$$T_p = T_t = T_{cpl} = \omega_p^2 \left( t_{tp1} + t_{tp2}SR + t_{tp3}SR^2 \right) \quad (2.38)$$

and  $MR = 1$  by definition.

### 2.4.3 Gearing and differential

Gearing are purely mechanical components, with no inputs, outputs or controls. The external interfaces are two mechanical connections representing input and output shaft (or, using a more accurate terminology, base and follower shaft, referring to the shaft physical location without implying a direction of power flow). In the case of planetary gear sets, three mechanical connections are present instead of two: sun, carrier, and ring shaft.

The simplest model possible for a gearing only accounts for the speed and torque ratios, without considering the losses due to friction. Indicating with the subscripts  $B$  and  $F$  the base and follower shaft, and with  $g_{FB} = \frac{N_B}{N_F}$  the transmission ratio ( $N$  is the number of teeth of each gear), the lossless gear model is:

$$\begin{cases} \omega_F = g_{FB}\omega_B \\ T_F = \frac{T_B}{g_{FB}} \end{cases} \quad (2.39)$$

For energy analysis and in general for more accurate predictions, a lossy gear model is introduced, which takes into account power losses in the gearing. Given the fact that the speed ratio is fixed, being given by kinematic constraints, the power loss implies the reduction of the torque at the output shaft, taken into account by the gear efficiency  $\eta$ :

$$\begin{cases} \omega_{out} = g_{io}\omega_{in} \\ T_{out} = \eta \frac{T}{g_{io}} \end{cases} \quad (2.40)$$

In this case, the subscripts *in* and *out* refer to the shaft of input and output power flow, since the loss must reduce the output power; *in* and *out* can be either *B* (base) or *F* (follower). The identification of the shafts is based on the sign of the product  $T\omega$ , which is positive at the input shaft. The power loss is calculated as  $P_{loss} = \omega_{in}T_{in}(1 - \eta)$  and is always positive.

#### 2.4.4 Gearbox and transmission

Functionally, a gearbox is a gearing whose transmission ratio (and possibly other characteristics, such as efficiency) can change dynamically. The model implemented for the gearbox is the simplest possible, and consists in a lossy gear (see previous section) with variable gear ratio and variable efficiency (which depends on gear ratio, speed, and input torque). The variable gear ratio signal deriving from the gear selection index is filtered with a 1st order transfer function that simulates (albeit crudely) the delay involved in the actual procedure of gear shifting, that usually takes a few tenths of second to be completed. This model captures the essential functionality common to manual gearboxes and automatic transmissions, and can be used for both cases. For accurate drivability studies, which are not in the objectives of this dissertation, a complete transmission model (considering all the gears and their coupling) should be implemented.

#### 2.4.5 Wheels, brakes, and tires

The wheel represents the link between the powertrain and the external environment. Its model includes the motion of the wheel and the effect of the brakes, calculating the forces at the interface between tire and road surface. The tractive

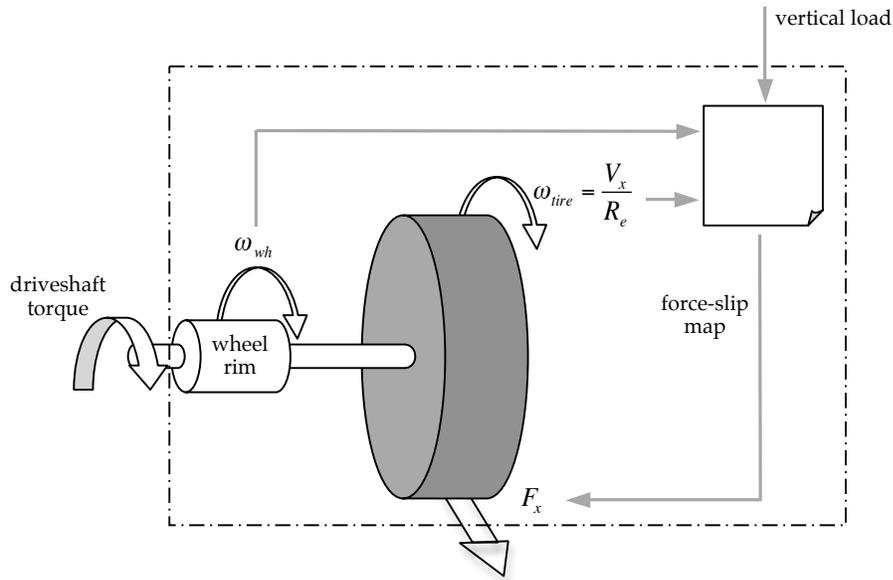


Figure 2.11: Wheel and tire model

force is calculated given the powertrain torque, the brake signal and the vertical load on the wheel.

### Quasi-static model

The simplest model is what could be defined a *perfect rolling* model, in which the torque applied to the wheel shaft is completely transformed into tractive force considering pure rolling motion between the tire and the soil, and neglecting tire deformation. The quasi-static model does not take explicitly into account wheel slip and relaxation length; however, it assumes that the dynamic response of the tire can be approximated by a first order delay and that the maximum force generated at the road/terrain interface is proportional to the vertical load on the wheel. The first order delay is useful to avoid numerical issues at very low vehicle speed, and to simulate (very approximately) the tire damping. The brakes are modeled as an additional torque that reduces the net torque acting on the tire. The brake torque is proportional to the brake input signal (which represents a normalized pressure). Therefore the net torque acting on the wheel is

$$T_{wh} = T_{shaft} - T_{brake} \quad (2.41)$$

where  $T_{shaft}$  is the torque at the driveshaft, and  $T_{brake} = \beta T_{brake,max}$  the braking torque (calculated as a function of the maximum available braking torque).  $\beta$  is the brake signal, equivalent to the normalized pedal position, and varies between 0 and 1.

However, when the vehicle is stopped, the torque generated by the brakes must only equilibrating the powertrain torque and the grade force on the vehicle. Therefore, independently from the value of  $\beta$ , it cannot exceed the value

$$T_{brake,stop} = T_{shaft} - R_e F_{grade} \quad (2.42)$$

where  $R_e$  is the effective rolling radius and  $F_{grade}$  is the quote of grade force acting on the wheel.

The effective tractive force generated is

$$F_x = \frac{1}{1 + \tau_x s} \frac{T_w}{R_e} \quad (2.43)$$

where  $F_x$  is the longitudinal force at the ground,  $T_w$  the torque on the wheel shaft,  $R_e$  the effective rolling radius, and  $\tau_x$  a time constant that introduces a delay between the torque and the force ( $s$  is the Laplace variable).

The wheel speed is obviously

$$\omega = \frac{V_{veh}}{R_e}, \quad (2.44)$$

being  $V_{veh}$  the longitudinal vehicle speed.

The value of longitudinal force is bounded by the vertical load acting on the wheel:

$$-F_z \mu_{x,max} \leq F_x \leq F_z \mu_{x,max} \quad (2.45)$$

where  $F_z$  is the vertical force on the wheel, and  $\mu_{x,max}$  is the peak value of the road/tire friction coefficient (usually around 0.8-0.9 for dry asphalt).

## Dynamic model

Another, more detailed model can be used to take into account the dynamic response of the tire. This is one of several models presented in [65] and is the simplest one dealing with transient tire response at low speed. The author calls it “semi non-linear” because the basic formulation is linear but a nonlinearity is introduced with a limitation on the time derivative of the tire deflection  $v$  at low speeds. This dynamic model uses the concept of relaxation length  $\sigma_k$ : this is the distance that the vehicle must travel before the force developed at the interface tire/ground reaches its steady-state value, and is such that

$$\frac{dv}{dt} + \frac{1}{\sigma_k} |V_{veh}| v = -V_{sx} \quad (2.46)$$

where  $v$  is the longitudinal tire deflection,  $V_{veh}$  the longitudinal vehicle speed, and  $V_{sx} = V_{veh} - R_e\omega$  the wheel slip velocity (which is what physically makes the tire produce force, but was implicitly set to zero in the static model).

The transient slip is calculated as  $\kappa' = \frac{v}{\sigma_k}$  and is used to calculate the longitudinal force  $F_x$  using the *magic formula*<sup>5</sup> [65]

$$F_x = D_x \sin (C_x \arctan [B_x \kappa' - E_x (B_x \kappa' - \arctan (B_x \kappa'))]). \quad (2.47)$$

The parameters  $\sigma_x$ ,  $B_x$ ,  $C_x$ ,  $D_x$ ,  $E_x$  that appear in these equations are obtained through curve fitting of experimental data and are a function of the specific tire, the road conditions, the vertical load on the tire. Their values should be provided by the vehicle or tire manufacturer.

For low values of speed ( $V_x < V_{low}$ ), the deflection  $u$  is modified to avoid unrealistic values; in particular:

$$\begin{cases} \frac{du}{dt} = 0 & \text{if } V_x < V_{low}, |\kappa'| > \frac{3D_x}{B_x C_x D_x} \text{ and } \left( V_{sx} + \frac{1}{\sigma_k} |V_x| u \right) u < 0 \\ \frac{du}{dt} + \frac{1}{\sigma_k} |V_x| u = -V_{sx} & \text{otherwise} \end{cases} \quad (2.48)$$

<sup>5</sup>This is the common name attributed to this mathematical expression, introduced by Prof. Pacejka (T.U. Delft) as a method for curve-fitting experimental maps of tire force as a function of slip and other parameters. Despite not being based on physical parameterization and requiring several coefficients, it is widely used for the almost “magic” capability of accurately describing these maps.

In the dynamic tire model, the brakes are modeled as a dry clutch, composed by two discs that come in contact: one of them is fixed (to the vehicle's frame), the other rotates with the wheel.

#### 2.4.6 Electric machines

The electric machines are modeled using a system-level approach similar to the one used for the engine, employing maps of torque and efficiency. Desired values of electrical power or torque can be used as a control input. Rotor inertia is the only dynamic element modeled, as the electrical dynamics in any kind of machine are much faster. The electrical power flow is modeled using standard Simulink blocks rather than physical modeling tools; therefore, depending on the desired control input, different models are used.

Case 1 (Figure 2.12.a): Electric power is the input (typical example: the generator of a series hybrid electric vehicle). The torque needed at the shaft of the machine is calculated using the electric power command and the efficiency map:

$$P_{mech} = T\omega = \frac{P_{elec}}{\eta(\omega, P_{elec})} \Rightarrow T = \frac{1}{\omega} \frac{P_{elec}}{\eta(\omega, P_{elec})} \quad (2.49)$$

Note that the efficiency map  $\eta(\omega, P_{elec})$  is given as a function of speed and electrical power (expressed as a percent of the maximum power).

Case 2 (Figure 2.12.b): Torque demand is the input (this usually happens in traction motors). In this case, the electric power must be calculated given the torque request:

$$P_{elec} = \frac{P_{mech}}{\eta(\omega, T)} = \frac{\omega T}{\eta(\omega, T)} \quad (2.50)$$

In both cases, the power is positive when it corresponds to positive tractive force or positive engine power, and negative otherwise. The power loss is positive in both cases:

$$P_{loss} = \begin{cases} P_{elec} - P_{mech} = \frac{\omega T}{\eta(\omega, T)} - \omega T = \omega T \left( \frac{1}{\eta} - 1 \right) = \omega T \left( \frac{1-\eta}{\eta} \right) & \text{motoring, } \omega T \geq 0 \\ |P_{mech}| - |P_{elec}| = P_{elec} - P_{mech} = \eta\omega T - \omega T = -\omega T (1 - \eta) & \text{generating, } \omega T < 0 \end{cases} \quad (2.51)$$

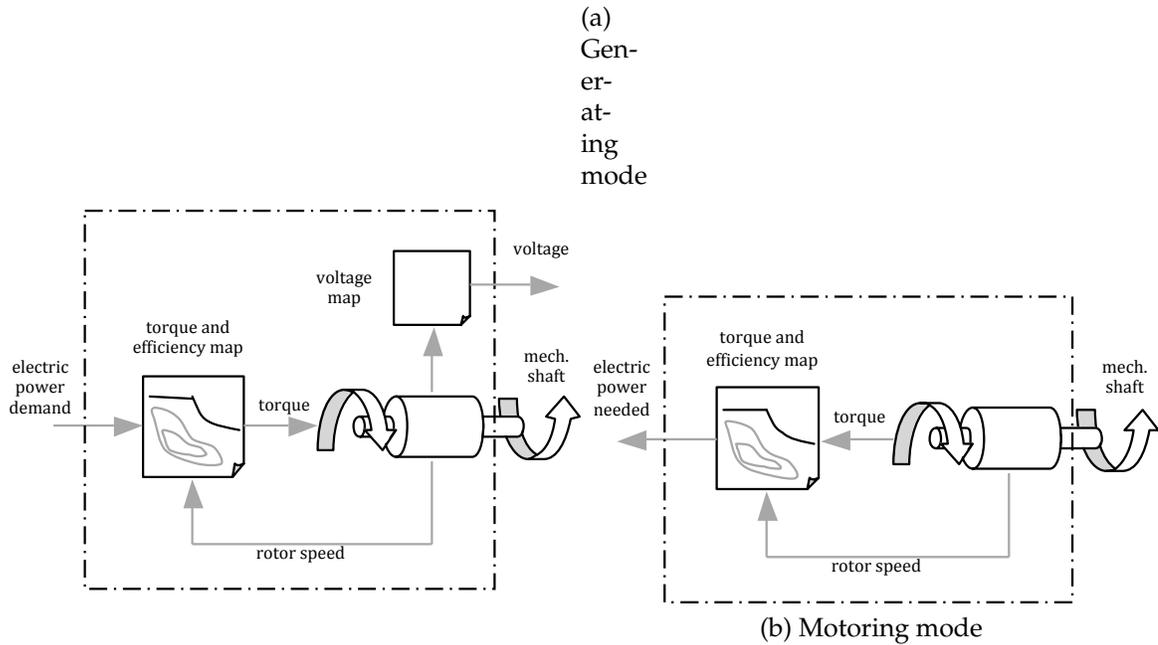


Figure 2.12: Electric machine model

### 2.4.7 Energy storage systems

Electrical energy storage systems such as batteries and capacitors are key component of hybrid vehicles. A variety of models have been proposed to evaluate their interaction with the rest of the powertrain; however, for fuel consumption and performance evaluation at a vehicle level, a simple circuit model is sufficient.

### 2.4.8 Batteries

Accurately modeling battery dynamics in hybrid electric vehicles is important and, unfortunately, it is not easy. The reason is that the main variables that characterize battery operation, i.e. state of charge, voltage, current, and temperature, are dynamically related to each other in a highly non-linear fashion. In general, the objective of the battery model in a vehicle simulator is to predict the change in state of charge given the electrical load.

The state of charge (SOC) is defined as the amount of charge stored in the battery, relative to the total charge capacity:

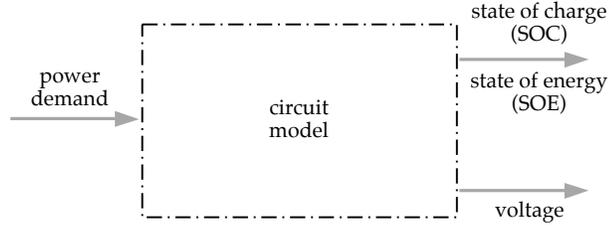


Figure 2.13: General model of energy storage system

$$SOC(t) = \xi(t) = \frac{\int_0^t I(\tau) d\tau}{Q_{batt}} \quad (2.52)$$

where  $Q_{batt}$  is the amount of charge that the battery can accept, i.e. its *charge capacity* or simply *capacity*, and  $\int_0^t I(\tau) d\tau$  is the amount of charge actually stored in the battery. Calculating the state of charge given the current is relatively straightforward, if the capacity is assumed to be a constant, known parameter. In reality, the battery capacity changes according to several parameters, mainly the magnitude of current and the age of the battery, but both these effects can be neglected for a model used for driving cycle evaluation: this does not introduce any modeling error, but simply implies a slightly different definition of state of charge, given in terms of nominal capacity rather than actual capacity.

The dependence on aging does not affect battery performance in the short term, but is only apparent over a long period of time, that exceeds any typical power-train/vehicle simulation horizon (if needed, it is possible to evaluate the effect of an aged battery by reducing the value of the capacity).

The battery voltage is a function of current  $I$  and state of charge  $\xi$ :

$$V_L = V_{oc}(\xi) + V_{circ}(I) \quad (2.53)$$

where  $V_L$  is the load voltage at the battery terminals,  $V_{oc}$  is the open circuit voltage, i.e. the voltage of the battery when it is not connected to any load ( $I = 0$ ), and

$V_{circ}$  the tension drop in the battery circuit, due to Ohmic losses and other electrochemical phenomena that make the terminal voltage different from the open circuit voltage.

The state of energy (SOE) of a battery is defined as the amount of energy stored, relative to the maximum amount of energy that the battery can hold. The amount of energy in a battery is equal to the product of the charge and the voltage; the maximum energy stored is thus

$$E_{batt} = Q_{batt} V_{oc,max} \quad (2.54)$$

and therefore the state of energy is

$$SOE(t) = \zeta(t) = \zeta(t) \frac{V_{oc}(t)}{V_{oc,max}}. \quad (2.55)$$

The simplest dynamic model of a battery is a circuit like the one in Figure 2.14. The series resistance  $R_0$  represents the Ohmic losses due to actual resistance of the wires and the electrodes and also to the dissipative phenomena that reduce the net power available at the terminals; the resistance  $R_1$  and the capacitance  $C_1$  are used to model the dynamic response of the battery. This model is a first-order approximation; the values of the parameters are estimated using curve fitting of experimental data, and are generally variable with the operating conditions (temperature, state of charge). Other models of the same kind, with more R-C branches in series, can be used if more accuracy is required. The number of parameters to be identified increases with the model accuracy.

The equations of the circuit in Figure 2.14 are:

$$V_L = V_{oc} - R_0 I - \sum_{i=1}^n V_i \quad (2.56)$$

$$C_i \frac{dV_i}{dt} = I - \frac{V_i}{R_i} \quad (2.57)$$

where  $n$  is the order of the dynamic model considered, i.e. the number of R-C branches. In the example shown,  $n = 1$ . The capacitance  $C_i$  and the resistance  $R_i$

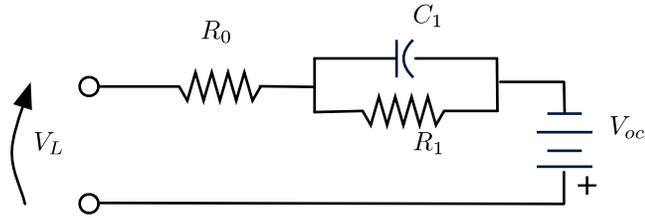


Figure 2.14: Battery circuit model

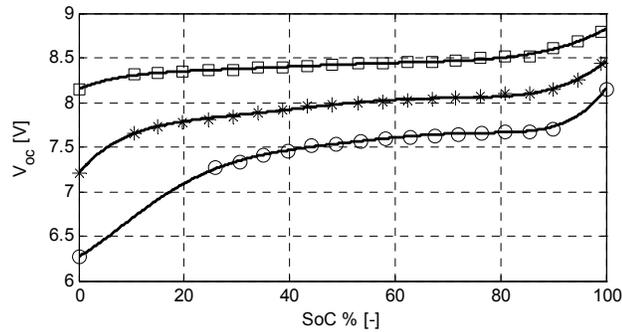


Figure 2.15: Open circuit voltage vs. state of charge for a Ni-MH battery [3]. The top and bottom curves correspond to charge and discharge (at 0.1C), the middle one is an average.

can, in principle, change with the direction of the current (charge or discharge) and with other operating conditions, such as temperature.

The curve that shows the variation of the open circuit voltage  $V_{oc}$  with the battery state of charge is called *discharge characteristic* and a typical example is shown in Figure 2.15. These curves are obtained by charging and discharging a battery at constant current, and depend on the value of the current. It is common practice to refer to the value of the current as a fraction of the battery capacity in Ah: for example, if the capacity is 6.5 Ah, a current of 1C corresponds to 6.5 A, 10C to 65 A, 0.1C to 0.65A. Steady-state characteristics of the battery, such as the  $V_{oc}$ -SOC curve or the value of rated capacity, are typically obtained using a current of 0.1C.

The magnitude of the current has an important effect on the discharge characteristic and on the capacity of the battery: the higher the current, the lower the amount of charge that can be extracted from the battery.

A fundamental component of the controller for a hybrid electric vehicle is an algorithm that provides an estimate of the battery state of charge given available measurements of terminal voltage and current. Apparently, integration of the current is sufficient to calculate the state of charge; however, in real-world applications, measurement errors and noise render the integration of the current a non-trivial task, because the result has the tendency to drift (the measurement noise has non-zero mean). Therefore, a more complex algorithm is needed, such as the one described in [3], which combines current integration with estimation of open circuit voltage (whose relation with the state of charge provides an alternative estimate).

An important issue related to battery usage in hybrid electric vehicles is their aging, due to the aggressive loading cycles to which they are subjected. Battery aging manifest itself as loss of capacity and increase of internal resistance [66, 67], and can reduce vehicle performance; recent and ongoing research is devoted to determine a suitable model that can predict the amount of residual life given the loading cycles [68, 66, 67, 69, 70].

#### 2.4.9 Capacitors

The capacitor pack is composed by several cells (single capacitors) in series and in parallel; for a simple, slow-dynamics model, each cell can be represented as a capacitance in series with a resistor. The capacitance models the ion accumulation, while the resistance accounts for the losses in the electrolyte. The entire pack can be modeled with the circuit shown in Figure 2.16, characterized by the following electrical equations:

$$R_{eq} = \frac{N_S}{N_P} R_{sing} \quad (2.58)$$

$$C_{eq} = \frac{N_P}{N_S} C_{sing} \quad (2.59)$$

$$V_L = R_{eq}I + V_C \quad (2.60)$$

$$\frac{dV_C}{dt} = \frac{I}{C_{eq}} \quad (2.61)$$

where  $V$  is the voltage,  $I$  the current,  $N_S$  and  $N_P$  are the number of cells in series and in parallel (respectively),  $R_{eq}$  is the equivalent resistance of the pack,  $C_{eq}$  the equivalent capacitance of the pack,  $R_{sing}$  the internal resistance of a single cell, and  $C_{sing}$  the capacitance of each cell.

The circuit equations can be written in the frequency domain as:

$$\begin{cases} V_L = R_{eq}I + V_C \\ sV_C = \frac{1}{C_{eq}}I \end{cases} \quad (2.62)$$

and therefore the transfer function that gives the load voltage as a function of the current is

$$\frac{V_L}{I} = \left( R_{eq} + \frac{1}{C_{eq}s} \right) = \frac{R_{eq}C_{eq}s + 1}{C_{eq}s} \quad (2.63)$$

By definition, the state of charge  $\xi$  is defined as the amount of charge stored in the capacitance, with relative to the maximum acceptable value; similarly, the state of energy  $\zeta$  is the amount of energy stored in the capacitance, relative to maximum energy that it can accept. Both quantities are related to the voltage across the capacitance:

$$SOC(t) = \xi(t) \doteq \frac{Q_{cap}(t)}{Q_{cap,max}} = \frac{CV_C(t)}{CV_{C,max}} = \frac{V_C(t)}{V_{C,max}} \quad (2.64)$$

$$SOE(t) = \zeta(t) \doteq \frac{E_{cap}(t)}{E_{cap,max}} = \frac{\frac{1}{2}CV_C^2(t)}{\frac{1}{2}CV_{C,max}^2} = \frac{V_C^2(t)}{V_{C,max}^2} = \xi^2(t) \quad (2.65)$$

#### 2.4.10 Power electronics and electric bus

Electric bus and power electronics for electric machines and energy storage devices are considered in the model as devices that absorb part of the power flowing

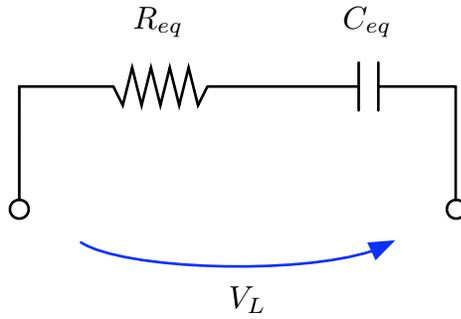


Figure 2.16: Circuit model of supercapacitor pack.

across them; efficiency maps (function of voltage and current) are used to calculate the power losses occurring in these elements.

#### 2.4.11 Engine accessories

The engine accessories are all secondary the accessories directly powered by the engine, such as air conditioning, compressor for pneumatic suspensions/brake, power steering, and alternator (for small electric loads). A simplified modeling approach is used, for the lack of detailed data and load cycles for all the components, and considering that the total accessory power is small compared to the tractive power that the engine delivers. For each accessory, the torque at the shaft is calculated as

$$T = \frac{1}{\eta} \frac{P_{net}}{\omega} \quad (2.66)$$

where the power comes from the load cycle, and the efficiency is calculated from power using a look-up table, curve fitting, or constant value (depending on the data availability).

#### 2.4.12 Auxiliary loads

The auxiliary loads model is used to generate a mechanical load cycle that mimics the power requirements of auxiliary loads for the vehicle. Auxiliary loads are

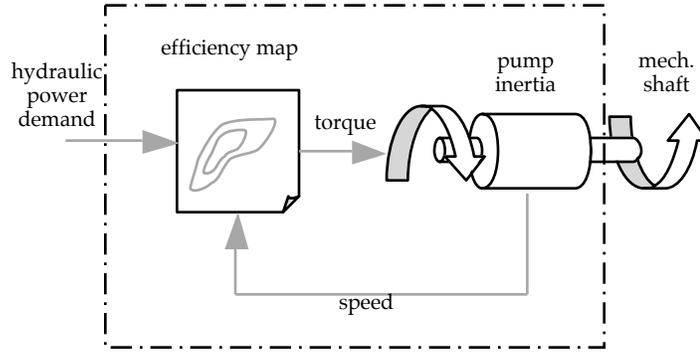


Figure 2.17: Hydraulic pump model

present in the refuse collection truck, and are the accessories used for refuse collection tasks, such as loading, packing, and dumping mechanisms. These are hydraulic devices driven by two identical pumps, one for the loading mechanisms and the other for packing and dumping. The block input (Figure 2.17) is the net hydraulic power required, and possibly the pressure in the circuit, obtained from data collected on a prototype vehicle; the output is the mechanical power at the shaft of each pump. In the prototype, these are both belted to the engine, but in principle it is also possible to connect the mechanical shafts to one or two electric motors. The main objective of the model is to describe the behavior of the hydraulic pumps, using efficiency and performance data provided by the manufacturer.

The load torque on the pump is calculated from the hydraulic power request  $P_{hyd}$  and the standby torque  $T_0$  as follows:

$$T_p = \min \left( \frac{1}{\eta(\omega, p_{hyd})} \frac{P_{hyd}}{\omega}, T_0 \right) \quad (2.67)$$

where the pressure  $p_{hyd}$  is estimated as the ratio of hydraulic power and flow  $Q_{hyd}$ :  $p_{hyd} = \frac{P_{hyd}}{Q_{hyd}(\omega)}$ . This approach means that the torque is normally calculated using the efficiency map, but it is bounded by the value of standby torque, which is the torque needed to spin the pump without producing any net power. The efficiency map, the flow-speed characteristic  $Q_{hyd}(\omega)$  and the standby torque are obtained

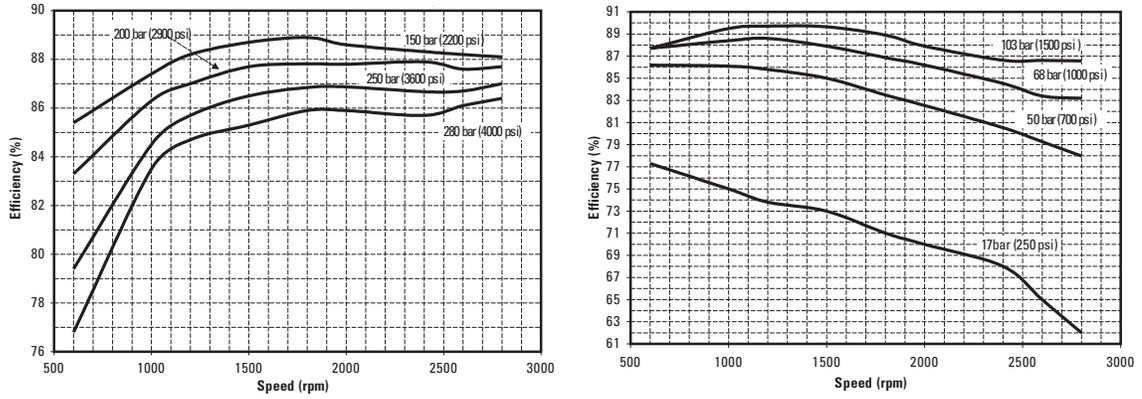


Figure 2.18: Efficiency map of the hydraulic pump (Eaton 062 ADU [4])

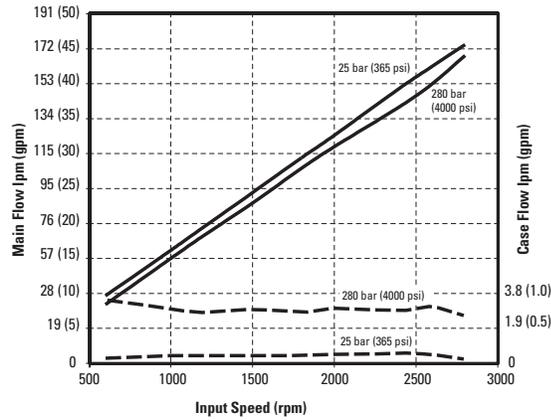


Figure 2.19: Flow speed characteristic of the hydraulic pump [4]

using data from the manufacturer’s manual [4], reported in Figure 2.18, 2.19, and 2.20.

### 2.4.13 Vehicle dynamics

As implemented, the vehicle dynamics model is rather simple and represents the longitudinal motion of the vehicle as well as the longitudinal load transfer between axles.

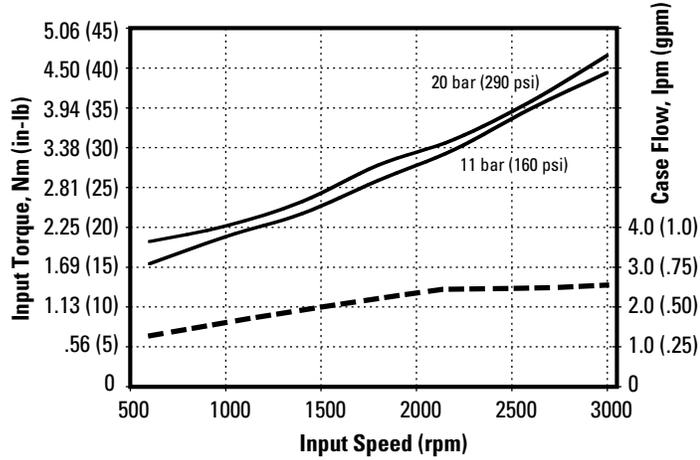


Figure 2.20: Standby torque curves [4]

### Longitudinal motion

The tire model predicts the force generated at the road/wheel interface by each individual tire due to traction and braking torque; the rolling resistance is introduced as an external force acting on the vehicle, together with the aerodynamic force. The resulting equilibrium equation is

$$m\dot{V}_{veh} = \sum F_{tire} - F_{roll} - F_{aero} - F_{grade} \quad (2.68)$$

where:

$m$  is the vehicle mass (variable with time, since the payload on the vehicle may increase or decrease);

$V_{veh}$  is the vehicle longitudinal velocity;

$F_{tire}$  represents the tractive and braking force generated by each tire;  $\sum F_{tire} = F_{trac}$  is the total traction force (see also Section 2.1.1) ;

$F_{roll} = r_0 mg + r_1 mg V_{veh}$  is the rolling resistance;

$F_{aero} = \frac{1}{2} \rho_{air} A_f C_d V_{veh}^2$  is the aerodynamic resistance;

$F_{grade} = mg \sin \alpha$  is the grade resistance (the road slope angle is positive when the vehicle is going uphill; the resistance becomes negative, i.e. makes the vehicle accelerate, when ).

Note that, since the inertia of the various driveline components is taken into account in the respective models, there is no need to account for the rotational inertia using a mass equivalent.

The coefficients of aerodynamic resistance and rolling resistance are measured using a coast-down test, consisting in letting the vehicle decelerate without using the brakes nor the powertrain to generate tractive effort ( $F_{trac} = 0$ ). In these conditions, the deceleration is due only to the rolling and aerodynamic resistance: thus, measuring the speed and deceleration and using (2.68), it is possible to derive experimentally a relation between the sum of the resistances to the vehicle speed. This relation is then fitted as a quadratic function of the speed:

$$F_{roll+aero} = A + BV_{veh} + CV_{veh}^2. \quad (2.69)$$

Assuming that the quadratic term is due to the aerodynamic resistance while the constant and linear term are due to rolling resistance, the respective coefficients are easily identified (note that the lumped resistance  $F_{roll+aero}$  defined by the coefficients  $A, B, C$  is perfectly acceptable for modeling a specific vehicle, but it is not generalizable, unlike the more physical description in which  $F_{aero}$  and  $F_{roll}$  depend on vehicle parameters).

### Longitudinal load transfer

The vertical load  $F_{z,i}$  acting on the  $i$ -th tire affects the longitudinal and transversal forces that the tire can generate. When the vehicle is at rest,  $F_{z,i}$  only depends on the total vehicle weight and the center of gravity (CG) position; during vehicle motion, inertia force acts on the vehicle at the height of CG, introducing a moment that is balanced by vertical load transfer between front and rear axles. Also, the presence of grade modifies the static load distribution (but not the load transfer, because the inertia force is parallel to the ground).

The model for the load on each axle (perpendicular to the ground) is therefore:

$$\begin{cases} F_{zf} = F_{zf0} \cos \alpha - m\dot{V} \frac{h_{CG}}{a+b} - mg \sin \alpha \frac{h_{CG}}{a+b} & \text{(front)} \\ F_{zr} = F_{zr0} \cos \alpha + m\dot{V} \frac{h_{CG}}{a+b} + mg \sin \alpha \frac{h_{CG}}{a+b} & \text{(rear)} \end{cases} \quad (2.70)$$

where:

$F_z$  is the vertical load on the wheels;

the subscripts  $f, r$  refer respectively to the front and rear axles, while the subscript 0 denotes the value when the vehicle is at rest on a flat surface (nominal load repartition between the axles);

$h_{CG}$  is the height of the center of gravity from the ground;

$a$  is the horizontal distance of the center of gravity from the front axle, and  $b$  is its distance from the rear axle.

Since no lateral dynamics is considered in the model, the load on each axle is equally distributed between wheels on the two sides of the vehicle. If the vehicle has more than two axles, but they are grouped such that two “sets” are identifiable, one including the front axle(s) and the other the rear axle(s), then the equations are still valid and it is possible to subdivide the load equally among axles in the same “set” (an example of this is a three-axle truck with twin rear axles). In this case,  $a$  and  $b$  are defined referring to the geometric center of the set of axles (the middle point of a twin axle). In other words, if the longitudinal distance between axles in the same set is small with respect to the distance between axle sets, the difference in load on each axle of the set due to moment balancing is negligible. If this is not the case, then a more complex model that takes into account the vertical stiffness associated to each axle could be used to determine the load distribution.

#### **2.4.14 Driver**

The driver model determines the position of accelerator and brake pedal for tracking the prescribed velocity profile. It is based on a PID controller that generates signals between -1 and +1. Positive values mean that the measured speed is lower than the desired one, therefore are intended as accelerator signals ( $\alpha$ , between 0 and 1); negative values mean the opposite and represent braking signals. The sign of the braking signal is reversed so that the brake signal  $\beta$  is also between 0 and 1.

## 2.5 Model validation: refuse collection vehicle

The original application of the vehicle model presented in this section was the refuse collection vehicle that represents the first case study in this dissertation (see Section 1.6 and Chapter 4). For this reason, the model was built keeping in mind specific details that pertain to heavy duty vehicles, such as accessories and variable payload; the validation of the model makes use of experimental data collected on a prototype of the refuse collection vehicle built by Oshkosh Corp. and tested at NREL facilities.

The vehicle modeled is a series hybrid electric truck with the architecture shown in Figure 2.21. It is characterized by the use of supercapacitors as energy storage device, by the presence of two traction motors (one for each of the twin rear axles), and by the fact that the engine drives mechanically not only the generator and the secondary accessories (air conditioning compressor, 12-V alternator, air brakes compressor etc.), but also the PTO (power takeoff) accessories for the refuse collection operations. These are two hydraulic pumps that drive the loading, packing, and dumping mechanisms.

Important outcomes of the model are the fuel consumption over a given driving cycle, the distribution of energy losses, the vehicle acceleration capability, and in general its dynamic response.

The model has been validated using special test cycles developed on purpose [51] with the objective of representing typical operating conditions for a refuse vehicle. The drive cycles include velocity profiles as well as load profiles (i.e., hydraulic power needed to load, pack, and dump refuses) and payload (amount of additional weight due to refuse collection during the cycle). Five standard cycles, shown in Figure 2.22, are used: *Approach* and *Return* cycles are highway cycles (from and to the deposit), *Route 1*, *Route 2* and *Route 3* represent three different phases of neighborhood operation. Together, the five cycles cover a significant range of typical vehicle operation.

The validation procedure has been conducted by focusing on one component or subsystem at a time and expanding gradually. Each component or subsystem is

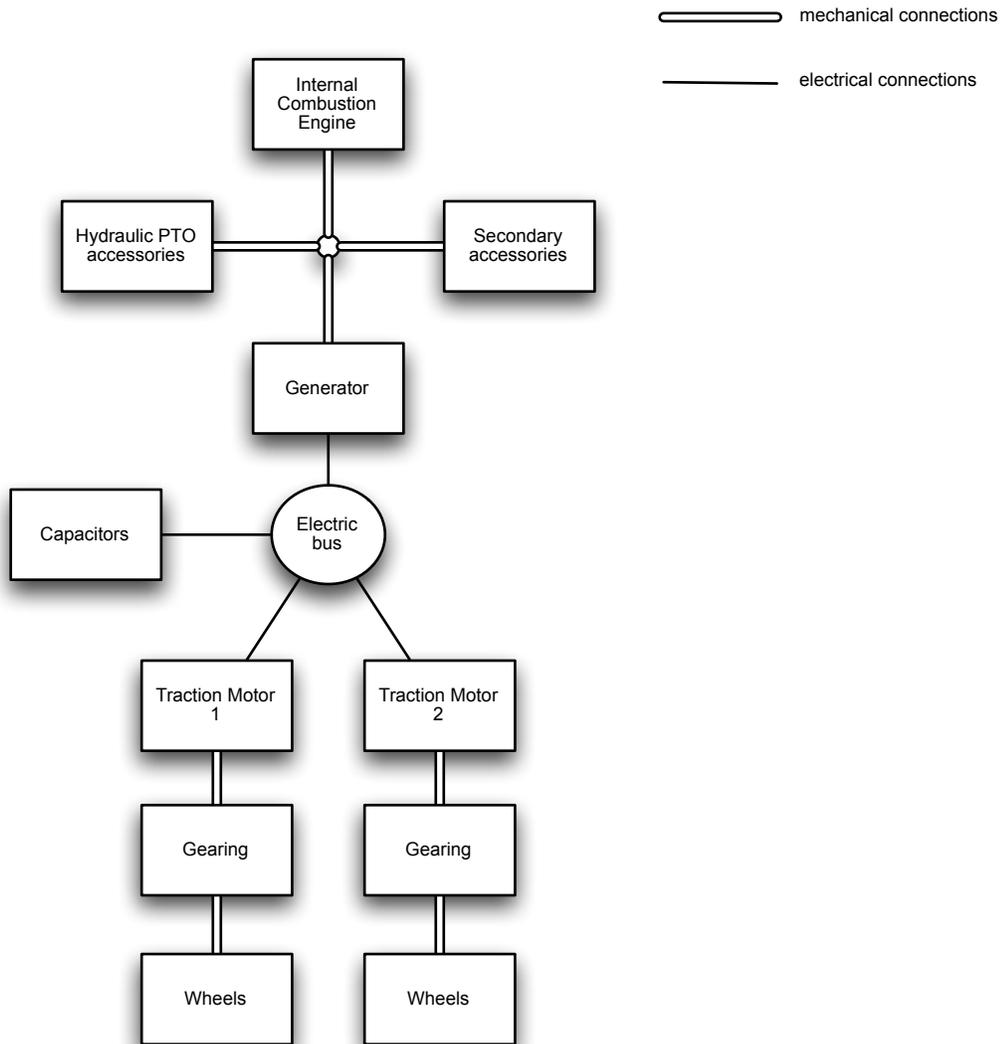


Figure 2.21: Architecture of the series hybrid electric prototype

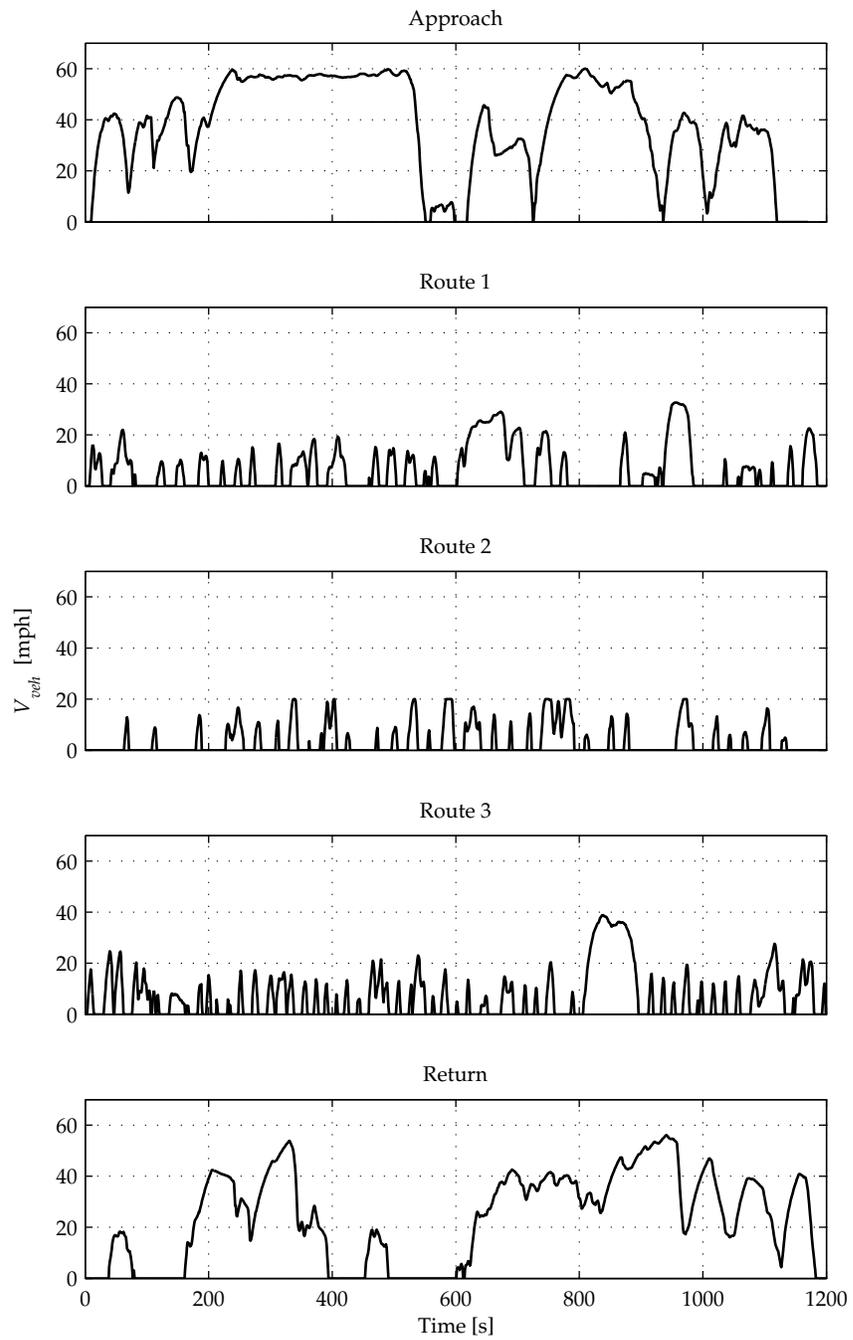


Figure 2.22: The five driving cycles used for model validation

validated by comparing the outputs of the simulator with the corresponding measured quantity, assuming that the input to the simulator is the same as the input quantity measured on the test vehicle. “Inputs” and “outputs” in a physical component are somewhat arbitrary concepts; in this context, the terminology refers to the causality of each model, therefore the way the simulator is created affects the choice of what is considered an input or an output. Control signals, on the other hand, are always inputs for both the physical systems and their models: whenever possible, they are used as the common input for validation.

It should be noted that most of the “measured” signals cited in the following are not direct measurements, but estimates provided by the various controllers (e.g., engine torque, generator current). They are still indicated as “measured” to distinguish them from simulation results, and represent the closest approximation to reality available for validation.

In several of the figures in this section, the actual values of the measurements are not reported because they are proprietary information.

### **2.5.1 Vehicle dynamics and road load**

In order to validate the model of longitudinal vehicle dynamics and road load, the torque measured at the traction motors is assumed to be delivered to the drive-line, and the corresponding vehicle speed is calculated and compared to the actual measurement. The brake pressure is not measured: the only information available regarding braking is whether the driver is pushing the pedal or not. Therefore, the brake signal is calculated using the driver model, i.e. using velocity feedback, and applied only during the time periods in which the actual brakes were on.

The validation consists in comparing the calculated vehicle speed with the measured value, as shown in Figures 2.25 – 2.26.

### **2.5.2 Supercapacitors**

The objective of the supercapacitor model is to predict the variation of state of energy (i.e., voltage) given the power exchanged with the electric bus. The available measurements are current and terminal voltage. To validate the model, the

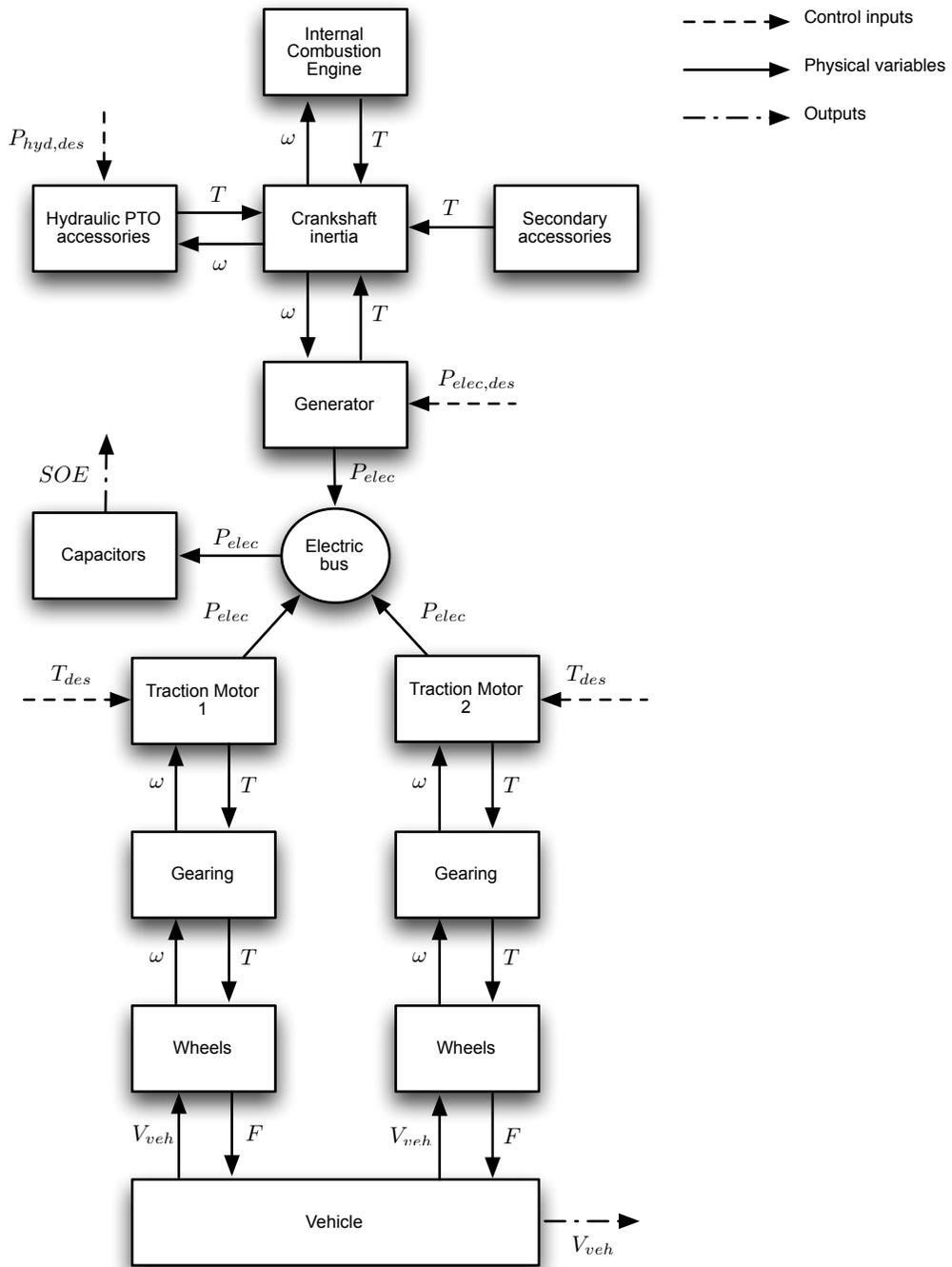


Figure 2.23: Powertrain architecture with model causality

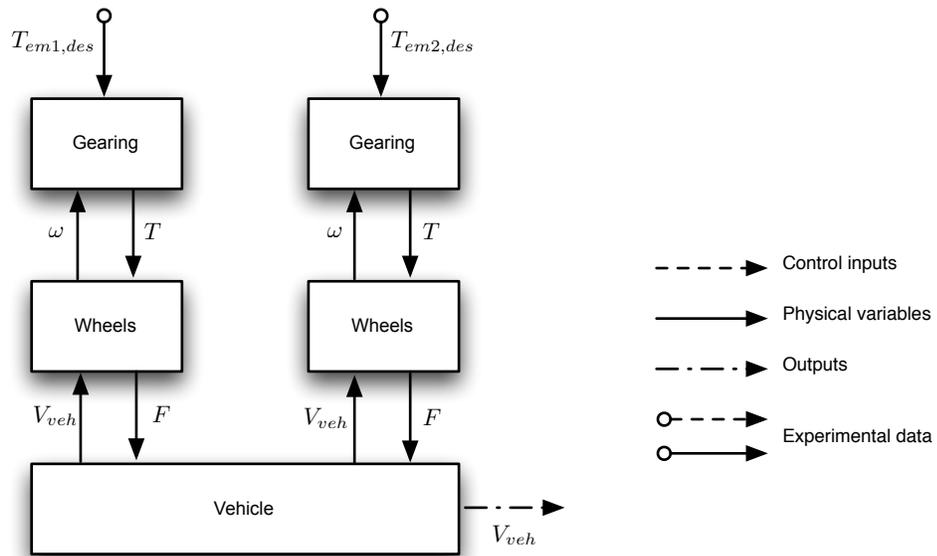


Figure 2.24: Scheme of the vehicle dynamics validation procedure

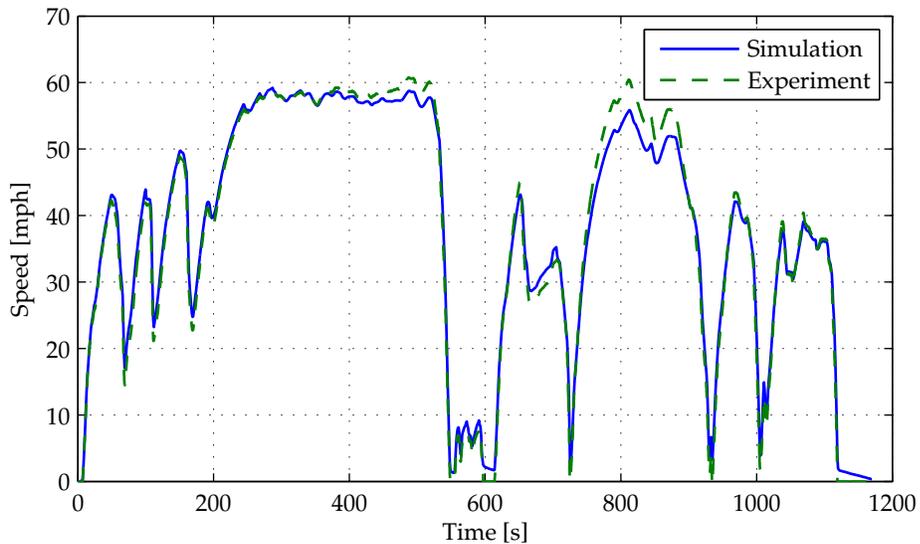


Figure 2.25: Comparison between calculated and measured vehicle speed (*Approach* cycle). The agreement between simulation and experiment is apparently good, except for the acceleration around 800 s, where the vehicle speed is very high and the model appears to slightly underestimate the resistances.

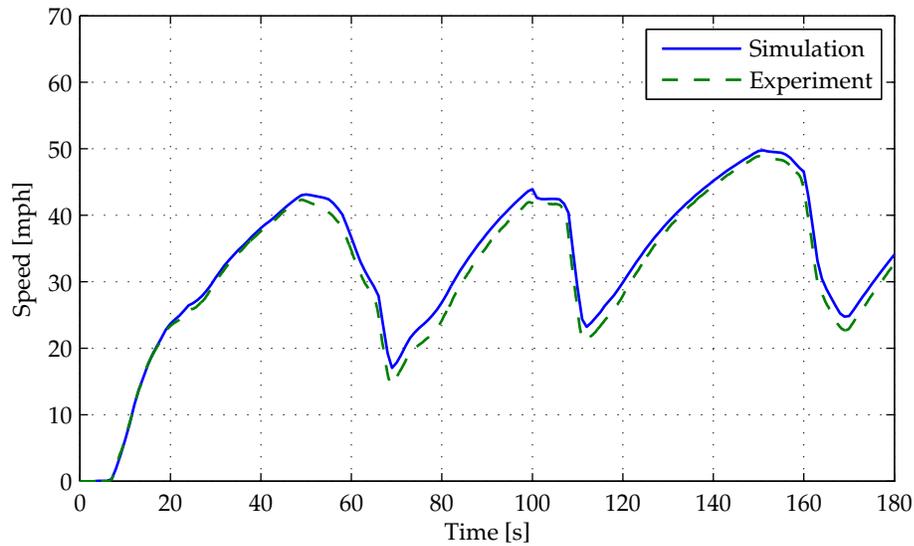


Figure 2.26: Comparison between calculated and measured vehicle speed (*Approach cycle, detail*)

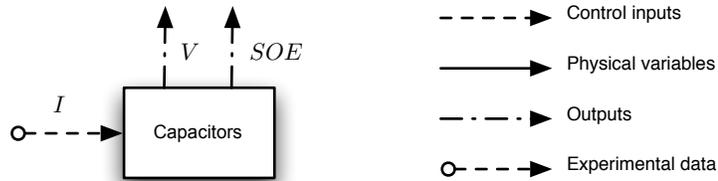


Figure 2.27: Validation scheme of capacitor model

supercapacitor block is fed with the measured current and the voltage is calculated using the circuit model presented earlier (2.4.9).

### 2.5.3 Traction motors

The external information experimentally available for the electric machines is the torque command to each traction motor and the current and voltage (hence, the electric power) at the generator and the supercapacitors. As shown in Figure 2.29,

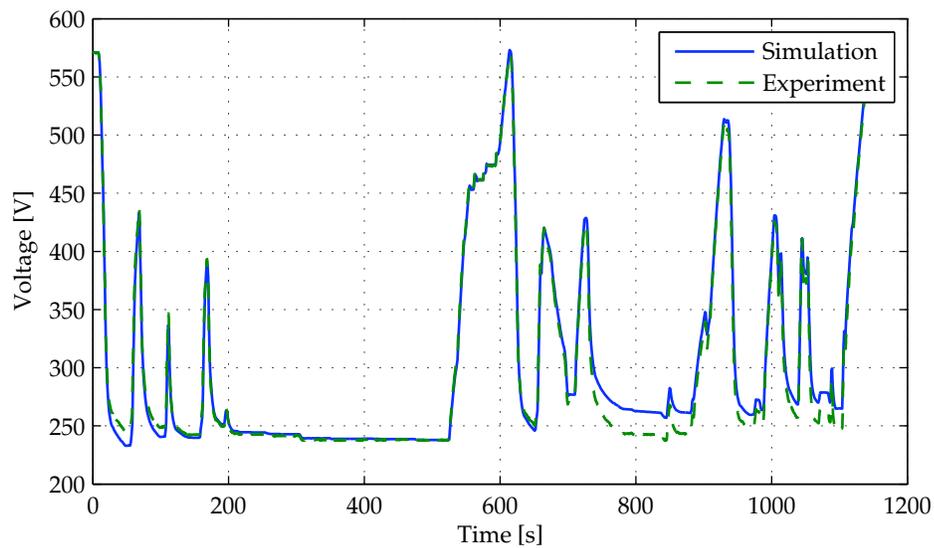


Figure 2.28: Comparison between calculated and measured voltage at the terminals of the supercapacitor pack (cycle *Approach*). The agreement is very good, despite the use of a simple first-order circuit model for the capacitors. The parameters of the circuit model (capacity, resistance) have been optimized using curve fitting of experimental voltage measurements (obviously, the one shown here is a validation data set; the calibration was done using a different experiment)

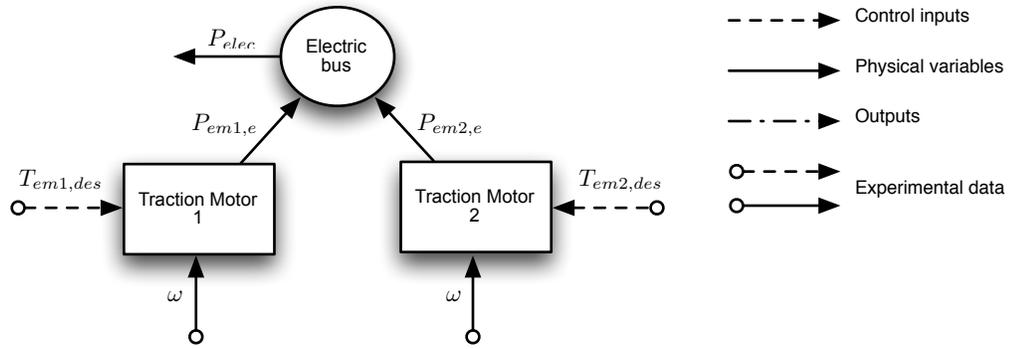


Figure 2.29: Validation scheme of electric machine model

we assume that the traction motor receives a torque command equal to the experimental torque, and the compare the electric power request obtained by simulation with the value measured. This is assumed to be the sum of the generator power and capacitor power, shared equally between the two traction machines. The quality of the match is an indication of the accuracy of the traction motor model (since the electric bus is assumed to be lossless and the power electronics efficiency is accounted for in the motor efficiency map).

#### 2.5.4 Engine, generator and accessories

Due to the lack of information at the engine shaft, the engine model is validated together with all the components directly connected to it, i.e. generator, hydraulic accessories, and mechanical accessories, as shown in Figure 2.31. The engine torque is calculated given the power request at the generator and the hydraulic load, assuming that the measured engine speed is the reference speed to be followed (the actual reference indicated by the controller is not an available measurement).

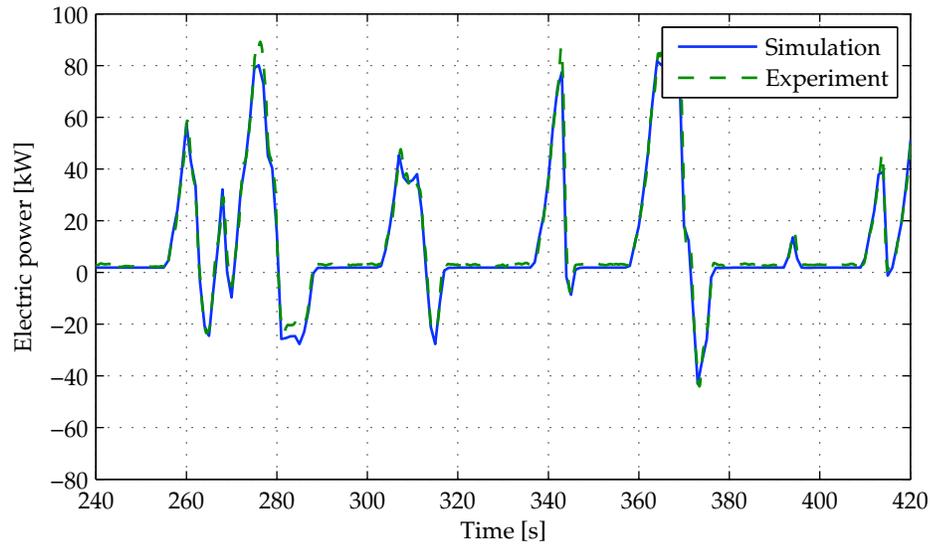


Figure 2.30: Comparison between calculated and measured electric power at one traction motor (detail of cycle *Route 2*). This curve represents the good quality of the efficiency map of the machine, since the electric power is computed directly using (2.50), based on torque and speed measurements.

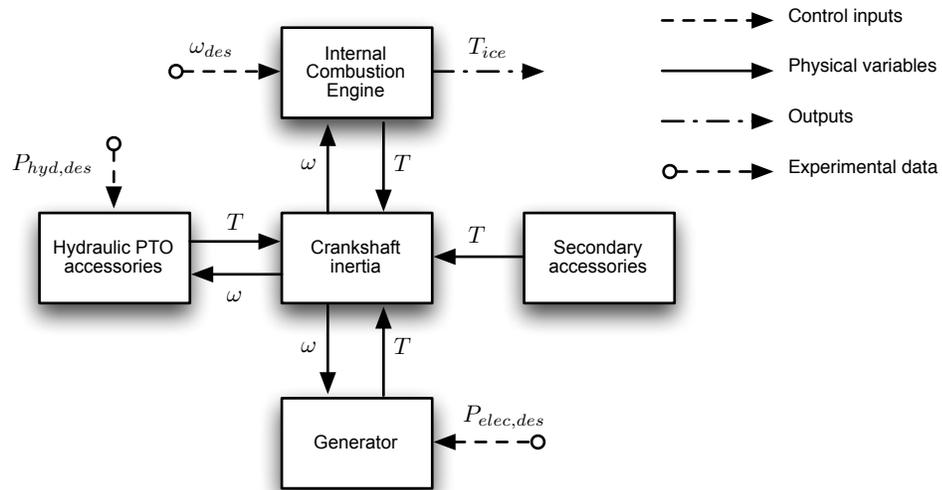


Figure 2.31: Validation scheme of engine model

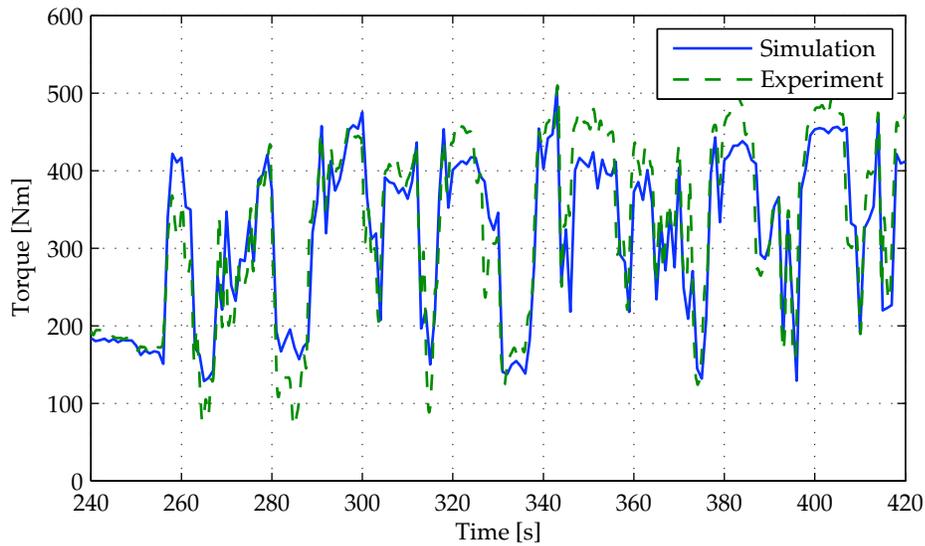


Figure 2.32: Comparison between calculated and experimental engine torque (detail of *Route 2*). This torque is the result of modeling the PTO accessories, the secondary accessories, and the generator. The models need to be validated together because there is no measurement of the torque request for each component. Considering this limitation, the agreement between experiment and simulation is acceptable.

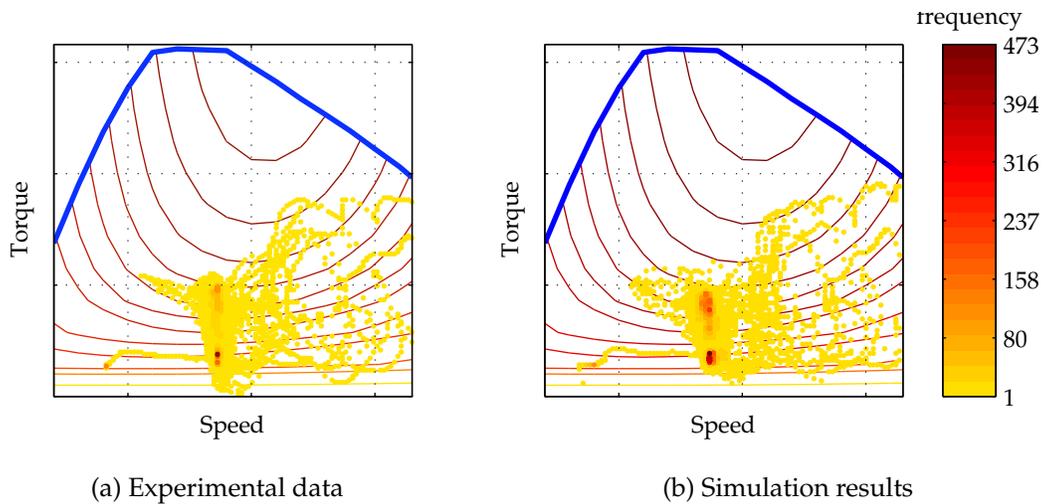


Figure 2.33: Comparison of engine operating points on efficiency map (cycle *Route 2*)

## 2.6 Conclusion

A behavioral model of the powertrain has been presented in this section. The model is designed with the intent to allow accurate energy analysis of the powertrain and fuel consumption, and to provide acceptable results in terms of performance estimation, while maintaining a fast execution time. Validation results collected using one of the case studies were also presented, to demonstrate the achievement of the design objectives. The modularity of the proposed implementation has been conceived to allow substantial improvements of the component models if needed (for example for drivability considerations), while maintaining the simulator structure.

## Chapter 3

### OPTIMAL CONTROL STRATEGIES FOR HYBRID VEHICLES

#### 3.1 Energy management of hybrid electric vehicles

Energy management in hybrid vehicles consists in deciding the amount of power delivered at each instant by the energy sources present in the vehicle. It is sometimes called *supervisory controller*, in opposition to *low-level* or *component-level* control strategies, which are used to drive single components so that they behave as dictated by the driver (or the supervisory controller). In this work, however, the terms *energy management* and *supervisory controller* are not synonyms. In fact, the supervisory controller is a layer above the energy management: while the latter is used to split the power demand between the powertrain actuators, the supervisory controller has the task to decide when such power split algorithm can be applied, and when, instead, a special behavior should be forced due to specific, possibly critical, situations<sup>1</sup>.

In a conventional (non-hybrid) vehicle, there is no need for an energy management strategy: the driver decides the instant speed and the instant power delivery using the brake and accelerator pedals, and, in manual transmission vehicles, decides what gear is engaged at each time. The driver's desires are translated into action by low-level controllers: for example, the engine control unit (ECU) determines the amount of fuel to be injected given the accelerator input; the automatic

<sup>1</sup>For example, if the driver steps suddenly on the brake pedal due to an emergency, there is no point in optimizing the brake repartition between the electric machine and the mechanical brakes for fuel consumption: the objective is decelerating as fast as possible, while the energy management aspect of the braking maneuver is secondary. Other cases in which the energy management strategy is superseded may include limp-home modes after a failure is diagnosed, towing, etc.

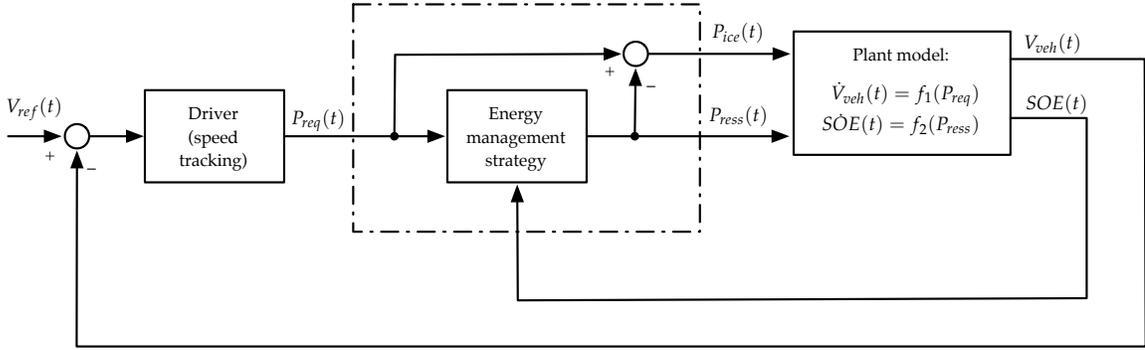


Figure 3.1: The role of energy management control in a hybrid electric vehicle

transmission controller decides when to shift gear based on engine conditions and vehicle speed, etc. In a hybrid vehicle, there is an additional decision that must be taken: how much power is delivered by each of the energy sources present on the vehicle. In principle, this could be delegated to the driver (for example, providing two separate accelerator commands); but it is much easier, simpler, and efficient if a computer takes care of it, leaving to the driver only the decision on how much total power is needed. This is why all hybrid vehicles include an energy management controller, which can be seen as an additional layer between the driver and the component controllers. As mentioned, the scope of a energy management control strategy in a hybrid electric vehicle is to determine the optimal power split between the energy sources present on board. The decision on what to consider optimal depends on the specific application: in most cases, the strategies tend to minimize the fuel consumption, but the objective could also include the minimization of pollutant emissions, the maximization of power delivery, or – most often – a compromise among all these goals.

The role of the energy management in a hybrid electric vehicle can be represented as in Figure 3.1. The outer layer in the figure is the speed control, which is the human driver in a real vehicle and a driver model (typically a PI controller) in simulation. The speed controller decides the total power request  $P_{req}(t)$  that the powertrain must deliver in order to follow the prescribed velocity profile. The

inner layer is the energy management strategy, which decides how to split the total power request between the energy sources present on-board: the rechargeable energy storage system (RESS) and the internal combustion engine (ICE) (the sum node is only conceptual: the total power is an algebraic function of both RESS and ICE power, but it is always smaller than their sum because of the power paths in the powertrain). The separation of the two control layers allows to consider only the battery state of energy dynamics in the energy management strategy, while the vehicle speed does not need to be treated as a state of the system, since it is controlled independently by the driver.

### 3.1.1 Definition of the optimal control problem for hybrid vehicles

In general, the energy management in a hybrid vehicle is an implementation of *optimal control*. This is the branch of control theory that deals with the problem of finding a control law for a given system such that a certain optimality criterion, usually defined as an integral performance index over a certain time frame, is achieved. In other words, it is a set of mathematical techniques for calculating a sequence of control actions, such that their integral effect is closest to a desired value. In the case of hybrid electric vehicles, the sequence of control actions taken by the energy management controller is represented by the instantaneous power repartition between energy sources; the integral effect is the fuel consumption over an entire trip or driving cycle, or the total pollution emission, or any other meaningful *cost function*, whose minimization is the optimization objective.

The traditional optimal control techniques can be used only with simple mathematical models of the system, and assume a perfect knowledge of the entire optimization horizon (time frame over which the optimization is defined); since both these conditions are usually not respected by real systems, optimal control implementation in a physical dynamic system, whose future is unknown, is necessarily *sub-optimal*, i.e., imperfect.

In more formal terms [33, 34], consider a generic dynamic system with state equation

$$\dot{x} = f(x, u, t), \quad (3.1)$$

where  $x \in \mathbb{R}^n$  indicates the vector of the state variables,  $u \in \mathbb{R}^m$  is the vector of the control inputs, and  $t$  denotes the time. The optimal control problem in the time interval  $t \in [t_0, t_f]$  corresponds to the choice of the law  $u(t) : [t_0, t_f] \mapsto \mathbb{R}^m$  that leads to the minimization of the *cost function* (or *performance index*)

$$J = \phi(x(t_f), t_f) + \int_{t_0}^{t_f} L(x(t), u(t), t) dt \quad (3.2)$$

under the terminal conditions (boundary conditions on the state)

$$\psi(t_f, x(t_f)) = 0 \quad (3.3)$$

and the local constraints

$$\begin{cases} G(x(t), t) \leq 0 \\ u(t) \in \mathcal{U}(t) \end{cases} \quad \forall t \in [t_0, t_f] \quad (3.4)$$

where  $\mathcal{U}(t)$  indicates the set of admissible control values at time  $t$ .

$L(x(t), u(t), t) \in \mathbb{R}$  is the instantaneous cost function, and  $\phi(x(t_f), t_f) \in \mathbb{R}$  represents the terminal cost (function of the system state at the final time). In the case of a hybrid electric vehicle, if fuel consumption minimization is the only objective of the controller, then  $L$  is the instantaneous fuel consumption, i.e. the mass flow rate of fuel into the engine. On the other hand, if pollutant emissions are also a concern, then  $L$  might be a weighted average of fuel consumption and emission rates. In general, the cost function can be any scalar index. The terminal cost  $\phi(x(t_f), t_f)$  can be used, for example, to account for the difference between the initial and final value of the state of charge.

The optimization problem is also subject to several constraints. Some of them are integral in nature (for example, the fact that the state of charge at the end of the trip must not be too different from the initial value); some are local (instantaneous power limits, state of charge boundaries).

The integral (or terminal) constraints can be enforced either as *hard* or *soft* constraints [71]. Hard constraints consist in the boundary conditions (3.3) on the dynamic equations that constitute the optimization problem; soft constraints, on the

other hand, modify the cost function (3.2) with the term  $\phi(x(t_f), t_f)$  in order to induce the final value of the constrained variable to be close, but not necessarily identical, to the desired target.

The local constraints (3.4) are instantaneous conditions that must be satisfied at each instant of time. The notation  $G(x(t), t) \leq 0$  is generic; in most cases, it represents the inequalities  $x_{min}(t) \leq x(t) \leq x_{max}(t)$ .

In the case of hybrid electric vehicles, the global constraints are related to the state of energy (SOE) of the energy storage system. The concept of state of energy is extremely important in HEV strategies, and its formal definition will be introduced later. For now, suffice it to say that it is a measure of the amount of energy stored electrically, expressed as a fraction of the maximum storage available. More often, the measure of state of charge (SOC) is used instead: this is a different way to express the same concept, based on the amount of electrical charge rather than energy, and is commonly used in the field of electrochemical batteries. The constraints on the state of charge are so significant that they define the typology of a hybrid electric vehicle: a *charge-sustaining* hybrid vehicle is so called because its battery SOC at the beginning and the end of each trip is roughly the same, which means that the vehicle maintains its own electrical charge without need for external recharging. On the other hand, a *charge-depleting* (or *plug-in*) hybrid electric vehicle (sometimes called PHEV) can be recharged using an electric outlet, and therefore the SOC after a trip can be lower than it was at the beginning. In other words, in charge-sustaining HEVs ultimately all the energy consumed derives from fuel, while in charge-depleting HEVs part of it is not derived from fuel, but from the electric power grid. In both charge-sustaining and charge-depleting hybrid vehicles, the condition on the terminal SOC is often considered as a soft constraint, as some difference between the desired and actual SOC at the end of a cycle is perfectly acceptable in a real vehicle and does not affect its functionality.

The local (instantaneous) constraints are represented by the power limitations of each energy source, variable with the operating conditions, and by the fact that the state of charge must remain between a maximum and a minimum value (to

make the batteries work at high efficiency and preserve their cycle life). Other constraints could be related to drivability or comfort issues: for example, it is usually a good idea to limit the frequency of switching between operating modes.

The optimization problem defined by equations (3.11) – (3.4) can be addressed in various ways. The solution to this problem is generally impossible (for practical implementation) without strong simplifications or assumptions.

### 3.1.2 Classification of energy management strategies

Several families of energy management strategies have been proposed in literature, with different characteristics, and many possible implementations. In fact, energy management strategies for hybrid electric vehicles can be categorized focusing on some of their characteristics. Guzzella and Sciarretta [71] propose the following subdivision:

1. numerical optimization methods, in which the entire driving cycle is taken into consideration and the global optimization is found numerically; dynamic programming and numerical search methods belong to this category, as well as model predictive control and stochastic dynamic programming;
2. analytical optimization methods, which again consider the entire driving cycle, but use an analytical problem formulation to find the solution in closed, analytical form, or at least provide an analytical formulation that makes the numerical solution faster than the purely numerical methods (at cost of oversimplifying the problem in order to obtain a suitable description). Among these methods, Pontryagin's minimum principle and the Hamilton-Jacobi-Bellmann equation are the most significant;
3. equivalent consumption minimization strategies, consisting in the minimization, at each time step of the optimization horizon, of an appropriately defined instantaneous cost function. This leads (ideally) to the minimization of the global cost function, if the instantaneous cost function (similar to an instantaneous *equivalent* fuel consumption) is suitably defined;

4. heuristic strategies, which do not involve explicit minimization or optimization, but rely on a set of rules to decide the value of the control to apply at each time. Rule-based control and fuzzy logic approaches are part of this category.

An alternative classification can be made as follows:

1. global optimization methods (either analytical or numerical), considering the complete optimization problem as a whole. At each instant of the optimization horizon, information about past, present and future driving conditions is assumed to be completely known. This category includes dynamic programming and Pontryagin's minimum principle;
2. local optimization methods, which reduce the global problem to a sequence of local problems (but not necessarily instantaneous). In this case, information about past and present conditions is used, and prediction of future driving conditions may be used, for example in a receding-horizon approach. In this sense, model predictive control, stochastic dynamic programming, and ECMS all belong to this category;
3. heuristic strategies, with no explicit optimization, and use of past and present information only.

The most important strategies described in literature are presented individually in the following sections, pointing out the assumptions on which they are based and the applications for which they are suitable.

### 3.2 Pontryagin's minimum principle

Pontryagin's minimum principle<sup>2</sup> [32, 33, 34] is a mathematical theorem that gives a set of instantaneous necessary optimality conditions. This means that the optimal solution to the global problem must satisfy these conditions (which are not, by themselves, a guarantee of optimality).

<sup>2</sup>The minimum principle was originally proposed (as a *maximum* principle) by Russian mathematician Lev Semenovich Pontryagin and his students in 1958 and later described in a textbook [32]. Some regard this theorem as the beginning of modern optimal control theory [31]

There are several formulations of the principle, depending on the way the optimal control problem is specified [34]. The two of most relevance for the HEV energy management problem are reported in the following sections.

### 3.2.1 Minimum principle for problems with no constraints on the state

In this kind of problem, the state variable can assume any value in its general domain  $x(t) \in \mathbb{R}^n$ . Given the system equation (3.1), the cost function (3.2), and the terminal constraints (3.3), the minimum principle states that the optimal control law  $u^*(t)$  must satisfy the following necessary conditions [34, §§2.1, 2.3]:

1.  $u^*(t)$  minimizes at each instant of time the Hamiltonian of the system

$$H(t, u(t), x(t), \lambda(t)) = \lambda^T(t) \cdot f(t, u(t), x(t)) + L(t, u(t), x(t)) \quad (3.5)$$

i.e., the optimal solution  $u^*(t)$  is such that

$$u^*(t) = \arg \min_{u \in \mathcal{U}} (H(x, u, \lambda, t)) \quad (3.6)$$

where  $\mathcal{U}$  is the set of admissible solutions.  $\lambda(t) \in \mathbb{R}^n$  (same dimension as the state vector  $x$ ) is a vector of optimization variables, also known as *adjoint states* or *co-states*;

2. the co-state must satisfy the dynamic equation

$$\dot{\lambda} = - \left. \frac{\partial H}{\partial x} \right|_{u^*, x^*} \quad (3.7)$$

3. the state variable  $x^*(t)$  must satisfy the terminal constraints

$$\psi(x(t_f), t_f) = 0; \quad (3.8)$$

however, if no terminal conditions are imposed on the state, then the terminal condition is given on co-state, which must be such that

$$\lambda^*(t_f) = \left. \frac{\partial \phi(x(t_f), t_f)}{\partial x} \right|_{*, t_f} \quad (3.9)$$

where  $\phi(x(t_f), t_f)$  is the terminal cost appearing in (3.2).

### 3.2.2 Minimum principle for problems with constraints on the state

In this case, the state variables are constrained to remain within some boundaries, which can be time-varying:  $x(t) \in \Omega_x(t) \subset \mathbb{R}^n \forall t \in [t_0, t_f]$ . The state boundaries can be expressed by defining the set of admissible states as those for which the conditions  $G(x, t) \leq 0$  are satisfied, i.e.:

$$\Omega_x(t) = \{x \in \mathbb{R}^n | G(x, t) \leq 0\}, \quad (3.10)$$

where the function  $G(x, t) : \mathbb{R}^n \mapsto \mathbb{R}^p$  represents a set of  $p$  inequalities that the components of the state vector must satisfy.

The formulation of the minimum principle itself is the same as introduced in Section 3.2.1, but the state boundaries can be taken into account modifying the cost function (3.2) with an additional term that introduce extra cost whenever the state boundaries are reached [36]:

$$J = \phi(x(t_f), t_f) + \int_{t_0}^{t_f} L(x(t), u(t), t) + \mu \Gamma(x(t), t) dt \quad (3.11)$$

where

$$\Gamma(x(t), t) = \begin{cases} 0 & \text{if } G(x(t), t) < 0 \text{ (constraints not active)} \\ 1 & \text{if } G(x(t), t) \geq 0 \text{ (constraints active)} \end{cases} \quad (3.12)$$

If  $G(x, t)$  is a set of several constraint functions,  $\Gamma(x, t)$  has the same number of components as  $G(x, t)$ ; each component is defined on the basis of the corresponding component of  $G(x, t)$ .  $\mu \in \mathbb{R}^p$  is a vector of constants with the same dimensions as  $G(x, t)$  and  $\Gamma(x, t)$ . The value of the constants  $\mu$  is arbitrary, but should be high enough as to guarantee that the additional cost due to hitting (or overcoming) a state constraint makes the corresponding solution unacceptable. This definition of the cost function introduces discontinuities in the Hamiltonian function at the time instants in which the state boundaries are reached, and this translates into discontinuities in the value of the co-state  $\lambda(t)$  at those instants, as can be inferred from (3.7). Therefore, at the instant  $\bar{t}$  in which the state  $x$  reaches a boundary of the domain  $\Omega_x$ , the co-state variable “jumps” by an arbitrary quantity [34, §2.5]:

$$\lambda(\bar{t}^+) = \lambda(\bar{t}^-) + \mu_0 \quad (3.13)$$

### 3.2.3 Notes on the minimum principle

Pontryagin's minimum principle is a rather powerful theorem. It allows to re-define the global optimal control problem in terms of local conditions expressed by the differential equations (3.1), (3.7) and by the instantaneous minimization (3.6). Clearly, the global nature of the problem does not disappear, and remains evident in the fact that the boundary conditions are given at the initial and final time, therefore the problem cannot really be solved as a standard dynamic evolution problem.

The minimum principle only gives necessary (not sufficient) optimality conditions, but these can be used to find optimal control candidates. The solutions satisfying all the conditions of the minimum principle are called *extremal controls*; Pontryagin's principle ensures that the optimal control, if it exists, must be an extremal control. If the optimal control problem admits one solution, and there is only one extremal control, then that is the optimal control. Even if several extremal controls are found, it may be relatively easy to simply apply all of them one at a time and then identify the optimal control as the extremal giving the lowest total cost.

The analytical nature of the theorem means that the equations must be derived using a system model expressed in simple form. However, if the simplification is excessive, the model may not be representative of the real system and the resulting optimal control may actually be sub-optimal. Another limit of this technique is that, in general, applying Pontryagin's minimum principle requires a-priori knowledge of the entire optimization horizon, which confines real-time implementation to cases in which the load on the vehicle is either known or predictable, or to some special cases in which the future conditions do not need to be considered because they disappear after analytical manipulation of the equations (an example of this last case is shown in [72]). These issues will be discussed in more detail in Chapter 4.

### 3.3 Dynamic programming

#### 3.3.1 General concepts

Dynamic programming [33] is a numerical method for solving multistage decision-making problems. It is the only optimal control technique capable of providing the optimal solution to problems of any complexity level (in the limits of computational capabilities); unfortunately, it is non-causal (since it requires the entire driving cycle to be known in advance) and therefore is only implementable in simulation environment. Dynamic programming is based on Bellman's principle of optimality, which can be put in words as follows:

if the optimal solution for a problem passes through an intermediate state  $(x_1, t_1)$ , then the optimal solution to the same problem starting at  $(x_1, t_1)$  must be the continuation of the same path [6].

or

An optimal control policy has the property that no matter what the previous decision (i.e., controls) have been, the remaining decisions must constitute an optimal policy with regard to the state resulting from those previous decisions [73].

This means that the optimal path from any of its intermediate steps to the end corresponds to the terminal part of the entire optimal solution. To describe this principle in formulas, consider a discrete-time system

$$x_{k+1} = f_k(x_k, u_k) \quad (3.14)$$

with  $k = 0, 1, \dots, N - 1$  and  $u_k \in \mathcal{U}(x_k)$  (control constraints at step  $k$ ). The control policy is

$$\pi = \{u_0, u_1, \dots, u_{N-1}\}, \quad (3.15)$$

where the controls  $u_k$  are such that  $u_k \in \mathcal{U}_k(x_k)$  for all  $x_k$ . The cost of  $\pi$  starting at  $x_0$  is

$$J_\pi(x_0) = L_N(x_N) + \sum_{k=0}^{N-1} L_k(x_k, u_k) \quad (3.16)$$

where  $L_k$  is the instantaneous cost function (the same as the integrand in the continuous-time formulation (3.11)); the optimal cost function is the one that minimizes the total cost

$$J^*(x_0) = \min_{\pi} J_{\pi}(x_0) \quad (3.17)$$

and the optimal policy  $\pi^* = \{u_0^*, u_1^*, \dots, u_{N-1}^*\}$  is such that

$$J_{\pi^*}(x_0) = J^*(x_0). \quad (3.18)$$

Consider now the “tail subproblem” of minimizing the *cost-to-go* from time  $i$  (and state  $x_i$ ) to time  $N$ :

$$V_i = L_N(x_N) + \sum_{k=i}^{N-1} L_k(x_k, u_k) \quad (3.19)$$

and the “tail policy”  $\{u_i^*, u_{i+1}^*, \dots, u_{N-1}^*\}$ , i.e. the last part of the optimal policy  $\pi^*$ . Bellman’s principle of optimality states that *the tail policy is optimal for the tail subproblem*. This statement, equivalent to the one given at the beginning of this section, has an analytical proof [33] in which the induction principle is used to show that  $J_k(x_k) = J_k^*(x_k)$ , where  $J_k^*(x_k)$  is the optimal cost of the tail subproblem that starts at time  $k$  and state  $x_k$ .

The dynamic programming algorithm is based on Bellman’s principle of optimality. Starting from the final step  $N$ , the algorithm proceeds backwards using the sequence of policies

$$\mu_k^* = \arg \min_{u_k \in \mathcal{U}_k} (L_k(x_k, u_k) + J_{k+1}(f_k(x_k, u_k))) \quad k = N-1, N-2, \dots, 1, 0 \quad (3.20)$$

$J_0(x_0)$ , generated at the last step, is equal to the optimal (minimum) cost  $J^*(x_0)$ .

In other words, it is possible to determine the optimal sequence of control actions proceeding backwards from the final state, choosing at each step the path that minimizes the cost-to-go (integral cost from that time step until the final state).

A classical example is shown in Figure 3.2: the problem is to go from point  $A$  to point  $B$  in the shortest time, assuming that the travel time for each leg is represented by the number next to it. Given the size of the problem, it would be possible

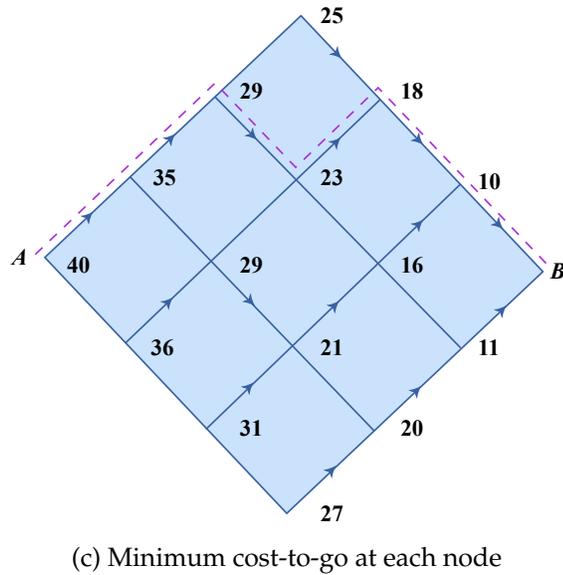
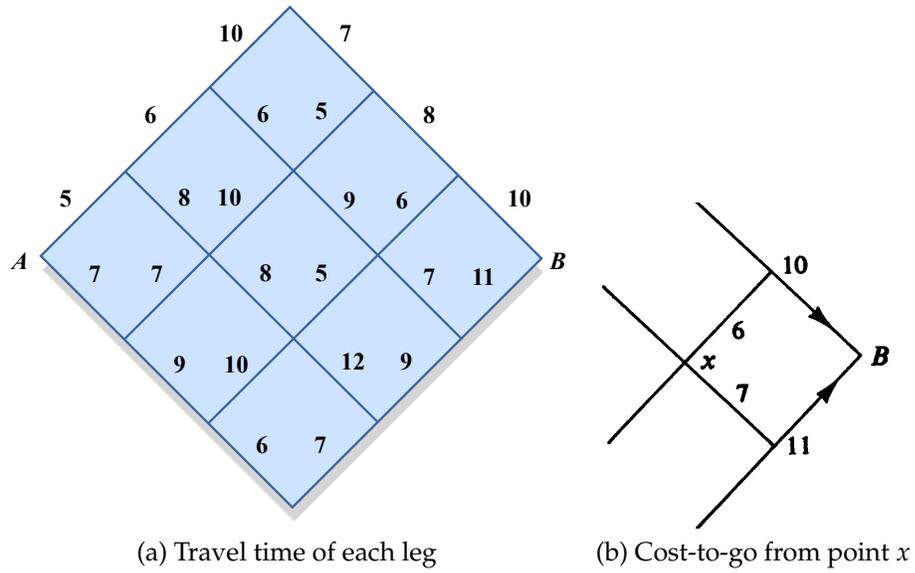


Figure 3.2: Shortest path problem [5, 6]

(but very tedious) to consider all possibilities and then pick the one corresponding to the minimum travel time. Using the principle of optimality, instead, we can reduce the number of evaluations needed, by working backwards from B: if the optimal path passes through  $x$ , then the fastest way to arrive at B is to go up and then down ( $6+10 = 16$ ), rather than down and up ( $7+11 = 18$ ). This means that the cost-to-go at point  $x$  is 16 and the corresponding optimal path is already determined (passes through the upper node). Repeating similar considerations for all other nodes, until the origin  $A$  is reached, gives the minimum cost-to-go at each node (see Figure 3.2.c); the optimal path is the one that corresponds to this minimum cost.

### 3.3.2 Application to HEVs

Dynamic programming can be implemented as supervisory controller in a hybrid vehicle, as first shown in [28]. The sequence of choices represents the power split between the internal combustion engine (ICE) and the rechargeable energy storage system (RESS, i.e. battery pack, supercapacitors, or other energy accumulation device) at successive time steps. The cost corresponds to fuel consumption, energy consumption, emissions, or any other design objective. The set of choices at each instant is determined by considering the state of each powertrain component and the total power requested by the driver. Given the current vehicle speed and the driver's demand (accelerator position), the controller determines the total power that should be delivered to the wheels. Then, using maps of the components and feedback on their present state, it also determines the maximum and minimum power that each energy source can deliver. If the power demand equals or exceeds the total available power from both sources, there is no choice to be made: each of them should be used at the maximum of its capabilities. Otherwise, there are infinite combinations such that the sum of the power from ICE and RESS equals the power demand. In most algorithms, including dynamic programming, instead of considering this continuum of solutions, a discrete number of them is selected and evaluated. The number of solution candidates that can be considered is a compromise between the computational capabilities and the accuracy of the

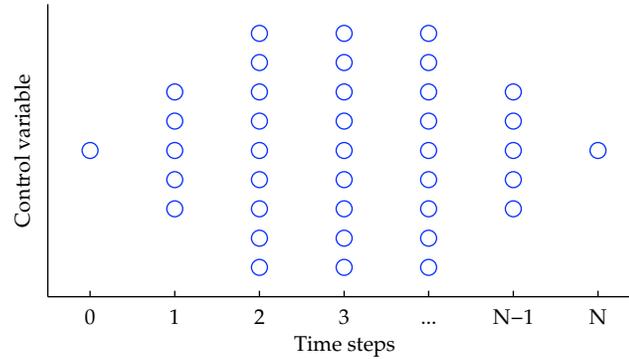
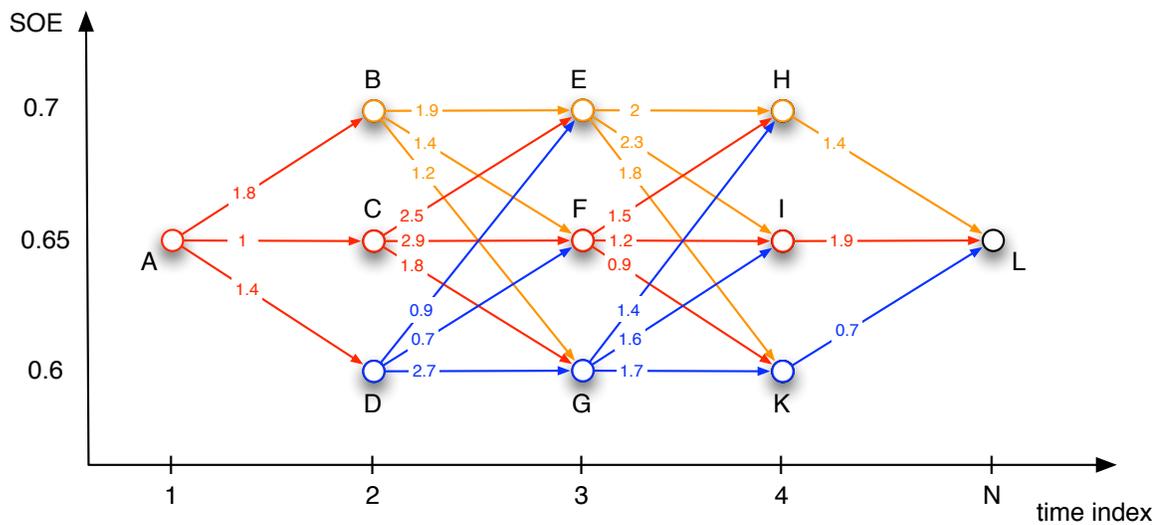


Figure 3.3: Dynamic programming in HEVs: sequence of feasible power splits

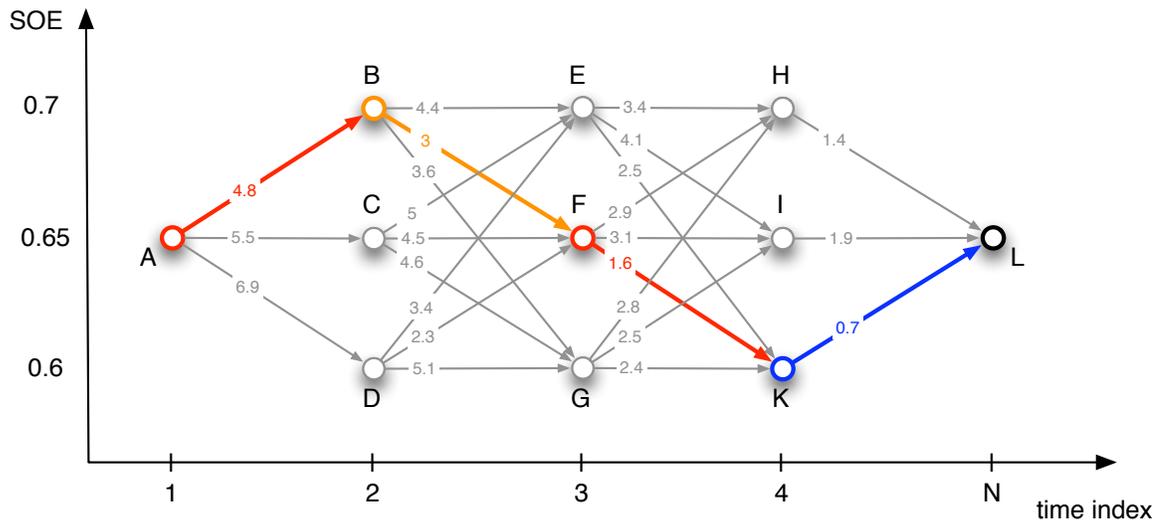
result: in fact, the minimum cost may not exactly coincide with one of the selected points, but the closer these are to each other, the better the approximation of the optimal solution.

Once the grid of possible power splits, or solution candidates, is created (see Figure 3.3), the procedure outlined earlier can be used, associating a cost to each of the solution candidates. Proceeding backwards (i.e., from the end of the driving cycle), the optimal cost-to-go is calculated for each grid point, and stored in a matrix of costs. When the entire cycle has been examined, the path with the lowest total cost represents the optimal solution.

The procedure can be explained with the example shown in Figure 3.4, which refers to a generic HEV configuration with a single degree of freedom. The decision variable is the battery state of energy SOE, which can take a finite number of values (in the example, just three: 0.6, 0.65 or 0.7). The objective of the dynamic programming algorithm is to select the optimal sequence of SOE such that the total cost is minimized. Selecting a sequence of SOE is equivalent to deciding a sequence of values of battery power, because the variation of SOE between time steps is proportional to the integral of the battery power between those steps. The SOE (and not the power) is chosen as the decision variable because this allows to satisfy the constraints on the maximum and minimum state of energy very easily, since only the admissible values are considered; also, the initial and final values



(a) Arc costs



(b) Cost-to-go and optimal path

Figure 3.4: Example of application of dynamic programming to a hybrid electric vehicle

of SOE are set with no effort. The constraints on the battery power are expressed in terms of maximum and minimum variation of SOE between two subsequent time steps. The first step for applying the algorithm is to calculate all the arc costs. These are the costs of moving from all admissible nodes at time  $k$  to all the admissible nodes at time  $k + 1$ . Figure 3.4.a shows all the admissible arc costs in this case: for example, at time  $k = N - 1 = 4$ , all three values of SOE are admissible (nodes H, I, K), but only one is accepted at the final time (node L); thus, three arc costs must be defined ( $H \rightarrow L$ ,  $I \rightarrow L$  and  $K \rightarrow L$ ). At time  $k = 3$ , instead, there are nine possible combinations (from any of the nodes E, F, G to any of the nodes H, I, K). Similar considerations can be made for all other time steps. Once all the arc costs have been determined, the cost-to-go can be calculated, starting from the final point and going backwards (Figure 3.4.b). At time  $k = 4$ , the cost-to-go of each node H, I, K corresponds to the arc cost because the following time instant is the end of the optimization horizon. At time  $k = 3$ , the cost-to-go of each node corresponds to the minimum cost associated with moving from that node to the end. So, for node E, the cost-to-go is the one corresponding to path with minimum cost among the possible alternatives:  $E \rightarrow H \rightarrow L$ ,  $E \rightarrow I \rightarrow L$ , and  $E \rightarrow K \rightarrow L$ . The respective costs are (from Figure 3.4.a):  $2+1.4=3.4$ ,  $2.3+1.9=4.2$ , and  $1.8+0.7=2.5$ ; these values are shown in Figure 3.4.b in correspondence of the respective path. Thus, the best path from E to L passes through K and has a cost of 2.5; the best path from F to L passes through K and has a cost of 1.6, and the best path from G to L passes through H and has a cost of 1.4. This is all the information needed before the algorithm moves to the preceding time step ( $k = 2$ ), and computes the arc costs for points B, C, and D. Because of Bellman's optimality principle, the optimal path from E, F, or G to L is not affected by the choice at the previous time step, therefore the cost-to-go from B to L is given by the sum of the arc cost from B to either E, F, G and of the optimal cost from there to L: for example, going from B to L passing through E costs 1.9 (cost of  $B \rightarrow E$ ) plus 2.5 (lowest cost of  $E \rightarrow L$ ). With similar reasoning, the entire graph of Figure 3.4.b is completed with the arc costs, and at that point it is possible to choose the optimal path as the one that gives the lowest cost from A to L. This is 4.9 and is obtained passing through B, F and K.

Dynamic programming offers the mathematically optimal solution, within the accuracy limits due to the discretization of the candidate solutions. However, it is not applicable in a real vehicle, for two important reasons:

1. the solution has to be calculated backwards, and therefore the entire driving cycle must be known a priori, and
2. it is a procedure computationally heavy, requiring the backward solution of the entire problem before being able to determine the first control action.

Despite these important shortcomings, dynamic programming provides the closest approximation to the optimal solution, and is often used to determine the maximum potentiality of a given architecture, thus serving as a benchmark for other control strategies [28, 45, 74, 46, 72, 29, 44].

## **3.4 Stochastic Dynamic Programming**

### **3.4.1 Generalities**

As we have seen in the previous section, dynamic programming can be used to obtain the optimal solution to the energy management problem, but has some noticeable drawbacks. First, the optimization is performed with respect to a specific driving cycle and might be neither optimal nor charge-sustaining under other cycles; secondly, the feedback solution to the dynamic programming is not directly implementable. It can be used to extract heuristic control rules, but the rule extraction process can be time-consuming. One method to implement dynamic programming would be to use a forecasting technique to determine with reasonable accuracy the future load profile, then solve the dynamic programming algorithm for the predicted profile. If the prediction is good enough and is updated at each time step, and only the control action corresponding to the first instant of the prediction horizon is used (receding horizon approach), then the solution can be fairly close to the optimal one. However, the method just described presents two important issues that prevent its actual feasibility: the inherent difficulty in obtaining a good prediction of future states for a relatively long horizon (thus reducing the

optimality of the solution), and the extremely high computational cost of solving a complete dynamic programming problem at each time step.

To overcome these drawbacks, a design procedure based on stochastic dynamic is discussed. Instead of being optimized over a given driving cycle, the power management strategy is optimized over a family of random driving cycles in an average sense. In order to obtain a time-invariant control strategy, an infinite-horizon optimization problem is formulated and solved by using *stochastic dynamic programming* (SDP), which is an evolution of the deterministic, discrete-time dynamic programming algorithm.

Stochastic dynamic programming has been successfully applied as a method for obtaining a quasi-optimal policy that is implementable on-line and in real time and can obtain results close to the optimal solution. The online implementation is possible because the method is used to generate a static mapping of system states into control actions, so that only a look-up table and interpolation is needed. The control map is obtained using statistical properties of the expected load profiles (driving cycles), and is optimal in the average sense (i.e., for a given combination of states, the control that derives from the SDP map is optimal for the most likely load sequence). The power demand is estimated under the assumption that the sequence of values it assumes can be modeled using a Markov chain, as described in the following section.

### **3.4.2 Markov chains**

A Markov system (or Markov process or Markov chain) is a system that can be in one of several (numbered) states, and can pass from one state to another each time step according to fixed probabilities. Having the Markov property means that, given the present state, future states are independent of the past states. In other words, the description of the present state fully captures all the information that could influence the future evolution of the process. Future states will be reached through a probabilistic process instead of a deterministic one.

The property of the Markov chain of the future to be conditionally independent of the past, can be formally stated as:

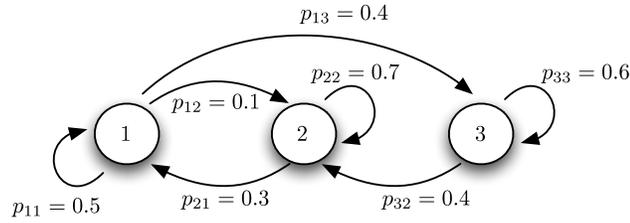


Figure 3.5: Simple example of a Markov chain

$$p(x_n = a_{i_n} \mid x_{n-1} = a_{i_{n-1}}, \dots, x_1 = a_{i_1}) = p(x_n = a_{i_n} \mid x_{n-1} = a_{i_{n-1}}) \quad (3.21)$$

In a Markov process, the probability that the process jumps from a state  $i$  to a state  $j$  does not depend on the states visited by the system before coming to  $j$ . This is called the memoryless property, and distinguishes a Markov process from other stochastic processes.

Because of the memoryless property, one can associate a probability with a transition from a state  $i$  to a state  $j$ , denoted as  $p(i, j)$ .

As an example, consider a Markov chain with three states, numbered 1, 2, and 3, as represented in Figure 3.5. Each arrow indicates a possible transition, and the number on the arrow denotes the probability of that transition. Thus, the probability of transitioning from 2 to 1 is  $p(2, 1) = 0.3$  and the probability of transitioning from 2 to 2 is  $p(2, 2) = 0.7$ . Notice that some transitions (for example, from 2 to 3 or from 3 to 1) are not represented, which means that they are not possible.

When the system comes to a state  $i$ , where it jumps to depends on the transition probabilities:  $p(i, 1)$ ,  $p(i, 2)$  and  $p(i, 3)$  — quantities which are not affected by the states visited before coming to  $i$ . Thus, when it comes to jumping to a new state, the process does not “remember” what states it has had to go through in the past. The next state to which it jumps depends only on the current state (say  $i$ ) and the probabilities of jumping from that state to other states—  $p(i, 1)$ ,  $p(i, 2)$  and  $p(i, 3)$ .

In general, when the system is ready to leave state  $i$ , to which state it jumps depends only on  $p(i, j)$  if  $j$  is the state to which it jumps. Furthermore,  $p(i, j)$  is completely independent of where the system has been before coming to  $i$ .

All the transition probabilities of a Markov chain can be conveniently stored in the form of a matrix, called the *one-step transition probability matrix* or simply *transition probability matrix*  $\mathbf{p}$ , with dimensions  $(m \times m)$  if  $m$  is the number of states in the Markov chain.

The transition probability matrix in our 3-state example is:

$$\mathbf{p} = \begin{bmatrix} 0.5 & 0.1 & 0.4 \\ 0.3 & 0.7 & 0 \\ 0 & 0.4 & 0.6 \end{bmatrix}. \quad (3.22)$$

$p(i, j)$  denotes the element  $(i, j)$  ( $i$ th row and the  $j$ th column) of the matrix  $\mathbf{p}$  and represents the one-step transition probability of jumping from state  $i$  to state  $j$ . The zero probability indicates that there is no transition possible between the corresponding states and is equivalent to a missing arrow in the transition diagram. The entries in each row of the transition probability matrix add up to 1.

The outputs, or *emissions*, of a Markov chain can be different from its states. They are defined by the set  $\{s_1, \dots, s_n\}$  and calculated using the emission matrix  $\mathbf{b}$ , an  $(m \times n)$  matrix whose  $(i, k)$  entry gives the probability of emitting the output  $s_k$  given that the model is in state  $x_i$ . If the outputs of the Markov chain correspond to its states, the emission matrix is the identity matrix of size  $m$ .

A sequence of 10 values generated according to the Markov chain described by

(3.22) and assuming  $\mathbf{b} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$  is, for example<sup>3</sup>:

$$x = \{1; 3; 2; 2; 2; 2; 2; 1; 1; 2\}$$

or

$$x = \{1; 1; 3; 2; 2; 2; 2; 2; 2; 2\}.$$

<sup>3</sup>The Matlab function `hmmgenerate` (in the Statistics toolbox) can be used to generate Markov chains

### 3.4.3 Stochastic dynamic programming for HEV energy management

The procedure for generating and implementing a stochastic dynamic programming solution in a hybrid vehicle can be outlined as follows:

1. collect driving cycle data in the form of sequences of values of power demand and vehicle speed ;
2. discretize the state variables and the control variables into a finite number of values;
3. generate a stochastic model (Markov chain model) of these variables, i.e., identify a suitable probability distribution that is able to generate at random a sequence of values of speed and power request such that the characteristics of the original driving cycle data are preserved;
4. solve the stochastic dynamic problem for all possible combinations of state and control values, and store the results in a map, which is time-invariant and contains the values of the controls that are optimal in an average sense (i.e., considering the statistical distribution on which the solution is based);
5. implement the control on-line by using the map generated in the previous step.

A basic description of the stochastic dynamic programming approach can be found in [33], and its application to hybrid vehicle energy management has been the subject of several papers [26, 75, 76, 77, 78]. The formulation presented here is based on [33] and particularized to the case of HEVs following the examples in [77] and [26].

In all the discrete-time equations that follow,  $k$  is the time index, varying between 0 and  $N$ .

In order to implement the stochastic dynamic programming algorithm, the system is described using three states: the battery state of energy  $\zeta$ , the vehicle speed  $V_{veh}$  and the power demand  $P_{req}$ . The latter two states are usually not included when formulating the problem for deterministic optimal control or dynamic programming, because both speed and power demand are assumed to be known. In

this case, instead, they are treated as variables whose prediction is part of the control problem and thus they must be explicitly included in the problem formulation.

The system state equations are:

- the discrete-time version of the state of energy dynamics

$$\zeta_{k+1} = f(u_k, \zeta_k) \quad (3.23)$$

- the vehicle dynamics equation

$$v_{k+1} = v_k + a_k(u_k, P_{req,k}) \Delta t \quad (3.24)$$

where  $a_k(u_k, P_{req,k})$  is the vehicle acceleration generated applying the control  $u_k$  and considering the power request  $P_{req,k}$ , and  $\Delta t$  is the sampling time;

- the stochastic model of the power request

$$P_{req,k+1} = w_k(v_k, P_{req,k}). \quad (3.25)$$

The power demand the vehicle speed and the battery state of energy are quantized in a finite number of values:

$$P_{req} \in \{P_{req}^1, P_{req}^2, \dots, P_{req}^{N_p}\} \quad (3.26)$$

$$v \in \{v^1, v^2, \dots, v^{N_v}\} \quad (3.27)$$

$$\zeta \in \{\zeta^1, \zeta^2, \dots, \zeta^{N_z}\} \quad (3.28)$$

$N_p$ ,  $N_v$  and  $N_s$  are, respectively, the number of possible values for the power demand, the vehicle speed and the state of energy. A superscript is used to specify the index of each variable in the set of its discretized values, while a subscript index denotes the time instant at which the variable is considered. The state of the system is defined as

$$x = [\zeta, v, P_{req}]^T \quad (3.29)$$

### 3.4.4 Stochastic driving cycle models

In the context of stochastic dynamic programming, the power demand is not considered to be known a priori; instead, it is modeled as stationary Markov chain [77]:

$$P_{req,k+1} = w_k \quad (3.30)$$

where  $w_k$  is a random number generated according to the probability distribution

$$p_{il,j} = \Pr \left\{ w = P_{req}^j | P_{req} = P_{req}^i, v = v^{ml} \right\} \quad (3.31)$$

i.e. the probability that, at the given time instant  $k$ , the power demand assumes the value  $P_{req}^j$ , given that at time step  $k - 1$  the (measured) values of power demand and vehicle speed were, respectively,  $P_{req}^i$  and  $v^l$ .

The indices cover the entire discretization interval for all variables:  $i = 1, 2, \dots, N_p$ ;  $j = 1, 2, \dots, N_p$ ; and  $l = 1, 2, \dots, N_v$ .

Notice that  $p_{il,j}$  is actually a simplification of a two-state Markov chain  $p_{il,jm}$  (with  $m = 1, 2, \dots, N_v$ ), justified by the fact that the variation of velocity from one time step to the next is a deterministic process, since it can be predicted using the system state equation.

The transition probability  $p_{il,j}$  can be estimated from a set of measured data as

$$p_{il,j} = \frac{n_{il,j}}{n_{il}} \quad \text{if } n_{il} \neq 0 \quad (3.32)$$

where  $n_{il}$  is the number of times that the value of power demand  $P_{req}^i$  occur when the vehicle speed has value  $v^l$ , and  $n_{il,j}$  is the number of times that the transition from  $P_{req}^i$  to  $P_{req}^j$  occurs, given that the vehicle speed at the time that  $P_{req} = P_{req}^i$  was  $v^l$ . Clearly, it is necessary to have a very large data set in order to obtain significant values of probability distribution for all possible combinations of power and speed.

The sequence of random power demand generated by (3.31) can be used to create what is essentially a random driving cycle with statistical properties similar to the training data.

### 3.4.5 Problem formulation

The cost function that we aim to minimize is defined as the sum of the cost at all steps. Unlike deterministic dynamic programming, in which the length of the optimization horizon is usually well defined, in stochastic DP there is no actual driving cycle. For this reason, it makes sense to define an *infinite horizon* problem, which consists in finding an optimal control policy  $u = \pi^*(x)$  that minimizes the expected total cost (defined over an infinite time)

$$J_\pi(x_0) = \lim_{N \rightarrow \infty} E \left\{ \sum_{k=0}^{N-1} \gamma^k c(x_k, \pi(x_k)) \right\} \quad (3.33)$$

where  $E \{ \cdot \}$  indicates the expected value<sup>4</sup> of the quantity  $\{ \cdot \}$ ,  $k$  is an index representing the time step,  $c(x_k, \pi(x_k))$  is the cost at time step  $k$ , and  $\gamma$  ( $0 < \gamma < 1$ ) is a discount factor that is used to ensure that the infinite series converges to a finite cost.  $J_\pi(x_0)$  represents the expected total cost incurred if the system starts at state  $x_0$  and the control policy  $\pi$  is applied. The instantaneous cost  $c(x_k, \pi(x_k))$  can be the fuel consumption [26] or a weighted sum of fuel consumption and emissions [77], possibly with a term to penalize excessive deviation of the state of charge from a nominal value.

The method that is commonly used to solve the SDP problem (e.g. in [75, 77, 26]) is the *policy iteration* algorithm, composed by an iteration of two steps. In the first step, the cost  $J_\pi$  of a policy  $\pi$  for an infinite horizon is evaluated; in the second step, an improved policy is found. Then the first step is repeated to evaluate the improved policy. The sequence is repeated until the cost of the improved policy converges within a pre-defined tolerance, or, in other words, until the policy improvement step becomes ineffective (the cost of the improved policy is very close

<sup>4</sup>The expected value (or expectation) of a function of a single discrete variable is defined by  $E \{ f(x) \} = \sum_{x \in X} [f(x)P(x)]$ , where  $P(x)$  is the probability function and the sum is extended to the set  $X$  of discretized values of  $x$ .

to the cost of the previous policy). The entire procedure allows to determine the optimal control *given a value of the state*  $x^i$ : it represents the value of the control that minimizes the cost of a random cycle starting at  $x^i$ . Therefore, it must be repeated for all the set of quantized states  $x^i = \{x^1, x^2, \dots, x^{N_s N_p N_v}\}$ .

Policy evaluation step: this step is itself an iteration, which is necessary because the total cost of a policy is the sum of the instantaneous costs along an infinite horizon. Hence, the total cost is given by the last term of the iterative sequence:

$$\begin{aligned} J_{\pi}^1(x^i) &= c(x^i, \pi(x^i)) + J_{\pi}^0 \\ J_{\pi}^2(x^i) &= c(x^i, \pi(x^i)) + E \{ \gamma J_{\pi}^1(x') \} \\ &\vdots \end{aligned} \tag{3.34}$$

$$J_{\pi}^s(x^i) = c(x^i, \pi(x^i)) + E \{ \gamma J_{\pi}^{s-1}(x') \}$$

The sequence should be repeated until the estimated cost converges, i.e. until  $\|J_{\pi}^s(x^i) - J_{\pi}^{s-1}(x^i)\| < \epsilon$ ; however, thanks to the presence of the discount factor  $\gamma$ , the last terms in the sequence are much smaller than the first ones and therefore, for faster computation, it is possible to perform only a fixed number  $S$  of iterations, thus neglecting the lower-valued terms:  $s = S$ . The closer  $\gamma$  is to one, the more weight is attributed to long-term results (thus a larger  $S$  is needed).

Each time (3.34) is evaluated, the cost at a successive time step is calculated;  $x'$  represents the stochastic evolution of  $x^i$  under the control policy  $\pi$ :

$$x' = f(x^i, \pi(x^i), w) \tag{3.35}$$

where  $f(\cdot)$  is a three-component function defined by the state equations (3.23), (3.24) and (3.25).

Policy improvement step: after the cost of the policy  $\pi$  has been evaluated, the policy is improved using the following equation:

$$\pi' = \arg \min_{u \in \mathcal{U}(x_i)} \left( c(x^i, u) + E \{ \gamma J_{\pi}(x') \} \right) \tag{3.36}$$

## 3.5 Equivalent Consumption Minimization Strategy

### 3.5.1 Basic concepts

The *equivalent consumption minimization strategy* (usually referred to as *ECMS*) was introduced by Paganelli et al. [7, 40] as a method to reduce the global minimization problem defined in Section 3.1.1 to an instantaneous minimization problem to be solved at each instant, without use of information regarding the future. The description reported here is taken from [7].

This strategy is based on the concept that, in charge-sustaining vehicles, the difference between the initial and final state of charge of the battery is very small, negligible with respect to the total energy used. This means that the electrical energy storage is used only as an energy buffer. Since all the energy ultimately comes from fuel, the battery can be seen as an auxiliary, reversible fuel tank. The electricity used during a battery discharge phase must be replenished at a later phase using the fuel from the engine (either directly or indirectly through a regenerative path).

Two cases are possible at a given operating point:

1. the battery power is positive (discharge case): a recharge with the engine will require some additional fuel consumption in the future;
2. the battery power is negative (charge case): the stored electrical energy will be used to reduce the engine load, which implies a fuel saving.

In both cases, an equivalent fuel consumption can be associated with the use of electrical energy; the equivalent future fuel consumption can be summed to the present real fuel consumption to obtain the instantaneous equivalent fuel consumption

$$\dot{m}_{eqv} = \dot{m}_f + \dot{m}_{ress} = \dot{m}_f + \frac{s}{Q_{lhv}} P_{elec} \quad (3.37)$$

where  $\dot{m}_f$  is the engine instantaneous fuel consumption (fuel mass flow rate),  $Q_{lhv}$  is the fuel lower heating value (energy content per unit of mass),  $\dot{m}_{ress}$  is the

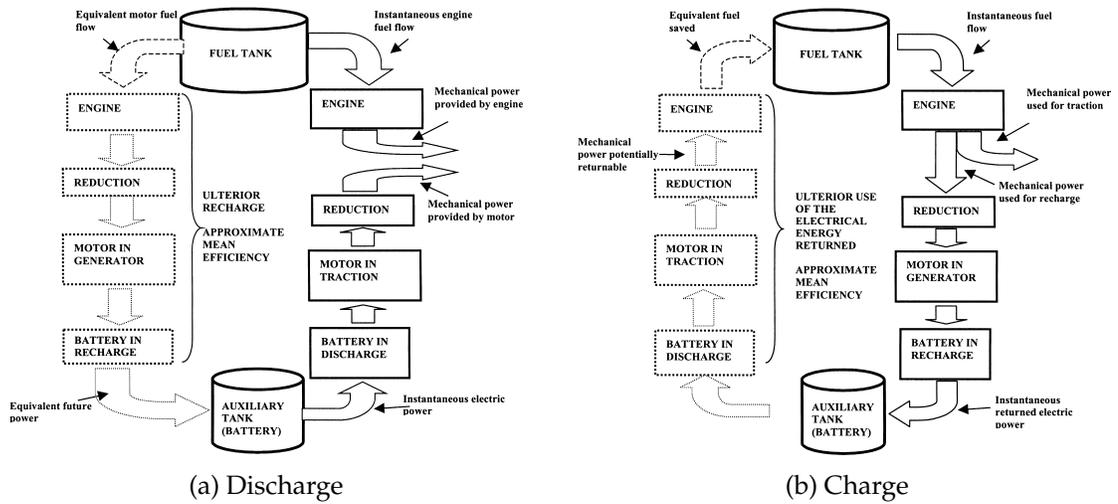


Figure 3.6: Energy path during charge and discharge in a parallel HEV [7]

virtual fuel consumption associated with the use of the electrical rechargeable energy storage system,  $P_{elec}$  the power delivered by the electric actuator(s).  $s$  is called *equivalency factor* and is used to convert electrical power into equivalent fuel consumption; it plays an important role in the ECMS, as will be shown later. Depending on the sign of  $P_{ress}$  (i.e., on whether the battery is charged or discharged), the virtual fuel flow rate can be either positive or negative, therefore the equivalent fuel consumption can be higher or lower than the actual fuel consumption. This definition is rather generic and can represent the basic energy flow in any kind of hybrid electric powertrain; however, for the strategy implementation,  $P_{elec}$ ,  $\dot{m}_f$  should be expressed as a function of the road load (power required to satisfy the driver's demand). This requires the knowledge of the powertrain topology and of the efficiency characteristics of the powertrain components. In particular, the path followed by the power flow in each operating mode of the powertrain influences the overall efficiency and therefore the relation between net power (road load) and total fuel consumption and electric power.

The concept of equivalent fuel consumption is illustrated in Figure 3.6, which refers to a parallel HEV (for a series HEV the only difference is the location of

the power summation node). In the discharge case (Figure 3.6.a), the electric motor provides mechanical power. The dotted route is related to the future return of the used instantaneous electrical energy. Of course, the operating point of this recharge cannot be known a priori, and thus an approximate mean efficiency should be set. In the charge case (Figure 3.6.b), the electric motor receives mechanical energy and converts it into electrical energy stored in the battery. The dotted route is related to the future use of this electrical energy to produce mechanical power. This amount of mechanical energy will not have to be produced by the engine and is considered as a fuel saving. In this case the equivalent fuel flow of the electric motor is negative.

The global problem of minimizing the total cost  $J$  can then be reduced to the local (instantaneous) problem of minimizing  $\dot{m}_{eqv}$ . The instantaneous minimization problem is computationally less demanding than the global problem solved with dynamic programming, and applicable to real-world situations since it does not rely (explicitly) on information about future driving conditions. For any vehicle operating point (speed, power required from the powertrain) the entire range of possible power splits is examined, and the equivalent fuel flows are determined for every combination. The combination with the lowest instantaneous fuel cost is selected. This can be achieved using the actual efficiency maps of the engine and electric machines.

### 3.5.2 Equivalence factor and charge-sustainability

The concept of equivalent fuel consumption is tied with the necessity of attributing a meaningful value to the equivalency parameter  $s$ . This parameter is representative of future efficiency of the engine and the energy storage device, and its value affects both the charge sustainability and the effectiveness of the strategy: if it is too high, an excessive cost is attributed to the use of electrical energy and therefore the full hybridization potential is not realized; if it is too low, the opposite happens and the RESS is depleted too soon (loss of charge sustainability).

In practical implementations, for more robustness in terms of charge sustainability, the use of the electrical energy storage is further penalized when the SOC is

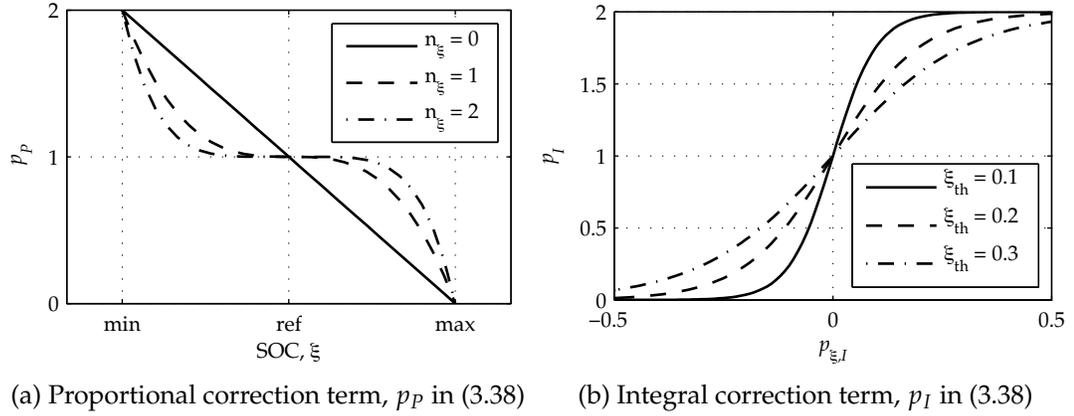


Figure 3.7: The two factors in the ECMS correction term [8]

low, and encouraged when the SOC is high. It is therefore critical to have a reliable on-line estimation of the battery state-of-charge.

The penalization is implemented multiplying the equivalence factor by a correction term. An expression for such term is [8]:

$$p(\xi) = p_P \cdot p_I = \left( 1 + \left( \frac{\xi_{ref} - \xi(t)}{\Delta\xi} \right)^{2n_{\xi}+1} \right) \cdot \left( 1 + \tanh \left( \frac{p_{\xi,I}(\xi(t))}{\xi_{th}} \right) \right) \quad (3.38)$$

$$p_{\xi,I} = 0.99 \cdot p_{\xi,I}(t - \Delta t) + 0.01 \cdot (\xi_{ref} - \xi(t)) \quad (3.39)$$

where  $\xi(t)$  is the instantaneous value of state of charge,  $\xi_{ref}$  is the desired nominal (average) value, and  $\Delta\xi$  is the range of variation around  $\xi_{ref}$  (symmetrical with respect to  $\xi_{ref}$ );  $n_{\xi}$  is an integer number greater than, or equal to, zero (a value of 2 or 3 is typical). The first factor represent a proportional correction term, the second factor is an integral correction term.  $\Delta t$  is the sampling time used to update the ECMS, and  $SOC_{th}$  is the tolerance of the hyperbolic tangent function. The correction function  $p(\xi)$  defined in this way multiplies the equivalence factors and artificially increases or decreases their value near the boundaries of the desired SOC interval.

It was shown [41, 18, 72] that very good results, comparable to those achieved with dynamic programming, are obtained by using two values of the equivalency factor  $s$ , one for charging ( $s_{ch}$ ) and the other for discharging ( $s_{dis}$ ), each of them constant during a driving cycle. These values are different for different driving cycles and must be obtained with a numerical optimization procedure, which is possible if the driving cycle is known a priori. Therefore, in ideal conditions (simulation of a known cycle), the results obtained by implementing the ECMS are very close to optimal. However, since the strategy is very sensitive to these parameters, the control works well only on driving cycles very similar to those in which and were obtained. This means that, despite its “instantaneous” formulation, ECMS still implicitly relies on some information about future driving conditions. If this information is wrong, i.e. if the vehicle is driven on a driving cycle sensibly different than the one for which the strategy was tuned, the control still works, but the results are not as good as they could potentially be.

### 3.5.3 Adaptive ECMS

The fact that the ECMS results are rather sensitive to the values of the equivalence factors means that there is an optimal value of  $s_{ch}$  and  $s_{dis}$  for each particular driving cycle, or at least for a given driving pattern (e.g. highway, city, suburban roads etc.). The adaptive ECMS, or A-ECMS is a development of ECMS in which the equivalence factors are not pre-coded, but rather calculated online, and updated at regular intervals.

One method to implement A-ECMS [41, 9] is to estimate the equivalence factors on the basis of a small section of the driving cycle (around 100 s), composed by measurements from the recent past and short-term (10-20 s) prediction of future driving conditions (with autoregressive models). In other words, the optimization procedure for  $s_{ch}$  and  $s_{dis}$  is repeated every few minutes, using vehicle speed measurement and a simplified vehicle model embedded in the vehicle controller. Despite the small horizon, this is still computationally heavy and a simplification is necessary for actual real time implementation: therefore, in [41] and [9] only one equivalency factor is calculated, for both charge and discharge. This introduces

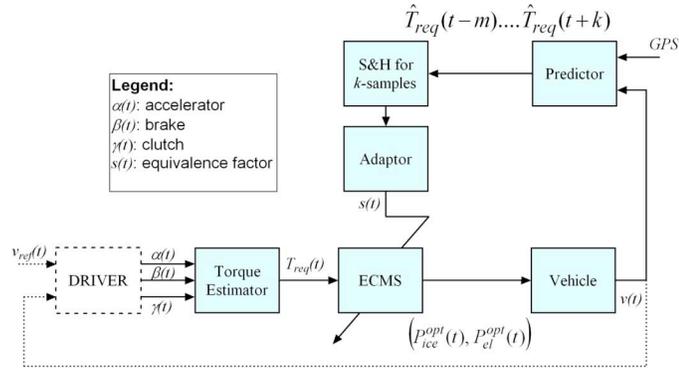


Figure 3.8: Control diagram of adaptive ECMS with online optimization [9]

some approximation, but the equivalency factor is representative of present driving conditions and the overall process works well. The performance of A-ECMS is slightly inferior to the standard ECMS tuned on a perfectly known driving cycle, but in general the results are quite good, and, most importantly, achievable in real world application (if enough computational power is available).

An alternative adaptive ECMS strategy [10, 19] exploits the fact that the equivalency factors are similar for cycles with similar statistical properties. A pattern recognition algorithm is used to identify which kind of driving conditions the vehicle is undergoing, and select the most appropriate equivalency factors from a predefined set. The optimal values of  $s$  for several cycle typologies (city, highway, etc.) are pre-calculated and stored in memory; during vehicle operation, the adaptation algorithm uses the past and present driving conditions to determine the current cycle type, and therefore selects the appropriate equivalency factor. This new implementation is faster than the previous one and may be more suitable for production environment.

The equivalent consumption minimization strategy is capable of dealing well with multi-objective optimization, for example in the case in which one of the objectives is the reduction of pollutant emissions. In fact, this can be achieved by reformulating the instantaneous cost in order to include the pollutant emissions as well as the actual and virtual fuel consumption [79, 9].

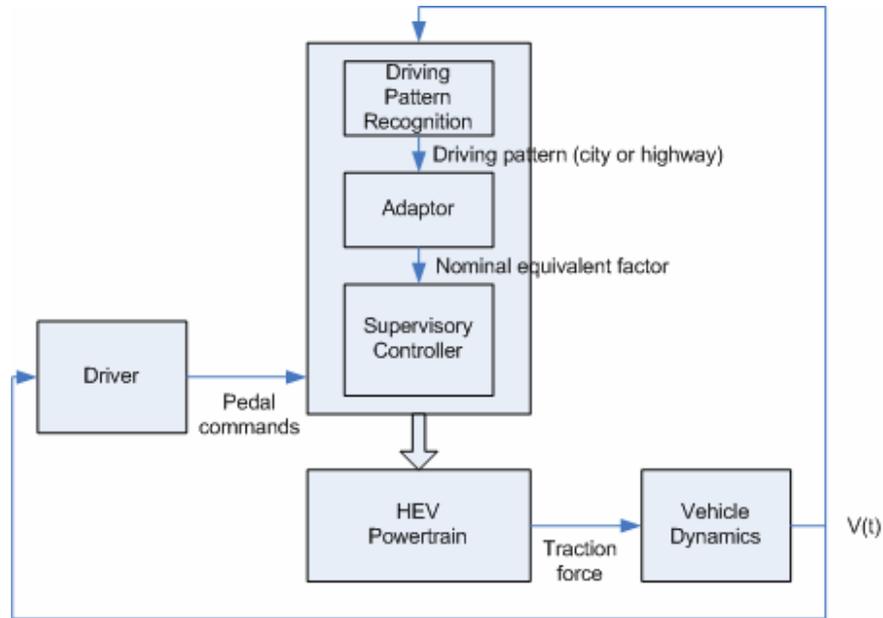


Figure 3.9: Control diagram of adaptive ECMS with driving pattern recognition [10]

### 3.5.4 Implementation issues

At any given time, the operating conditions of the vehicle (driveline speeds, power required from the powertrain, battery state of charge, etc) determine the limits within which the engine, the electric machines, and the battery can operate. For example, the engine and motor speed determine the maximum torque that each machine can deliver; the battery state of charge determines whether or not the battery can supply or accept charge; and the total power demand must be met by a combination of electrical and engine power. The acceptable values of the control variable are those that respect all these constraints.

In order to implement this control strategy, the range of acceptable values is discretized into a set of candidate control actions: for example, in a simple series HEV, the control action is the percentage of the total power that comes from the battery; this can range from 0 to 100%, and 20 or 30 possible values could be considered in the discretization. The equivalent consumption defined by 3.37 (which

is, obviously, a function of the control variable) is evaluated for each of the control candidates, and the one among them that generates the lowest equivalent consumption is selected. The evaluation of equivalent consumption is obtained using efficiency maps of the various components.

If more than a degree of freedom is present (for example, if the power can be delivered by two electric machines and the engine), the number of control actions to evaluate and compare increases sensibly: for example, with two degrees of freedom and 20 possible values for each, the number of possible combinations is 400. Clearly, evaluating each of these points at each instant of time is computationally demanding in a vehicle application (outside of prototyping environment). To overcome this problem, two methods can be used: pre-calculation of the optimal power split, or online calculations with reduced search space.

The first method consists in calculating, off-line, the optimal power split given all the possible values (discretized) of all strategy variables: given the entire range of power demand, engine speed, electric machine speed, battery SOC, equivalency factors etc., the equivalent fuel consumption is calculated for every combination, and the optimal power split corresponding to the given operating conditions is stored in a map. During vehicle operation, the operating conditions are measured and matched to those stored in the map to retrieve the optimal power split at each time. This approach reduces the processor requirements for the online implementation, and is very easy to implement, but presents some disadvantages: it implies the need for large memory on board; it prevents in-vehicle tuning, since everything is stored in memory; and interpolation of the resulting multi-dimensional maps may generate unexpected results.

The online calculation approach is more flexible, since it does not store any pre-calculated results, and does not suffer from interpolation issues. However, calculating the equivalent fuel consumption for a large number of cases is very demanding computationally. One solution is to reduce the number of cases to evaluate, by considering as possible control actions only those contained in a “box” around the previous value, instead of the entire range. The size of this box and the number of elements in it can be a function of the available computation capabilities.

A difference between the map and the online calculation approach is that the amount of data stored in the maps cannot be reduced, while the code can be optimized (for production environment) and therefore run much faster than the code produced by automatic compilation of a Simulink diagram.

### **3.5.5 Chattering issues**

A rather common problem with ECMS implementations is the presence of chattering in the control outputs. This means that the distribution of power demand keeps oscillating very quickly between two distinct values, perhaps very different from each other. The reason for this is that the equivalent fuel consumption in two separate points of the control space may be very similar, and small variations in the driving conditions make one of the two points being, alternatively, a little better or a little worse than the other one. The consequence is a very high frequency of switching between operating modes (for example from electric traction to engine traction), and therefore poor comfort for the passengers, who perceive this chattering as “roughness” in the vehicle behavior.

Several methods can be used to prevent or reduce this phenomenon. First, the inputs to the ECMS should be filtered, to limit the influence of noise (or in general of small perturbations) on the value of equivalent fuel consumption. Also, reducing the size of the search box can help preventing large oscillations, since these will be limited to the range of controls considered. Another approach could be to introduce some hysteresis, and therefore preventing the control from changing if the gain in terms of equivalent fuel consumption is below a given threshold. Equivalently, one could slightly penalize solutions that differ from the previous ones, in order to prevent change when the gain would be too small.

## **3.6 From the minimum Principle to the ECMS**

The concepts behind ECMS (i.e. equivalence, in energy terms, of fuel consumption and electrical energy) are originated from engineering intuition. However, an analytical derivation of the equivalent fuel consumption can be obtained using Pontryagin’s minimum principle. In fact, the ECMS and the minimum principle

share the same underlying idea of reducing the global optimization problem to a local one: the ECMS by defining an equivalent fuel consumption, the minimum principle by minimizing the Hamiltonian  $H$  at each time  $t$ .

Consider the hybrid electric powertrain as a system described by the dynamic equation

$$\dot{\zeta}(\zeta, P_{ress}) = -\varepsilon_{ress}(\zeta, P_{ress}) \frac{P_{ress}(t)}{E_{ress}} \quad (3.40)$$

where  $\zeta$  is the RESS state of energy,  $E_{ress}$  its total energy capacity, and the control variable is the RESS power:  $u = P_{ress}$ . This state equation is typical of any vehicle in which the state of energy of the RESS represents the most relevant dynamic variable. The term  $\varepsilon_{ress}(\zeta, P_{ress})$  represents the charge or discharge efficiency (see Section 4.2.2 for a detailed explanation of its meaning), that accounts for losses in the energy storage system and its power electronics, and in general can depend on both the state of energy  $\zeta$  and the RESS power  $P_{ress}$ .  $\varepsilon_{ress}(\zeta, P_{ress})$  is related to the efficiency as follows:

$$\varepsilon_{ress} = \begin{cases} \frac{1}{\eta_{ress}} & \text{if } P_{ress} > 0 \\ 1 & \text{if } P_{ress} = 0 \\ \eta_{ress} & \text{if } P_{ress} < 0 \end{cases} \quad (3.41)$$

If the minimization objective is the fuel consumption and the vehicle is charge-sustaining, that is, the initial and final SOE are the same, the cost function to be minimized is expressed as

$$J = \int_{t_0}^{t_f} L dt = \int_{t_0}^{t_f} \dot{m}_f(P_{ress}) dt. \quad (3.42)$$

According to Pontryagin's minimum principle, the Hamiltonian of this problem is given by

$$H = \lambda \dot{\zeta} + L = -\lambda(t) \varepsilon_{ress}(\zeta, P_{ress}) \frac{P_{ress}(t)}{E_{ress}} + \dot{m}_f(P_{ress}) \quad (3.43)$$

where the co-state  $\lambda$  must satisfy the dynamic equation

$$\dot{\lambda}(t) = -\frac{\partial H}{\partial \zeta} = \lambda(t) \frac{P_{ress}(t)}{E_{ress}} \frac{\partial \varepsilon_{ress}(\zeta, P_{ress})}{\partial \zeta}. \quad (3.44)$$

It is immediately clear that this expression of the Hamiltonian has the same structure as the ECMS equivalent fuel consumption

$$\dot{m}_{eqv} = \dot{m}_{ress} = \frac{s}{Q_{lhv}} P_{ress} + \dot{m}_f, \quad (3.45)$$

because both are the sum of the actual fuel consumption  $\dot{m}_f$  and of a term proportional to the RESS power. The expressions (3.43) and (3.45) are identical if

$$s(t, \zeta, P_{ress}) = -\lambda(t) \varepsilon_{ress}(\zeta, P_{ress}) \frac{Q_{lhv}}{E_{ress}}. \quad (3.46)$$

This shows that ECMS is an implementation of the minimum principle if the equivalence factor is time varying depending on the co-state  $\lambda(t)$ , on the actual battery efficiency at a given instant, which can be easily modeled using a map of  $\varepsilon_{ress}(\zeta, P_{ress})$ , and on the ratio of fuel energy content and RESS energy capacity. The latter two are constant vehicle parameters, while the co-state is a function of both vehicle and driving cycle. The standard ECMS formulation in which the equivalence factor is only a function of the sign of  $P_{ress}$  can be seen as an approximation of the minimum principle solution.

However, the assumption of constant equivalence factors is valid in the case in which the efficiency  $\eta_{ress}$  is constant and does not depend on  $\zeta$  nor on  $P_{ress}$ , because this implies  $\dot{\lambda} = 0$  (from (3.44)) and therefore  $\lambda(t) = \lambda_0 \forall t$ ; furthermore, the term  $\varepsilon_{ress}$  in (3.46) is only dependent on the sign of  $P_{ress}$ . In this case of constant efficiency, replacing the definition (3.41) of  $\varepsilon_{ress}$  into (3.46) allows to derive an interesting relation between the charge and discharge equivalence factors:

$$\begin{cases} s_{dis} = s(P_{ress} > 0) = -\lambda_0 \frac{1}{\eta_{ress}} \frac{Q_{lhv}}{E_{ress}} \\ s_{chg} = s(P_{ress} < 0) = -\lambda_0 \eta_{ress} \frac{Q_{lhv}}{E_{ress}} \end{cases} \implies \frac{s_{chg}}{s_{dis}} = \eta_{ress}^2 \quad (3.47)$$

## 3.7 Model predictive control

### 3.7.1 Overview

Model Predictive Control (MPC) is defined in a family of control techniques that make an explicit use of a model of the process to obtain the control signal by minimizing an objective function<sup>5</sup>.

This technique originated in process control for chemical plants, characterized by slow dynamic processes, which give enough time for the optimization computations to be performed within control updates. With the increasing computational power available on microcontrollers, MPC is now becoming increasingly popular in all control applications, including systems with faster dynamics.

Model predictive control represents the solution of a standard optimal control problem over a finite horizon, performed online using a model to predict the effect of the control on the system output. It also represents an implementation of the receding-horizon strategy: at each instant, the optimal control sequence for the prediction horizon is calculated, but only the first element is applied; then, at the next time step, the prediction horizon is displaced towards the future.

As it is easily inferred from these characteristics, MPC is a control technique that requires high computational effort and an accurate model of the system in order to give good results; on the other hand, it can be applied to many cases and can be very effective if the reference trajectories are known. This does not happen in the case of vehicular applications, and some prediction techniques must be used in order to implement MPC as an energy management strategy for HEVs.

### 3.7.2 Receding Horizon Technique

MPC is based on iterative, finite horizon optimization of a plant model. At time  $t$  the current plant state is sampled and a cost minimizing control strategy is computed (via a numerical minimization algorithm) for a relatively short time horizon in the future:  $[t, t + N]$ . Specifically, an online or on-the-fly calculation is used to explore state trajectories that emanate from the current state and find a

<sup>5</sup>This definition, and the description in this section, are mostly based on [11].

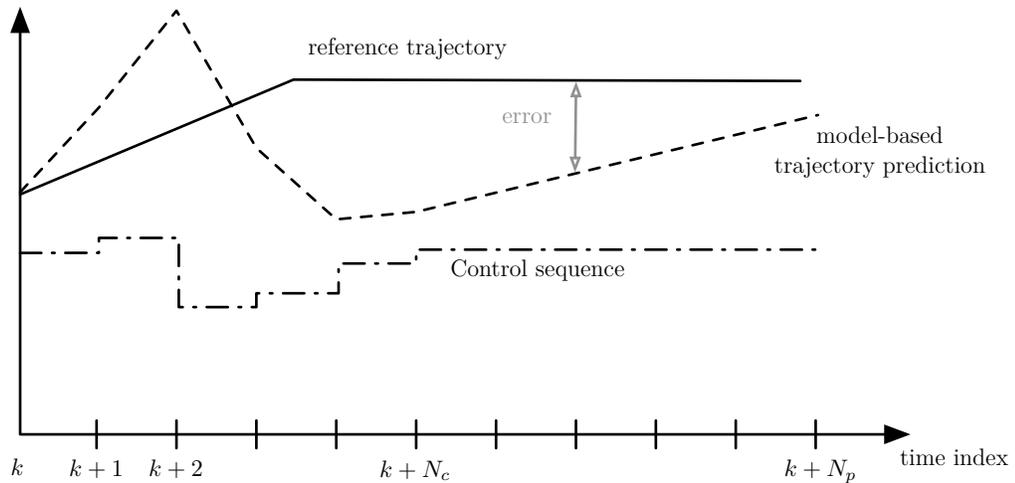


Figure 3.10: Examples of reference trajectory, actual trajectory, and control sequence

cost-minimizing control strategy until time  $t + N$ . Only the first step of the control strategy is implemented, then the plant state is sampled again and the calculations are repeated starting from the now current state, yielding a new control and new predicted state path. The prediction horizon keeps being shifted forward and for this reason MPC is a *receding horizon control*.

The following steps, illustrated in Figure 3.10, represent a typical MPC implementation [11]:

1. The future outputs  $y$  for a determined horizon  $N_p$ , called the prediction horizon, are predicted at each instant  $k$  using the process model. These predicted outputs  $y(k + i | k)$ ,  $i = 1, \dots, N_p$  depend on the known values up to instant  $k$  (past inputs and outputs) and on the future control signals  $u(k + i | k)$ ,  $i = 0, \dots, N_c - 1$ , which are those to be sent to the system and to be calculated.  $N_p$  indicates the length of the prediction horizon and  $N_c \leq N_p$  is the length of the control horizon (expressed in number of time steps).
2. The set of future control signals (from time  $k$  to time  $k + N_c - 1$ ) is calculated by optimizing a given criterion in order to keep the process as close as

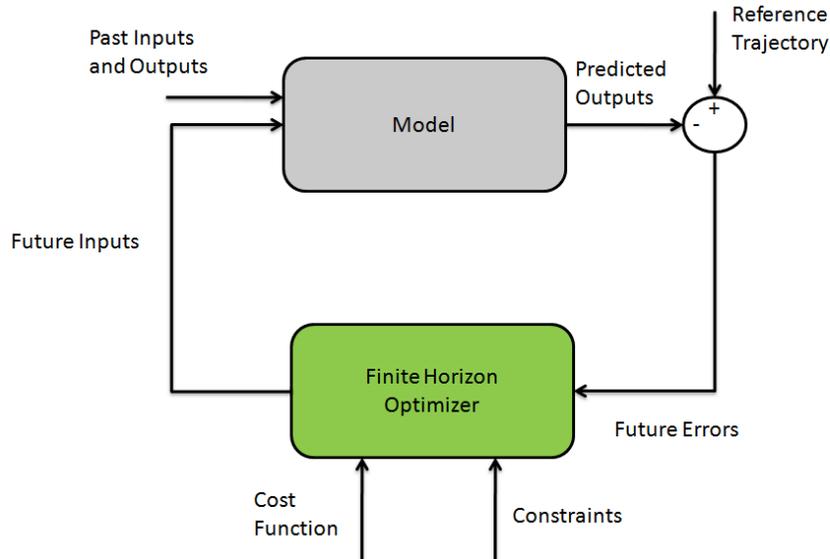


Figure 3.11: Basic structure of MPC

possible to the reference trajectory or set point. This criterion usually takes the form of a quadratic function of the errors between the predicted output and the reference trajectory. The control effort is included in the objective function in most cases. An explicit solution can be obtained if the criterion is quadratic, the model is linear and there are no constraints, otherwise an iterative optimization method has to be used.

3. The control signal  $u(k | k)$  (i.e., the control at time  $k$  calculated based on the information available at time  $k$ ) is sent to the process while the next control signals ( $u(k + 1 | k)$ ,  $u(k + 2 | k)$  etc.) are calculated but not used, because at the next sampling instant  $y(k + 1)$  will be known and thus the optimal control value will be recalculated. The procedure is repeated with this new value and the entire sequence is brought up to date. Then  $u(k + 1 | k + 1)$  is calculated (which in principle will be different from  $u(k + 1 | k)$  because of the new information available) using the receding horizon concept.

An example that helps understanding the MPC concepts is the analogy with the way people drive a car [11]. The driver decides how to control the car in order

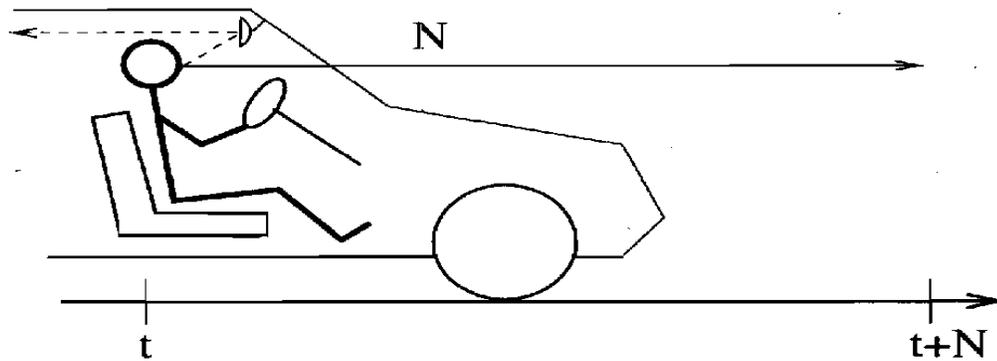


Figure 3.12: Applying MPC is like driving a car: the drivers decide what to do predicting the consequences of their actions [11].

to follow the desired trajectory based on the knowledge of the desired reference trajectory and on his ability to predict the consequences of the control actions on the behavior of the car. The decision is repeated at every instant, always shifting the prediction horizon forward and updating the prediction on the basis of the present condition.

Although the MPC approach is not strictly optimal, in practice it has given very good results, despite the fact that a thorough theoretical understanding of MPC optimality has not been formalized yet.

### 3.7.3 Application to energy management of HEVs

A few works have been published on the application of model predictive control (or at least, of some of its concepts) to hybrid electric vehicles.

Back et al. [80] examined the potential for the application of predictive control to a parallel hybrid electric powertrain. Assuming constant speed and using GPS information to estimate the road slope during the prediction horizon, the torque request at the wheels during the prediction horizon is calculated, then a dynamic programming problem is solved in order to obtain the optimal solution. The optimal control sequence of control inputs is obtained by minimizing a cost function representing the total fuel consumption during the prediction horizon. Only the

first control input is applied, then the prediction horizon is shifted forward and the procedure is repeated. This gives a closed loop controller (state feedback). Because of the concept of receding horizon, the quantization grid of the DP algorithm is finer at the beginning of the prediction horizon and coarser at the end, allowing for faster on-line computation. The updating of the states of the system into the cost function ensures convergence. The simulation results suggest that results of this strategy improve, as expected, with the length of the prediction horizon.

A similar approach (receding horizon dynamic programming) is proposed by Nuijten et al. [81] for the case of a conventional vehicle with a 42-V electric power net and an advanced alternator that can also supplement torque to the driveline when needed. The problem is therefore very similar to the power split in a parallel hybrid vehicle. The prediction of future power demand is assumed to be perfect, and the results show once again the benefit of a longer prediction horizon on the strategy performance.

De Vito et al. [82] presented the application of model predictive control to a fuel cell vehicle; in this case, the algorithm uses a model of the fuel cell to predict its dynamic response and thus decide the battery power needed in order to satisfy the driver's power demand with the fastest dynamic response possible, while minimizing the overall use of hydrogen fuel and maintaining the battery state of charge around its nominal value. These objectives are met using a cost function which is the sum of three terms: the hydrogen use, the deviation of state of charge from the reference value, and the difference between power demand and power supplied.

### **3.8 Rule-based control strategies**

The most common way of implementing supervisory control in a hybrid electric vehicle is to introduce a set of rules that, given the observed values of some meaningful parameters, decide the power split between the engine and the energy storage device. Unlike techniques based on optimal control or equivalent consumption minimization, rule-based control [12, 83, 45, 84, 46, 85, 44] does not rely on formal (ODE) models and formal description of the problem. Instead, the rules

are based on engineering intuition: the objective is to make each element of the powertrain work in high-efficiency conditions, and to recuperate as much energy as possible from regenerative braking. A simple rule structure looks like the one shown in Figure 3.13. The rules are usually in the form if-then-else, and efficiency maps or fuzzy-logic methods may be used in the implementation. Since rule-based controllers are based only on instantaneous conditions, it is very easy to take into account the local constraints (i.e., limitations on power, torque, speed etc). On the other hand, it is impossible to formally guarantee the optimality of the solution and the respect of the integral constraints, for example the charge-sustainability: in this case, the rules can only force a given integral measure (typically the state of charge) to remain between two limits. The parameters of the rule-based controller (e.g. the threshold values that decide when to switch from a mode to another) are usually obtained through calibration based on modeling and simulation of the powertrain, possibly using optimization techniques. Optimal control results such as those described earlier (especially dynamic programming) can be used to benchmark or validate the effectiveness of the strategy [9, 72] or as a guideline to determine the control rules [44, 46].

The main advantage of rule-based control is its conceptual simplicity and the ease of implementation on running vehicles. The rules can be made as complex and detailed as needed and (in principle) can take care of any special event that may affect the vehicle, thus incorporating “limp-mode” capabilities in the supervisory control itself. The main disadvantages are the lack of proof of optimality, and the fact that there is no standard methodology for synthesizing the rules (i.e., the rules are decided on a case-by-case basis but there is no way to determine a priori that a given set of rules is appropriate for a given application). Also, the presence of many thresholds and parameters makes it quite difficult to obtain an appropriate calibration that works for a wide variety of driving conditions. Nonetheless, rule based strategies are widely used in production environment, possibly in conjunction with other algorithms based on optimal control. For example, it may be possible to implement effectively ECMS only in a part of the vehicle operating range, using the rule-based controller for the remaining conditions [44].

```

* A Thermostat is working in the background

* If the lower SOC value is reached , then,:

- APU is on
- Papu=maximum(Popt,Pcm)
- Papu <= Pmax
- Pbatt=Pcm-Papu

* If the higher SOC value is reached, then:

- APU is idle if (Pcm <= Pmin )
  # Papu=0
  # Pbatt=Pcm

- APU is on if (Pmin < Pcm < Pmax)
  # Papu=Pcm
  # Pbatt=0

- APU is on if (Pcm >= Pmax)
  # Papu=Pmax
  # Pbatt=Pcm-Papu

* In any case:

- If Acrm > 3.5 m/s^2 then Papu=P_max_rated
- If Pcm < 0 then Regenerative Braking is active provided:
  # V > 5mph

* P_batt_ch <= P_batt_ch_max
* P_batt_disch <= P_batt_disch_max

```

Figure 3.13: An example of rule-based control [12]

## **3.9 Implementation issues common to all energy management strategies**

Several issues need to be considered when implementing a supervisory energy management strategy in a vehicle. This section will address the most important of them from a general standpoint.

### **3.9.1 Stability**

Stability for an energy management strategy is an open issue that has not been much researched. However, some stability properties are intuitively necessary for practical implementation. A first step for the formalization of the stability problem is presented in Section 4.12, while qualitative considerations are made here.

In a charge-sustaining hybrid, it is important to keep the RESS state of charge (or state of energy) in the acceptable range (between minimum and maximum values) at all times. An energy management strategy is defined stable if it is such that, in presence of a constant power request, the state does not diverge from a given reference value, which is typically the average between the two operating limits. Therefore, if the power request remains constant for an arbitrarily long time, the SOE or SOC should either remain constant or converge to the reference value. This kind of stability is necessary for practical implementations in order to guarantee the appropriate behavior of the strategy at all times: in fact, when the SOE or SOC is at the upper or lower bound, the RESS cannot be fully exploited (for example, it cannot be charged while at the upper bound), which prevents the strategy to operate optimally.

### **3.9.2 Interface with low-level controllers**

It is very important to interface correctly the energy management strategy to the low-level controllers present on the vehicle components.

#### **Engine controller**

The engine control in a conventional vehicle transforms the driver's input (accelerator pedal position) into fueling commands (timing, injection amount etc) and

other actuations. The strategy behind these actions is rather complex, is based on minimization of fuel consumption and emissions, and requires a lot of calibration.

When the same engine is used in a hybrid electric vehicle, its controller must be modified in order to work based on a different set of inputs. In particular, the accelerator pedal position is replaced, as an input, by torque or power request. Another, more important difference, is that the operating points in which the engine operates most often can change sensibly, and in general can be more easily controlled (to different degrees depending on the kind of hybrid architecture considered). This introduces the need to re-calibrate the engine in order to further optimize its emission and fuel efficiency, matching it to the energy management strategy.

Another example of conflict between energy management strategy and engine controller is the effect of idle speed controller. When the driver is not requesting any power from the engine (accelerator pedal released), the idle speed controller tends to maintain the engine speed at idle level, rejecting all external disturbances. Imagine now that the HEV energy management strategy decides to use the unloaded engine to charge the battery instead of just idling, applying load torque from the electric starter/alternator. The engine idle controller, in this case, sees the additional load torque as a disturbance and tries to contrast it, not allowing the engine speed to increase, working against the energy management strategy. There is an obvious conflict in this case, which can be solved by giving to the energy management strategy the ability to suppress engine idle control in certain conditions.

### **Transmission controller**

The power demand generated by the energy management strategy is satisfied by given combinations of torque and speed of the various machines. If a stepped transmission is present, it is important to coordinate the gear shifting scheduling with the results of the energy management strategy. If possible, the energy management strategy should determine the gear to engage at each time. The engine and electric motor controllers must also be coordinated with the transmission controller in order to allow for smooth gear shifting.

## **ABS and other vehicle dynamics controllers**

The presence of additional braking torque deriving from the traction machine(s) may interfere with the functioning of the ABS and other safety systems, such as ESP or traction control.

## **Special events and need for a supervisory controller**

Some events in the operation of a vehicle may require specific actions to be taken, thus making necessary to force the switch from one state to a different one, overruling the energy management strategy. This makes necessary the presence of a supervisory control layer, positioned above the energy management strategy.

### **3.9.3 Power demand**

The main input to the energy management strategy is the total power demand from the vehicle. This can be calculated in several ways, and it has a prominent effect on the results of the strategy implementation.

The most common method for deciding the total power demand is to consider the total available power and the position of the accelerator pedal .

The available power is the sum of the maximum power that each machine can deliver at a given time. In a hybrid vehicle, can vary greatly during operation, because of the different modes in which the transmission can be: for example, the total available torque can be either the sum of engine and traction motor torque (if the battery SOC is high enough), or just the engine torque (when battery SOC is too low). The variation of available power means that, if the driver maintains the accelerator pedal at a fixed position, the power request changes over time, which is not acceptable.

Another issue to consider is that, in a conventional vehicle, usually the accelerator pedal position is not linearly proportional to the power (or torque) demand: the relation is similar to the one shown in Figure 3.14. Therefore, in order to achieve a predictable behavior, the suggested expression for positive (acceleration) power demand is

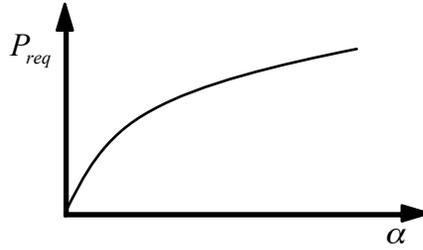


Figure 3.14: Relation between power demand and accelerator pedal position in a conventional vehicle

$$P_{req} = f(\alpha)P_{avail} \quad (3.48)$$

where  $f(\alpha)$  is a mapping of the accelerator pedal position  $\alpha$  into a suitable factor. The shape of the function  $f(\alpha)$  (which may also depend on other factors, e.g. vehicle speed and engine speed) is calibrated in order to maximize drivability and in general smoothness of operation; correct calibration of this map is very important.

### 3.9.4 Regenerative braking

When the power demand is negative (regenerative braking), the driver's demand is expressed using the brake pedal. In this case, to mimic the familiar operation of conventional vehicles, the negative torque demand should be proportional to the pedal position. In fact, in a conventional vehicle, the brake pedal position is proportional to the pressure in the hydraulic brake circuit, which in turn is proportional to the braking torque. There are two approaches to implement regenerative braking: "series" and "parallel". In the "series" method, regenerative braking is applied for low levels of deceleration, or until some electric power limit is reached; then, if more deceleration is desired, the mechanical brakes are used in addition to the regenerative braking. In the "parallel" method, electrical (regenerative) and mechanical braking are always blended together using some splitting strategy, for example giving prevalence to the regen when the deceleration request is lower,

and to the mechanical brakes in cases of higher deceleration or to come to a complete stop. Therefore, some degree of control or observation over the mechanical brakes is necessary, in order to blend the electrical and the mechanical braking consistently, providing acceptable feeling for the driver. Integration with the ABS or stability control ECUs becomes very important at the implementation level.

### **3.9.5 Actuator bandwidth**

The actuators in a hybrid electric powertrain are the engine and the electric motor. Since electric machines have a much faster response than internal combustion engines, they tend to provide the torque requested by the energy management strategy in a shorter time, which causes severe drivability problems. One solution is to artificially slow down the machines, filtering the torque request to a bandwidth compatible with the engine dynamics, in order to obtain comparable response time from both machines.

### **3.9.6 Battery characterization**

An accurate representation of the battery characteristics and limitations is very important for all kinds of energy management strategies. In particular, a correct estimation of the instantaneous state of charge is necessary, in order to make decisions that exploit the battery as much as possible without over-charging or over-discharging it. It is also important to model the variations of battery voltage due to power demand: if these are exceeded, or if the voltage limitations of an electric machine are exceeded (e.g., due to excessive power demand), then problems may occur, varying from drivability issues (torque “holes” due to machines going in safe mode and reducing their torque output) to the risk of damaging components (if they do not have integrated diagnostics/safety modes).

## **3.10 Conclusion**

Several energy management strategies for HEVs proposed in literature have been described in this chapter, giving an overview of their basic ideas and their possible implementation. [...]

## Chapter 4

### APPLICATION OF THE STRATEGIES

Chapter 3 presents an overview of the most important control strategies described in the literature for the energy management of hybrid electric vehicles. In this chapter, three of these strategies – dynamic programming (DP), Pontryagin’s minimum principle (PMP), and equivalent consumption minimization strategy (ECMS) – are applied to two case studies, with the aim to show that the ECMS, despite being a local minimization strategy, is capable of providing quasi-optimal results if appropriately implemented. Dynamic programming is used to provide the reference optimal solution, while Pontryagin’s minimum principle represents the connection between the optimal solution and the ECMS, as explained in Section 4.10.

The two case studies, introduced in Section 1.6, represent vehicles with completely different uses: a refuse collection truck (*case study 1*) and a mid-size SUV that represents a possible implementation of the OSU vehicle for the EcoCAR competition (*case study 2*). In the following, the two case studies are identified as *refuse truck* and *EcoCAR*.

The powertrain architecture in both cases is series hybrid electric, in which the engine drives a generator to power the electric traction motor(s) and/or recharge the energy storage device. However, there are some fundamental differences in the way the two vehicles operate. In the refuse truck, the energy storage device is a pack of ultracapacitors, which provide high power (matching the power rating of the traction motors), high charge/discharge efficiency, and virtually infinite operating life, but have limited energy capacity: therefore, they are completely charged

and discharged very frequently. The EcoCAR, on the other hand, is a plug-in hybrid electric vehicle, in which a large pack of lithium-ion batteries provides pure electric operation for a substantial driving distance, while the engine-generator is intended to be used as an auxiliary power unit (APU) to recharge the battery when needed. The two case studies thus represent the two extreme embodiments of a series HEV architecture.

This chapter is organized as follows: first, the optimal control problem is defined specifically for each of two case studies, providing details about the specific architecture and power flow; then, the solution method with dynamic programming, Pontryagin's minimum principle, and ECMS is shown, including simulation results and a comparison of the strategies.

## 4.1 General problem formulation

A general formulation of the energy management problem in HEVs was given in Section 3.1.1: it is briefly reported here for convenience.

Given the system described by the dynamic equation

$$\dot{x} = f(x(t), u(t), t), \quad (4.1)$$

find the control law  $u(t)$  in the time interval  $[t_0, t_f]$  such that the cost function

$$J = \phi(x(t_f), t_f) + \int_{t_0}^{t_f} L(x(t), u(t), t) dt \quad (4.2)$$

is minimized, and the constraints

$$\psi(x(t_f), t_f) = 0 \quad (4.3)$$

$$\begin{cases} G(x(t), t) \leq 0 \\ u(t) \in \mathcal{U}(t) \end{cases} \quad \forall t \in [t_0, t_f] \quad (4.4)$$

are satisfied.

This problem can be completely defined providing explicit definitions of the state variables  $x(t)$ , control variables  $u(t)$ , set of admissible controls  $\mathcal{U}(t)$ , and

the functions  $L(x, u, t)$  (instantaneous cost),  $\psi(x(t_f), t_f)$  (boundary conditions, or terminal constraints, i.e. final value of the state variables),  $G(x(t), t)$  (local constraints, i.e. admissible values of the states at each instant). The following sections describe these quantities for the two case studies.

## 4.2 Definition of the control problem for refuse collection truck (case study 1)

In this section, the energy management problem formulated in Section 4.1 is specifically developed for the first case study, the series hybrid electric refuse collection vehicle. The generic expressions appearing in the problem formulation are given explicitly for the case at hand, using an energy flow model of the powertrain. The content of this section can also be regarded as a control-oriented model of the system (as opposed to the physical models described in Chapter 2).

In order to avoid the divulgation of proprietary information, the model used in this section uses parameters different than the refuse collection prototype built by Oshkosh Corp. and described in Section 1.6. In particular, the original engine map has been replaced by using data in the public domain relative to a similar engine; thus, the simulation results are realistic but do not correspond to the actual truck.

The only state variable to be considered in the optimal control problem is the state of energy of the capacitors, denoted by  $\zeta$ ; the dynamics of all the powertrain actuators (engine, electric machines) are neglected, since their response is much faster than the transients of the state of energy. Hence,

$$x = \{\zeta(t)\}. \quad (4.5)$$

The control variable in this series hybrid configuration is just one, the electric power delivered by the capacitors:

$$u = \{P_{cap}\}. \quad (4.6)$$

The terminal cost  $\phi(x(t_f), t_f)$  is not defined, assuming that the value of the state of energy at the final time is not important: this is due to the fact that the

amount of energy stored in the capacitors is rather small and the effect of the capacitor residual energy on fuel consumption and operating cost is negligible. The instantaneous cost function  $L(x, u, t)$  is the engine fuel consumption:

$$L(x, u, t) = \dot{m}_f(\zeta, P_{cap}, t) \quad (4.7)$$

The constraints on the final value of the state  $\psi(\zeta(t_f), t_f)$  are not defined (unconstrained final state), while the dynamic constraints enforce the state of energy and the control variable to remain within the allowable range:  $\zeta_{min} \leq x(t) \leq \zeta_{max}$  and  $P_{cap,min}(t) \leq P_{cap}(t) \leq P_{cap,max}(t)$ .

The following sections show the derivation of an explicit expression for the system dynamics equation, the cost function, the values of the constraints on the state  $\zeta$  and on the control variable  $P_{cap}$ .

#### 4.2.1 Power flow diagram

The starting point for a complete problem definition is the analysis of the power flow, with the objective of finding an expression that relates the fuel consumption to the control variable of the energy management strategy (i.e., the capacitor power). The powertrain architecture is described in Section 2.5 and the power flow diagram is shown in Figure 4.1.

The internal combustion engine is the primary energy converter and produces the mechanical power  $P_{ice}$  using the fuel power  $P_{fuel} = Q_{lhw}\dot{m}_f$  ( $Q_{lhw}$  is the fuel lower heating value, i.e. its energy content per unit of mass;  $\dot{m}_f$  is the fuel mass flow rate). This power is used to drive the electrical generator, the secondary accessories ( $P_{acc}$ , which includes air conditioning compressor, air brake compressor, water pump, oil pump etc.), and the power takeoff accessories (PTO), i.e. the hydraulic mechanisms for refuse loading, packing, and dumping. The generator transforms the mechanical power  $P_{gen,m}$  taken from the engine into the electrical power  $P_{gen,e}$ . The rechargeable energy storage system (RESS) is a pack of electrochemical supercapacitors that deliver the power  $P_{cap}$ . The electrical power from the generator and from the capacitors is summed electrically in the electric bus and is used to drive the two traction motors. These use the electrical power  $P_{em1,e}$  and

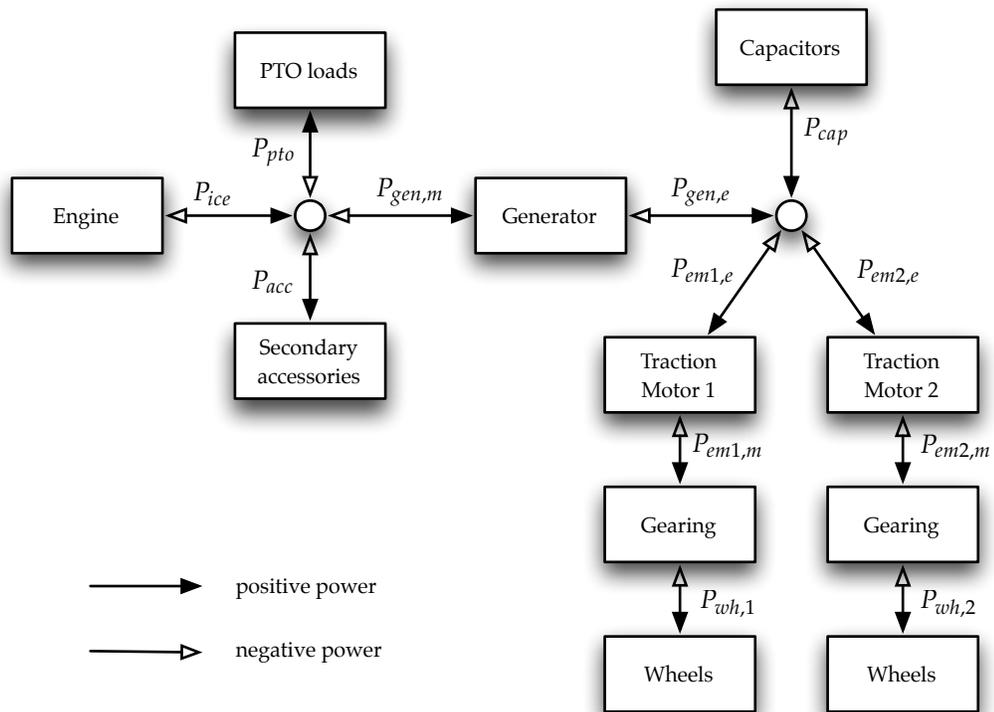


Figure 4.1: Series hybrid electric architecture

$P_{em2,e}$  to produce the mechanical power  $P_{em1,m}$  and  $P_{em2,m}$  during traction operation (vice versa, they produce electrical power from mechanical power during braking operation). The sign convention of each of these quantities is defined in the figure using a filled arrowhead for positive power flow, and an empty arrowhead for negative power flow.

There are two power summation nodes in the diagram of Figure 4.1: one mechanical, one electrical. The corresponding equations, expressing the conservation of power in each node, are:

$$P_{ice}(t) = P_{pto}(t) + P_{acc}(t) + P_{gen,m}(t) \quad (4.8)$$

and

$$P_{gen,e}(t) + P_{cap}(t) = P_{em1,e}(t) + P_{em2,e}(t). \quad (4.9)$$

The electric power at the two traction machines is a function of the road load and of the mechanical efficiency of the machines; the accessory power and PTO power depend on the load conditions. Eight variables appear in (4.8) and (4.9): four of these ( $P_{pto}$ ,  $P_{acc}$ ,  $P_{em1,e}$  and  $P_{em2,e}$ ) are determined by the characteristics of the driving and loading cycle and can be considered as time-varying parameters (in other words, they vary with time but do not depend on the control strategy);  $P_{gen,m}$  and  $P_{gen,e}$  are related biunivocally by the efficiency of the generator:

$$P_{gen,m} = \begin{cases} P_{gen,e} / \eta_{gen}(P_{gen,e}, \omega_{gen}) & \text{if } P_{gen,e} \geq 0 \text{ (generating mode)} \\ P_{gen,e} \cdot \eta_{gen}(P_{gen,e}, \omega_{gen}) & \text{if } P_{gen,e} < 0 \text{ (motoring mode)} \end{cases} \quad (4.10)$$

This leaves three independent variables ( $P_{ice}$ ,  $P_{cap}$ , and one between  $P_{gen,m}$  and  $P_{gen,e}$ ) and two equations, which means that there is one degree of freedom in the power split, corresponding to the repartition of the power request between the two on-board energy sources: the fuel (powering the engine) and the capacitors. The control variable of the power split problem can be any of the three power levels  $P_{ice}$ ,  $P_{cap}$  and  $P_{gen,e}$  (or  $P_{gen,m}$ ), or even a combination that expresses directly the power split, for example the ratio of capacitor power to total electric power:

$\frac{P_{cap}}{P_{gen,e}+P_{cap}}$ . The capacitor power  $P_{cap}$  is chosen in this case, since it allows for an easier expression of the system dynamics equation, as shown in Section 4.2.2. The generator electrical power can be expressed as function of the capacitor power using (4.9):

$$P_{gen,e}(t) = P_{em1,e}(t) + P_{em2,e}(t) - P_{cap}(t) \quad (4.11)$$

## 4.2.2 System dynamics

As mentioned earlier, the only dynamic state that is relevant for the energy management problem is the state of charge of the energy storage device, which has slow dynamics compared to the response of all the powertrain actuators.

The longitudinal inertial dynamics of the vehicle are characterized by a comparable time constant, but vehicle speed is not a state in the energy management problem, rather a measured input, because the speed controller (which in a real vehicle is the driver) is independent of the energy management controller. The energy management controller does not decide how much power is to be delivered to the wheels, but only how to apportion the power request that comes from the speed controller.

The state of charge of an electrical energy storage device is the amount of electrical charge stored in it, with respect to the total charge capacity; it is directly proportional to the integral of the current flowing into and out of the storage device. The state of energy is the amount of energy stored, relative to the maximum energy capacity, and is related to the integral of the power. Since the problem is being set up using power balance equations, it is more natural to use the state of energy of the capacitors as the system state.

The system dynamic equation expresses the variation of state of energy; by definition, the state of energy of a capacitor is

$$\zeta(t) = \frac{\frac{1}{2}CV_C^2(t)}{\frac{1}{2}CV_{C,max}^2} \quad (4.12)$$

and its time derivative is therefore

Table 4.1: Supercapacitor characteristics

Number of cells in series	10
Number of strings in parallel	1
Nominal cell voltage	56 V
Nominal pack voltage, $V_{cap,nom}$	560 V
Capacitance of each cell	110 F
Resistance of each cell	0.04 $\Omega$
Equivalent capacitance, $C$	11 F
Equivalent resistance, $R$	0.4 $\Omega$
Peak current (discharge), $I_{cap,max}$	500 A
Peak current (charge), $I_{cap,min}$	-500 A
Maximum state of energy, $\zeta_{max}$	0.95
Minimum state of energy, $\zeta_{min}$	0.25
Total capacitor energy, $E_{cap}$	1.9 MJ

$$\dot{\zeta}(t) = \frac{1}{E_{cap}} \frac{1}{2} C \cdot 2V_C(t) \dot{V}_C(t) \quad (4.13)$$

where  $C$  is the equivalent capacitance of the supercapacitor pack (which is composed by 10 modules in series, each of which with the characteristics shown in Table 4.1),  $V_C$  the voltage across the equivalent capacitance, and  $E_{cap} = \frac{1}{2} C V_{C,max}^2$  is the maximum amount of energy stored in the capacitors. The voltage derivative is  $\dot{V}_C(t) = -\frac{1}{C} I(t)$  where  $I(t)$  is the current (the minus sign is due to the convention of positive current during discharge); thus, the expression of  $\dot{\zeta}(t)$  becomes

$$\dot{\zeta}(t) = -\frac{1}{E_{cap}} V_C(t) I(t) \quad (4.14)$$

If the capacitor could be represented as a pure capacitance (i.e., with no losses), the quantity  $V_C(t)I(t)$  would equal the power at the terminals  $P_{cap}(t)$ ; instead, because of the presence of losses in the circuit,  $V_C(t)I(t)$  is the power that charges or discharges the capacitance, but it is different than the power at the terminals.

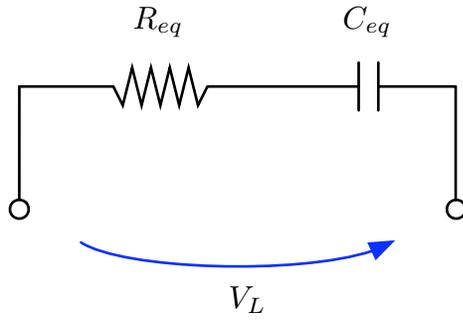


Figure 4.2: Circuit model of supercapacitor pack.

Using the very simple circuit model of Figure 4.2, the power at the terminals is defined as

$$P_{cap}(t) = V_L(t)I(t) \quad (4.15)$$

where  $V_L(t)$  is the voltage at the terminals, which is different from the capacitance voltage because of the losses represented by the equivalent resistance  $R$ :  $V_L(t) = V_C(t) + RI(t)$ . Thus,

$$V_C I = V_L I - RI^2 = P_{cap} - RI^2 \quad (4.16)$$

and the analytical formulation of the state of energy equation is then written as

$$\dot{\zeta}(t) = -\frac{1}{E_{cap}} \left( P_{cap}(t) + RI^2(t) \right) \quad (4.17)$$

or

$$\dot{\zeta}(t) = -\frac{1}{E_{cap}} P_{cap}(t) \left( 1 + \frac{RI^2(t)}{P_{cap}(t)} \right) \quad (4.18)$$

The factor

$$\varepsilon_{cap} = 1 + \frac{RI^2(t)}{P_{cap}(t)} \quad (4.19)$$

in (4.18) represents essentially an efficiency, since it is used to express the “net” power charging or discharging the capacitors in terms of the power at the terminals  $P_{cap}$ . However,  $1 + \frac{RI^2(t)}{P_{cap}(t)}$  can take values either lower or greater than 1, depending

on the sign of the power. Thus, the factor  $\varepsilon_{cap}$  can be defined “charge-effectiveness” and interpreted as

$$\varepsilon_{cap}(\zeta, P_{cap}) = 1 + \frac{R(\zeta, P_{cap}) I^2(t)}{P_{cap}(t)} = \begin{cases} \frac{1}{\eta_{cap}(x, P_{cap})} & \text{if } P_{cap}(t) \geq 0 \text{ (discharge)} \\ \eta_{cap}(x, P_{cap}) & \text{if } P_{cap}(t) < 0 \text{ (charge)} \end{cases} \quad (4.20)$$

where  $\eta_{cap}$  is the charge/discharge efficiency. The resistance  $R$ , and hence the efficiency, depend on both the state of energy and the power:  $\eta_{cap} = \eta_{cap}(\zeta, P_{cap})$ . The state of energy dynamics can then be described as

$$\dot{\zeta}(t) = \begin{cases} -\frac{1}{\eta_{cap}} \frac{P_{cap}(t)}{E_{cap, max}} & \text{if } P_{cap}(t) \geq 0 \text{ (discharge)} \\ -\eta_{cap} \frac{P_{cap}(t)}{E_{cap, max}} & \text{if } P_{cap}(t) < 0 \text{ (charge)} \end{cases} \quad (4.21)$$

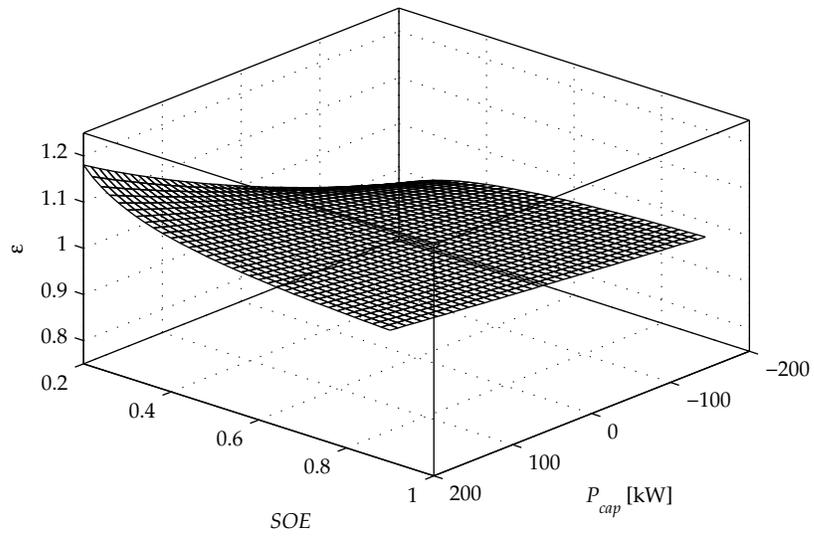
However, in order to simplify the notation and not dealing with equations that change expression depending on the sign of the variables involved, the expression with  $\varepsilon_{cap}$  is preferred:

$$\boxed{\dot{\zeta}(t) = -\varepsilon_{cap}(\zeta, P_{cap}) \frac{P_{cap}(t)}{E_{cap}}} \quad (4.22)$$

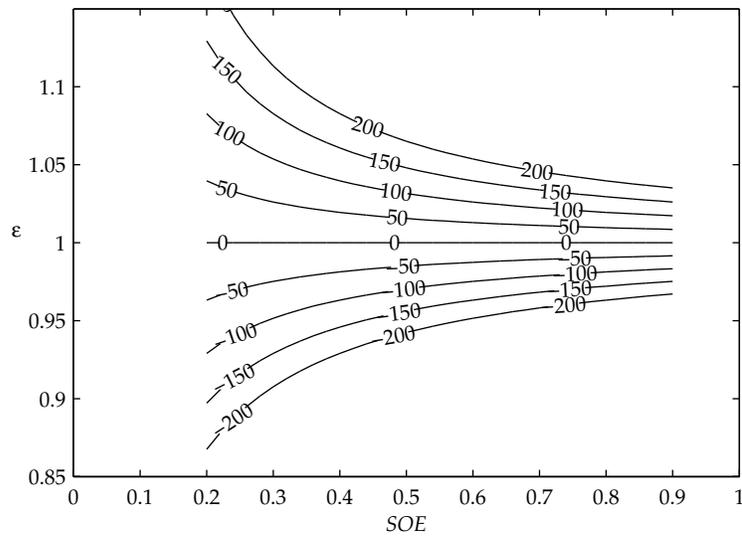
Note that the expression of  $\varepsilon_{cap}$  given here was derived using an equivalent circuit model of the capacitors; however, in the general case,  $\varepsilon_{cap}$  can always be defined as the ratio of the state of energy variation and the power at the terminals:

$$\varepsilon_{cap} \doteq \frac{\dot{\zeta}(t) E_{cap}}{P_{cap}(t)} \quad (4.23)$$

and can be measured experimentally or computed from a model of any kind, given the observation of the variables  $\dot{\zeta}(t)$  and  $P_{cap}(t)$ . Depending on the quality and accuracy of the model, the factor  $\varepsilon_{cap}$  can be a function of several variables. In the circuit models typically used for energy analysis, all parameters are a function of current intensity, voltage, and temperature  $\theta$ . Current and voltage are related to power and state of charge (or state of energy). As a consequence,  $\varepsilon_{cap}$  is a function of the same variables:  $\varepsilon_{cap} = \varepsilon_{cap}(\zeta, P_{cap}, \theta)$ . Using the capacitor model described in Section 2.4.9, this coefficient can be computed for the entire range of state of energy and power as shown in Figure 4.3.



(a) 3-d view of  $\epsilon (SOE, P_{cap})$



(b) Curves  $\epsilon (SOE)$  parameterized in  $P_{cap}$  (values in kW)

Figure 4.3: Charge-effectiveness factor for the ultracapacitor pack of the refuse truck

### 4.2.3 State constraints

The state constraints express the fact that the state of energy should remain within an acceptable range during the trip:

$$\zeta_{min} \leq \zeta(t) \leq \zeta_{max} \quad \forall t \in [t_0, t_f] \quad (4.24)$$

This chain of inequalities can be rewritten using the notation introduced in Section 4.1 as

$$G(\zeta) \leq 0, \quad G(\zeta) = \begin{cases} G_1(\zeta) = \zeta_{min} - \zeta \\ G_2(\zeta) = \zeta - \zeta_{max} \end{cases} \quad (4.25)$$

The maximum and minimum value of the state of energy are constant during the operation and are set to  $\zeta_{max} = 0.95$ ,  $\zeta_{min} = 0.25$ . In terms of state of charge  $\xi$ , recalling the relation  $\zeta = \xi^2$  (§2.4.9), these values correspond to  $\xi_{max} = 0.97$  and  $\xi_{min} = 0.5$ . The operational range of supercapacitors is higher than the typical values for batteries, thanks to the fact that supercapacitors are much more tolerant than batteries of very low and very high state of charge, and can be charged and discharged to the limits without problems of wear or aging. The reason for limiting the lower SOE value to 0.25 is to keep the minimum SOC to 0.5, i.e. not reducing the voltage below 50% of the nominal value, since otherwise the amount of total power deliverable would be too small.

The terminal constraints express the fact that the initial and final state of energy should be close (ideally equal) in a charge-sustaining vehicle. However, in the case of supercapacitors, the amount of energy that is stored is rather small, and negligible with respect to the total energy used during any driving cycle. Therefore, it is not necessary to penalize solutions that bring the state of energy to extreme values and no terminal constraints are posed on the system state at the final time, nor is a terminal cost defined.

### 4.2.4 Control constraints

The control variable for this problem is the capacitor power  $P_{cap}(t)$ . The constraints on the control variable express the physical limitations of the capacitors

and of the engine. The engine speed must be, at each time, within the idle and red-line values:

$$\omega_{ice,min} \leq \omega_{ice}(t) \leq \omega_{ice,max} \quad (4.26)$$

The capacitor power must be within a minimum and a maximum value that depend on the current power request and the limits of the generator. Keeping in mind the power flow diagram and sign conventions of Figure 4.1, the following chain of inequalities can be written:

$$-P_{gen,e,max} + P_{req}(t) \leq P_{cap}(t) \leq P_{gen,e,min} - P_{req}(t) \quad (4.27)$$

which derives from the following considerations:

- given a total power request  $P_{req}(t) = P_{em1,e}(t) + P_{em2,e}(t)$ , this is equal to the sum of capacitor and generator power;
- the capacitor power flowing into the traction machines cannot exceed their power request (this defines the right inequality);
- the capacitors can absorb negative power, up to the sum of generator power and negative loads (this defines the left inequality).

At the same time, the physical limitations of the capacitors must be taken into account:

$$P_{cap,min}(\zeta(t)) \leq P_{cap}(t) \leq P_{cap,max}(\zeta(t)) \quad (4.28)$$

where  $P_{cap,min}(\zeta)$  and  $P_{cap,max}(\zeta)$  are, respectively, the minimum and maximum capacitor power; they depend on the state of energy  $\zeta(t)$  because the power output is  $P_{cap} = I_{cap}V_{cap}$  and the state of energy is, by definition, proportional to the squared voltage:  $\zeta = \frac{1/2CV_{cap}^2(t)}{1/2CV_{cap,nom}^2}$ . The maximum and minimum values of capacitor power are given by

$$P_{cap,max}(t) = I_{cap,max}V_{cap}(t) = I_{cap,max}\sqrt{\zeta(t)}V_{cap,nom} \quad (4.29)$$

and

$$P_{cap,min}(t) = I_{cap,min}V_{cap}(t) = I_{cap,max}\sqrt{\zeta(t)}V_{cap,nom} \quad (4.30)$$

The maximum and minimum values of current (respectively during discharge and recharge) are listed in Table 4.1.

The control constraints can therefore be expressed as  $P_{cap}(t) \in [P_{cap,inf}(t), P_{cap,sup}(t)]$ , where:

$$\begin{aligned} P_{cap,inf}(t) &= \max \{ P_{req}(t) - P_{gen,e,max}, P_{cap,min}(x(t)) \} \\ P_{cap,sup}(t) &= \min \{ P_{gen,e,min} - P_{req}(t), P_{cap,max}(x(t)) \} \end{aligned} \quad (4.31)$$

#### 4.2.5 Fuel consumption

The objective of the control is to minimize fuel consumption; therefore, the instantaneous cost  $L(x, u, t)$  is given by the instantaneous fuel consumption of the engine  $\dot{m}_f$ . In order to formalize the control problem as in Section 4.1, the fuel consumption must be expressed as a function of the system states and control variables.

Fuel consumption is in general a function of engine torque and speed:  $\dot{m}_f = \dot{m}_f(T_{ice}, \omega_{ice})$ . In a series hybrid architecture, the engine speed is not related to the vehicle speed, since there is no mechanical connection between the engine and the wheels: therefore,  $\omega_{ice}$  can be considered a free variable, i.e. one of the control inputs. The engine torque is related to speed and power by the relation  $T_{ice} = P_{ice}/\omega_{ice}$ , which allows to redefine the fuel consumption as a function of engine power and speed:  $\dot{m}_f = \dot{m}_f(P_{ice}, \omega_{ice})$ . This function is tabulated using the fuel consumption map, and therefore it does not have an analytical expression, but it can be curve-fitted with a second-order polynomial function [37], or an affine relation with speed-dependent coefficients [62, 35, 39]. The engine power appearing in the expression of the fuel consumption is given by (4.8). Therefore, the instantaneous cost  $L(x, u, t)$  is expressed as a function of the control variable  $P_{cap}$ , the engine speed  $\omega_{ice}$ , and of the time-varying parameters

$v(t) = \{P_{pto}(t), P_{acc}(t), P_{em1,e}(t), P_{em2,e}(t)\}$ .  $L(x, u, t)$  is, in fact, not a function of the system state  $x = SOE$ .

In some cases, rather than using numerical map interpolation, it can be useful or necessary to curve-fit the characteristic maps of the engine and the electric machines, and express the fuel consumption as an analytical function of the variables involved. A physics-based approach to do this is the Willans line model introduced in Section 2.4.1: this model expresses the power entering a machine,  $P_{in}$ , as an affine function of the net (output) power  $P_{out}$ :

$$P_{in} = a_0(\omega) + a_1(\omega) P_{out} \quad (4.32)$$

This curve fit can be interpreted in physical terms:  $a_0(\omega)$  represents the friction losses, while  $a_1(\omega)$  represents the conversion efficiency of the machine. These coefficients may be fitted with good results with a linear or function of the machine speed; therefore, in general, they are expressed as:

$$\begin{cases} a_0(\omega) = a_{00} + a_{01}\omega + a_{02}\omega^2 \\ a_1(\omega) = a_{10} + a_{11}\omega + a_{12}\omega^2 \end{cases} \quad (4.33)$$

The meaning of  $P_{in}$  and  $P_{out}$  depends on the machine. For an internal combustion engine,  $P_{out}$  is the mechanical power, while  $P_{in}$  is the fuel power, i.e. the power  $P_{fuel} = \dot{m}_f Q_{lhv}$  associated with the fuel flow entering the engine. For an electric machine working as a motor,  $P_{in}$  is the electrical power while  $P_{out}$  is the mechanical power; the role may be switched if the machine works as a generator. For a reversible machine, the same equation can be used for both motoring and generation, but the coefficients for the two cases are different.

In this application, Willans line modeling is applied to the generator and engine.

For the generator, it was found that good results (shown in Figure 4.4) are obtained with the simplified form:

$$P_{gen,e} = g_{00} + g_{10} P_{gen,m} \quad (4.34)$$

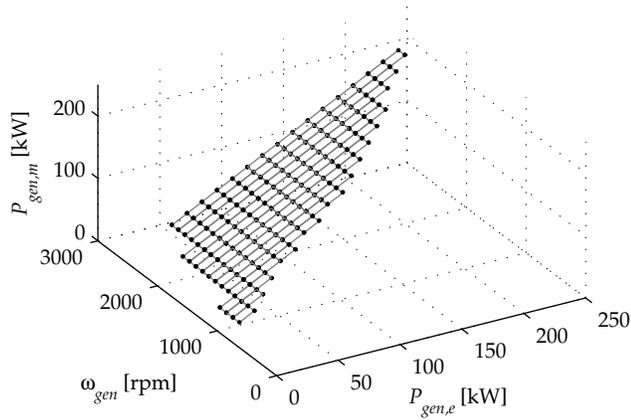


Figure 4.4: Willans line fit of the generator map (constant  $a_0, a_1$ )

i.e. neglecting the effect of speed on efficiency. The engine map, instead, is fitted with the standard form

$$P_{fuel} = Q_{lhv} \dot{m}_f = \left( e_{00} + e_{01} \omega_{ice} + e_{02} \omega_{ice}^2 \right) + \left( e_{10} + e_{11} \omega_{ice} + e_{12} \omega_{ice}^2 \right) P_{ice} \quad (4.35)$$

and the corresponding results are shown in Figure 4.5 and 4.6. The two figures show different representation of the data (power and speed, or torque, speed and fuel consumption); in each figure, the complete expression (4.35) is compared with the simplified case in which  $e_{01}$ ,  $e_{02}$ ,  $e_{11}$  and  $e_{12}$  are zero (i.e. speed dependency is neglected), showing that the engine speed does have an effect on the efficiency, even if relatively small. Eqs. (4.34) and (4.35) are capable of fitting the generator and engine map with good accuracy, thus providing a realistic model of the machines (at least in terms of steady-state efficiency).

Applying this modeling approach to the generator and engine of the series hybrid configuration allows to write the fuel consumption as an analytical function of the capacitor and load power. Based on the fitting results shown, the generator map can be represented using constant coefficients, while the engine map is fitted taking into account the speed dependency. Eq. (4.9) can be written using the Willans fit (4.34) as:

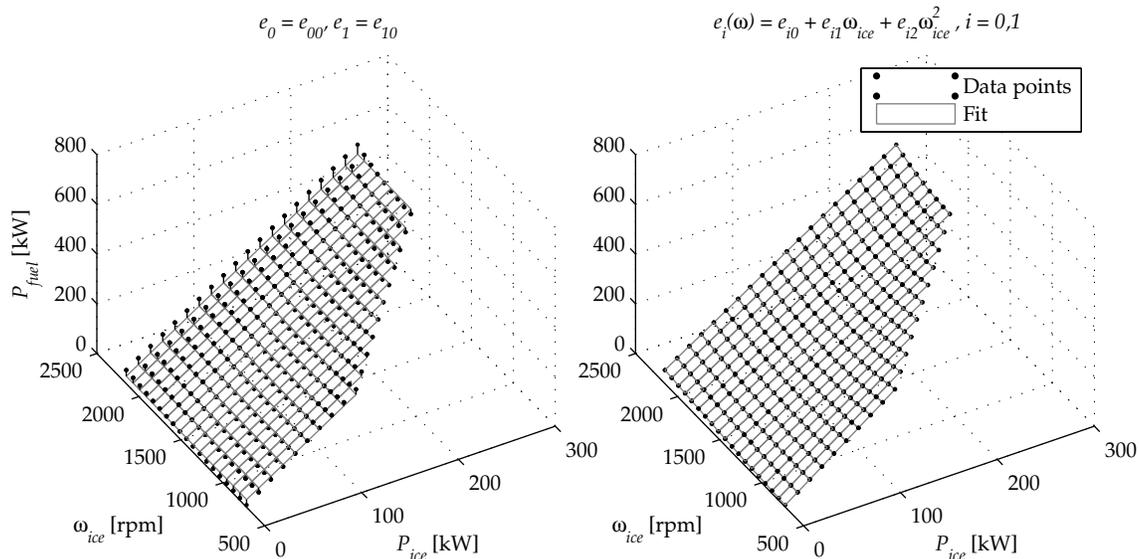


Figure 4.5: Willans line fit of the engine map in the power-speed plane (left figure: speed dependency of the maps is neglected; right: speed dependency is considered)

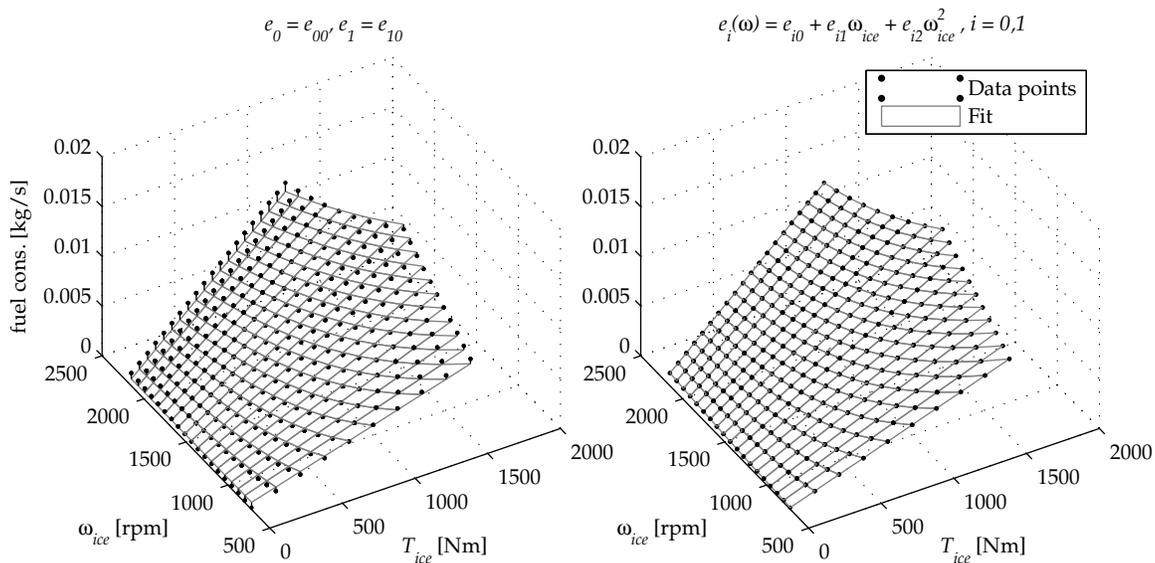


Figure 4.6: Willans line fit of the engine map in the torque-speed plane (left figure: speed dependency of the maps is neglected; right: speed dependency is considered)

$$P_{gen,m} = g_{00} + g_{10}P_{gen,e} = g_{00} + g_{10} (P_{em1,e} + P_{em2,e} - P_{cap}) \quad (4.36)$$

and the engine fuel consumption (4.35) can be expressed by using (4.8) as

$$P_{fuel} = \left( e_{00} + e_{01}\omega_{ice} + e_{02}\omega_{ice}^2 \right) + \left( e_{10} + e_{11}\omega_{ice} + e_{12}\omega_{ice}^2 \right) (P_{pto} + P_{acc} + P_{gen,m}). \quad (4.37)$$

Combining these equations, the fuel power is then expressed as a function of the capacitor power as follows:

$$P_{fuel} = e_{00} + e_{01}\omega_{ice} + e_{02}\omega_{ice}^2 + \left( e_{10} + e_{11}\omega_{ice} + e_{12}\omega_{ice}^2 \right) [P_{pto} + P_{acc} + g_{00} + g_{10} (P_{em1,e} + P_{em2,e} - P_{cap})] \quad (4.38)$$

and the corresponding fuel consumption is simply a rescaling of this quantity:

$$\dot{m}_f (\omega_{ice}(t), P_{cap}(t), t) = \frac{1}{Q_{lhv}} P_{fuel}. \quad (4.39)$$

Despite the presence of two control variables, this power split problem only has one degree of freedom, because the choice of engine speed does not affect the system state and therefore can be regarded as a local minimization problem. In other words, it is possible to pre-determine the speed that corresponds to the lowest fuel consumption at a given engine power level, and then assume that the engine operates along this minimum fuel consumption line. Using the curve fit presented here, the optimal engine speed is the one that satisfies the conditions

$$\frac{\partial P_{fuel}}{\partial \omega_{ice}} = e_{01} + 2e_{02}\omega_{ice} + e_{11}P_{ice} + 2e_{12}\omega_{ice}P_{ice} = 0 \quad (4.40)$$

$$\frac{\partial^2 P_{fuel}}{\partial \omega_{ice}^2} = 2e_{02} + 2e_{01}P_{ice} > 0. \quad (4.41)$$

The condition on the second derivative is, in fact, a condition on the engine power  $P_{ice}$ , which must be such that  $P_{ice} > -\frac{e_{02}}{e_{01}}$  for the function  $P_{fuel}(\omega_{ice})$  to have a minimum along the stationary curve  $\frac{\partial P_{fuel}}{\partial \omega_{ice}} = 0$ . The condition on  $P_{ice}$  is always

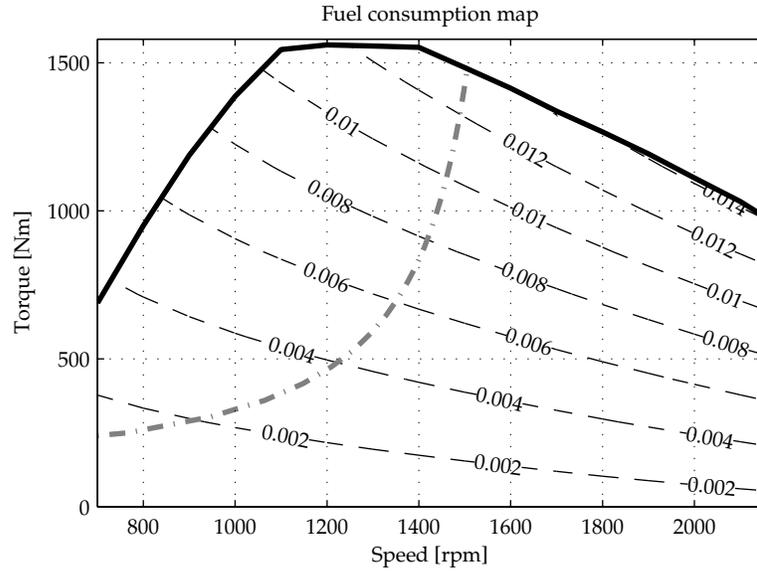


Figure 4.7: Engine fuel consumption map with optimal operating line

satisfied by positive engine power, given the values of the fitting coefficients listed in Table 4.2. The optimal speed is thus

$$\omega_{ice,opt} = -\frac{1}{2} \frac{e_{01} + e_{11}P_{ice}}{e_{02} + e_{12}P_{ice}} \quad (4.42)$$

where

$$P_{ice} = P_{pto} + P_{acc} + g_{00} + g_{10} (P_{em1,e} + P_{em2,e} - P_{cap}). \quad (4.43)$$

The optimal engine speed can be plotted on the engine map as shown in Figure 4.7. Superimposing constant power contours to the engine efficiency maps it is clear the fact that the optimal speed corresponds to the points in which the power contours are tangent to the efficiency lines, i.e. the points of highest efficiency for a given power level.

The fuel consumption corresponding to the optimal speed is shown in Figure 4.9; assuming operation in this condition at any time, the fuel consumption is a

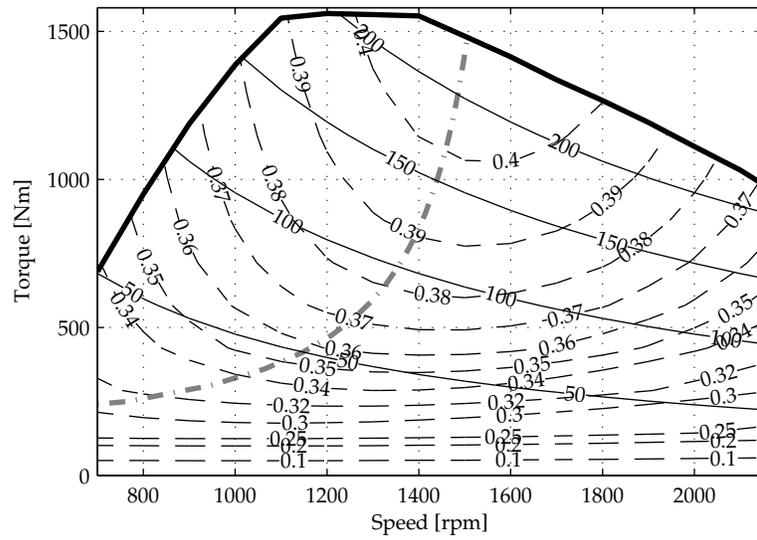


Figure 4.8: Engine efficiency map with optimal operating line and iso-power lines

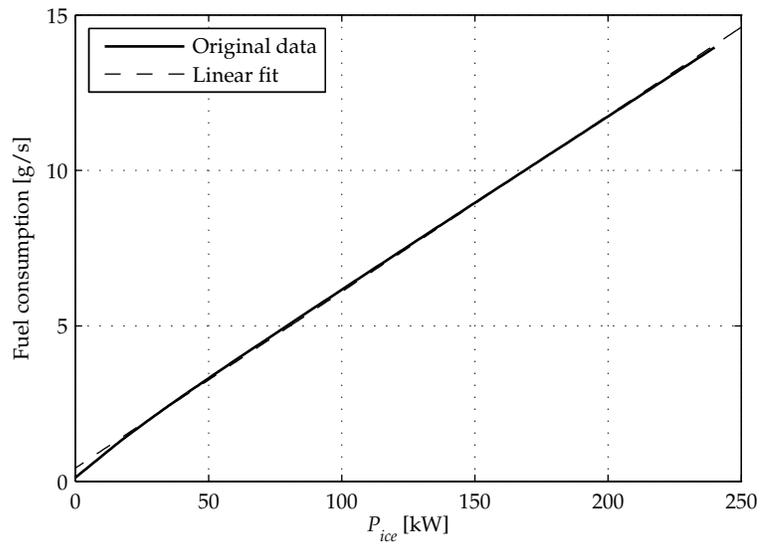


Figure 4.9: Engine fuel consumption along maximum efficiency line

function only of the engine power, and – in this particular case – can be fitted using an affine equation:

$$\dot{m}_f (P_{ice}) = m_0 + m_1 P_{ice}, \quad (4.44)$$

that is, using the explicit expression for  $P_{ice}$ :

$$\boxed{\dot{m}_f = m_0 + m_1 [P_{pto} + P_{acc} + g_{00} + g_{10} (P_{em1,e} + P_{em2,e} - P_{cap})]} \quad (4.45)$$

The coefficients present in this equation are shown in Table 4.3 and Table 4.4, while the engine parameters are shown in Table 4.2.

The engine speed calculated by the energy management strategy is actually a reference value fed to the engine speed controller, but for the assumptions made in the formulation of the energy management problem it can be assumed that the reference speed and the actual speed are coincident (i.e., the transients involved in engine speed tracking are faster than the state of energy transients).

A detail that can be noticed by someone familiar with series hybrid vehicles is the fact that the optimal efficiency line does not depend on the generator characteristics, but only on the engine map. This is not true as a general rule, as the two machines are considered together in the definition of an overall fuel consumption map that depends on the net electrical power delivered by the generator. However, in this case, the curve fitting with Willans line models is such that the generator efficiency does not depend on its speed, only on the power; thus, the effect of varying speed only affects the engine map (this is analytically seen in (4.40)). This fact makes the formulation of fuel consumption (4.60) much less cumbersome than in the general case, in which the engine and generator map should be combined considering the presence of accessory and PTO load between the two machines. In fact, in that case, the fuel consumption would be more easily computed using look-up tables rather than curve fitting.

Eq. (4.60) expresses the fuel consumption as a function of the control variable  $P_{cap}$ , assuming that the engine speed is set to the value defined by (4.42), and that the time-varying parameters  $P_{pto}$ ,  $P_{acc}$ ,  $P_{em1,e}$  and  $P_{em2,e}$  are known.

Table 4.2: Fitting coefficients for the Willans line model of the engine (4.39) (power in kW, speed in rad/s)

Coefficient	Value
$e_{00}$	-12.61
$e_{01}$	34.14
$e_{02}$	0.93
$e_{10}$	3.79
$e_{11}$	-0.017
$e_{12}$	$5 \cdot 10^{-5}$

Table 4.3: Fitting coefficients for the Willans line model of the generator

Coefficient	Value
$g_{00}$	3537
$g_{01}$	0
$g_{02}$	0
$g_{10}$	1.0318
$g_{11}$	0
$g_{12}$	0

Table 4.4: Fitting coefficients for the engine fuel consumption along maximum efficiency line ( $\dot{m}_f = m_0 + m_1 P_{ice}$ , with  $P_{ice}$  in W and  $\dot{m}_f$  in g/s)

Coefficient	Value
$m_0$	0.43
$m_1$	$5.7 \cdot 10^{-5}$

### 4.3 Definition of the control problem for case study 2

The same considerations presented in the previous section for the refuse truck are reported here to the EcoCAR vehicle. The architecture of the powertrain is the same, but some differences in the intended vehicle use require a slightly different problem formulation.

The state variable is the state of energy  $\zeta$  of the energy storage device, in this case Li-Ion batteries:

$$x = \{\zeta(t)\} \quad (4.46)$$

The control variable is the battery electric power:

$$u = \{P_{batt}\} \quad (4.47)$$

For the definition of the cost function, one should take into account the fact that this is a plug-in hybrid vehicle, in which the battery can be discharged completely during a trip and then recharged using electricity from the electric grid. In fact, a plug-in hybrid vehicle can operate in two modes: charge-depleting and charge-sustaining. If the battery is charged at the beginning of the trip, the vehicle starts in charge-depleting mode allowing battery discharge. When the SOC reaches a low threshold, the vehicle enters the charge-sustaining mode, in which the behavior is the same as a traditional hybrid vehicle: the battery state of charge should remain substantially equal between the beginning and the end of the charge-sustaining portion of the trip.

The cost function during the charge-sustaining operation is the same as in the first case study:

$$J_{cs} = \int_{t_0}^{t_f} m_f(\zeta, P_{batt}, t) dt \quad (4.48)$$

and accounts for fuel consumption only.

In charge-depleting operation, the cost function can be defined differently than for a charge-sustaining vehicle, in order to account for the electricity introduced

from an external source as well as for the fuel consumption. The definition of meaningful optimization objectives for plug-in hybrid vehicles is currently object of active research [86, 87] and reflects the complexity of a problem of larger scale than in-vehicle optimization. Cost functions used as optimization objectives in plug-in hybrid vehicles include fuel consumption (neglecting the electricity cost), fuel consumption and emissions, total energy used (fuel and electricity), total cost of consumed energy, total cost of consumed energy and battery wear, etc. In this case, it is assumed that the optimization objective is to minimize the cost (in dollars) of a trip, accounting only for the consumption of fuel and electrical energy during the trip. The terminal cost  $\phi(x(t_f), t_f)$  is used to account for the cost of the electricity and is proportional to the battery discharge:

$$\phi'(x(t_f), t_f) = (\zeta_{max} - \zeta(t_f)) E_{batt} C_{elec} \quad (4.49)$$

where  $\zeta_{max}$  is the maximum SOE level reached after recharge from an outlet (typically very close to 1),  $\zeta(t_f)$  is the value of SOE at the end of the optimization horizon,  $E_{batt}$  is the amount of energy that the battery can hold when it is fully charged,  $C_{elec}$  is the cost of the electricity in dollars per unit of energy (published rates are typically in \$/kWh). This definition of the terminal cost is coupled to the definition of the instantaneous cost  $L(x, u, t)$  as the cost (in dollars) of engine fuel consumption:

$$L(x, u, t) = C_{fuel} Q_{lhv} \dot{m}_f(\zeta, P_{batt}, t) \quad (4.50)$$

where  $C_{fuel}$  is the cost of the fuel in dollars per unit of energy (just like  $C_{elec}$ ) and  $Q_{lhv}$  the fuel energy content per unit of mass.

The resulting cost function is defined as

$$J' = (\zeta_{max} - \zeta(t_f)) E_{batt, max} C_{elec} + C_{fuel} Q_{lhv} \int_{t_0}^{t_f} \dot{m}_f(\zeta, P_{batt}, t) dt \quad (4.51)$$

and is an actual cost, measured in dollars. In order to express this definition in a form identical to the charge-sustaining case, the cost can be rewritten as follows:

$$J_{cd} = (\zeta_{max} - \zeta(t_f)) \frac{E_{batt,max}}{Q_{lhv}} \frac{C_{elec}}{C_{fuel}} + \int \dot{m}_f(\zeta, P_{batt}, t) dt \quad (4.52)$$

where the terminal cost is now

$$\phi(x(t_f), t_f) = (\zeta_{max} - \zeta(t_f)) \frac{E_{batt,max}}{Q_{lhv}} \frac{C_{elec}}{C_{fuel}}. \quad (4.53)$$

Note that the terminal cost has unit of mass, i.e. mass of fuel that is equivalent (in terms of monetary cost) to the battery discharge.

The constraints on the final value of the state  $\psi(\zeta(t_f), t_f)$  are not defined: the final state will assume a value dictated by the terminal cost.

The dynamic constraints are exactly the same as those defined for the previous case study, and indicate that the state of energy and the control variable (battery power) must remain within the allowable range:  $\zeta_{min} \leq x(t) \leq \zeta_{max}$  and  $P_{batt,min}(t) \leq P_{batt}(t) \leq P_{batt,max}(t)$ .

### 4.3.1 Power flow diagram

The powertrain architecture of the vehicle is shown in Figure 4.10, while the corresponding power flow diagram is in Figure 4.11. The architecture is very similar to the one analyzed for the first case study: the main difference is the lack of mechanically-powered accessories, since there is no need for PTO in this vehicle and the accessories (air conditioning, power steering etc.) are electrically powered from the main bus. However, the envisioned mode of functioning is different because the power of the engine-generator set is smaller than the power installed at the traction motor; therefore, the generator can provide the sustained power rating, while the peak power can be reached using the battery pack (even without assistance from the generator).

The electric power request that the generator and the battery must satisfy together is

$$P_{req} = P_{em,e} + P_{acc} \quad (4.54)$$

and must be satisfied using either the generator or the battery:

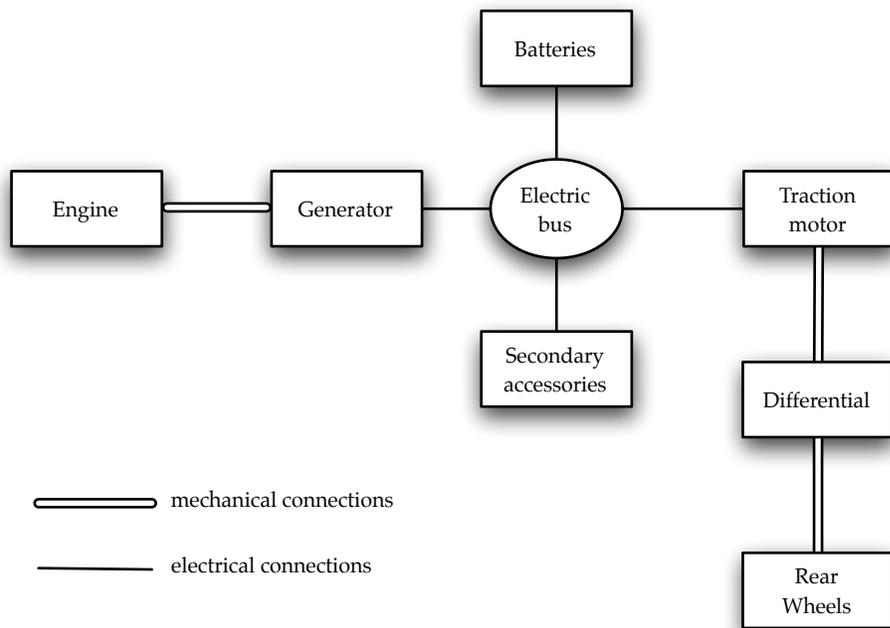


Figure 4.10: Powertrain architecture for case study 2

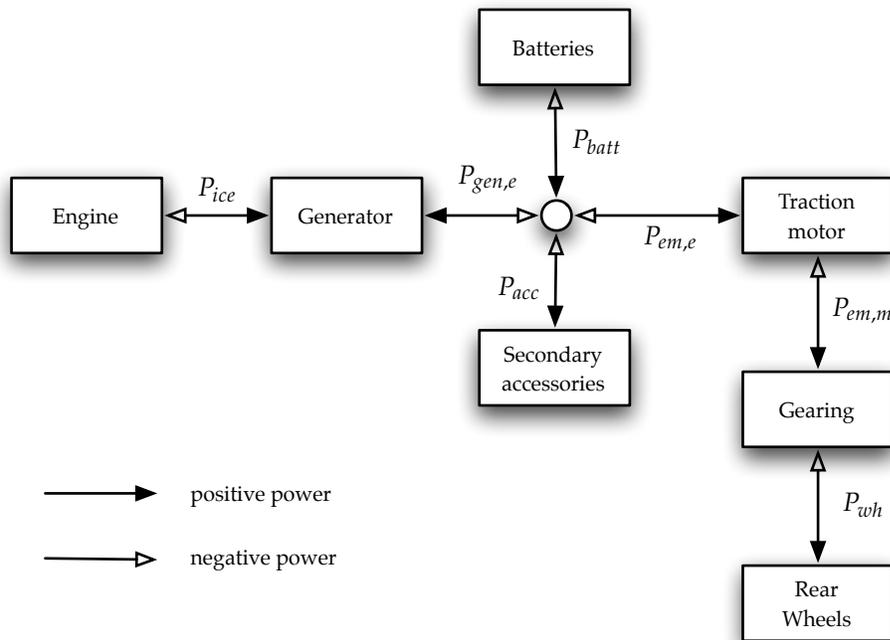


Figure 4.11: Power flow diagram for case study 2

$$P_{req} = P_{gen,e} + P_{batt} \quad (4.55)$$

### 4.3.2 System dynamics

Just as in the previous case, the system dynamic equation is provided by the battery state of energy dynamics

$$\dot{\zeta}(t) = -\varepsilon_{batt}(\zeta, P_{batt}) \frac{P_{batt}(t)}{E_{batt}} \quad (4.56)$$

where the factor  $\varepsilon_{batt}$  represents the ratio between the battery power  $P_{batt}$  and the variation of state of energy  $\dot{\zeta}$ , and is tabulated using a battery model that implements the equations of Section 2.4.8; in particular,

$$\varepsilon_{batt} = 1 + \frac{RI^2(t)}{P_{batt}(t)} \quad (4.57)$$

where  $R$  is the total resistance of the equivalent circuit and  $I$  the current. The function  $\varepsilon_{batt}(\zeta, P_{batt})$  is shown in Figure 4.12. The dependency on the state of energy is negligible, due to two effects: the fact that the internal resistance does not depend on the SOE (due to lack of detailed modeling information, an average value is considered) and the fact that the open circuit voltage variation with SOE<sup>1</sup> is rather small, as shown in Figure 4.13.

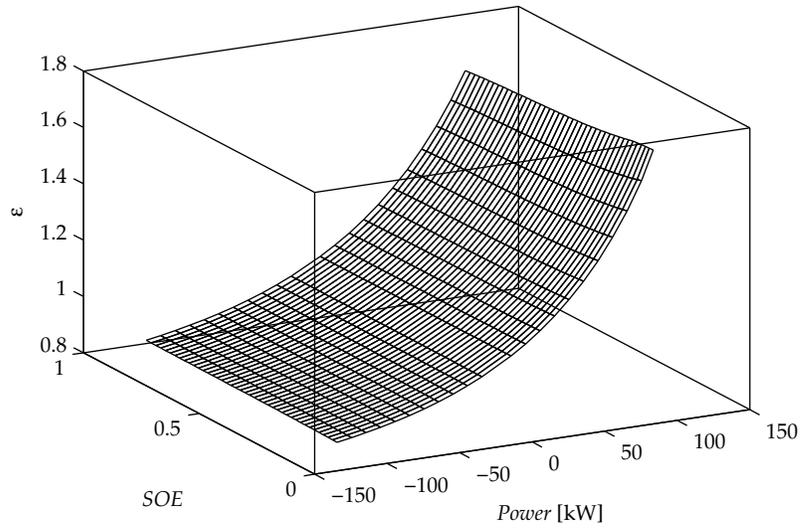
### 4.3.3 State constraints

The constraints imposed on the state values are expressed as

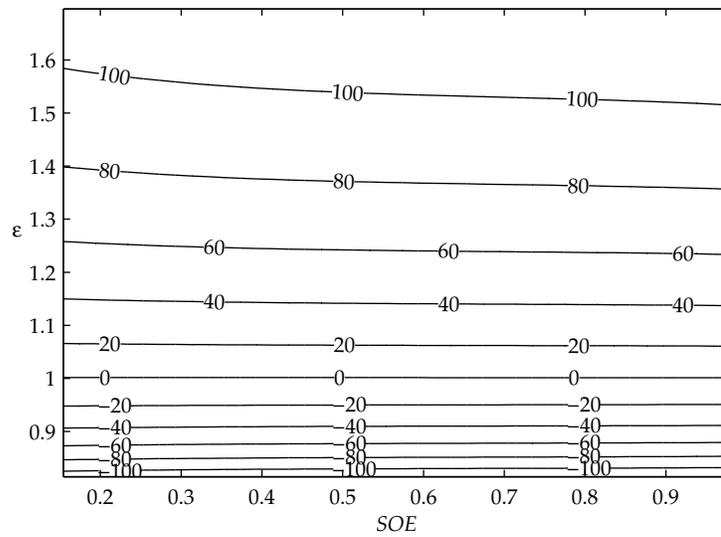
$$G(\zeta) \leq 0, \quad G(\zeta) = \begin{cases} G_1(\zeta) = \zeta_{min} - \zeta \\ G_2(\zeta) = \zeta - \zeta_{max} \end{cases} \quad (4.58)$$

which is identical to (4.58). The extreme values of the state of energy in this case are  $\zeta_{max} = 0.95$  and  $\zeta_{min} = 0.2$ , because of the charge-depleting nature of the vehicle.

<sup>1</sup>The curve of Figure 4.13 is the standard open circuit voltage vs. state of charge characteristic; because of the almost negligible voltage variation with SOC, the values of state of energy and state of charge are in fact very close to each other. Thus, the x-axis of the plot can be interpreted as SOE, with very good approximation.



(a) 3-d view of  $\epsilon(SOE, P_{batt})$



(b) Curves  $\epsilon(SOE)$  parameterized in  $P_{batt}$  (values in kW)

Figure 4.12: Charge-effectiveness factor for the battery pack of the EcoCAR

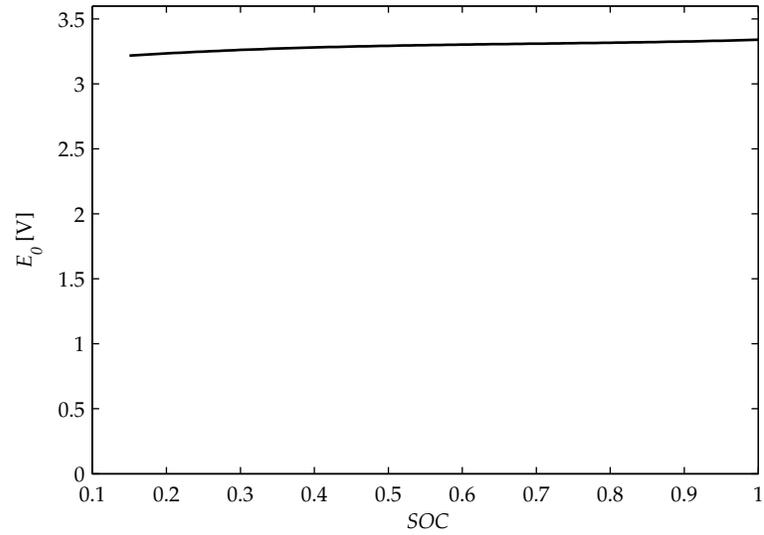


Figure 4.13: Open circuit voltage of the battery as a function of state of charge (data referred to a single cell)

Table 4.5: EcoCAR battery characteristics

Number of cells in series	110
Number of strings in parallel	1
Nominal cell voltage	3.3 V
Nominal pack voltage, $V_{cap,nom}$	330 V
Capacity of each cell	19.6 Ah
Resistance of each cell	0.01 $\Omega$
Peak current (discharge), $I_{batt,max}$	150 A
Peak current (charge), $I_{batt,min}$	-120 A
Maximum state of energy, $\zeta_{max}$	0.9
Minimum state of energy, $\zeta_{min}$	0.3
Total battery energy, $E_{batt}$	24.8 MJ

#### 4.3.4 Control constraints

Using the same considerations reported in Section (4.2.4) for the first case study, the control constraints can be expressed as  $P_{batt}(t) \in [P_{batt,inf}(t), P_{batt,sup}(t)]$ , with

$$\begin{aligned} P_{batt,inf}(t) &= \max \{ P_{req}(t) - P_{gen,e,max}, P_{batt,min}(x(t)) \} \\ P_{batt,sup}(t) &= \min \{ P_{gen,e,min} - P_{req}(t), P_{batt,max}(x(t)) \} \end{aligned} \quad (4.59)$$

$P_{batt,min}$  and  $P_{batt,max}$  being the physical limitations of the battery, while  $P_{batt,inf}(t)$  and  $P_{batt,sup}(t)$  are the effective limitations, taking into account the power  $P_{req}(t)$  that needs to be delivered to or received from the bus at a specific time  $t$ .

#### 4.3.5 Fuel consumption

The fuel consumption of the genset is modeled using the maps of engine and generator combined, and assuming that the resulting machine operates along the line that produces the minimum fuel consumption for each level of desired output power (generator electrical power). This is allowed by the fact that, in this case, there are no major mechanical accessories connected directly to the engine shaft, thus the two machines can be considered as a single one. The fuel consumption as a function of the electrical power output is shown in Figure 4.14, and can be expressed analytically as

$$\dot{m}_f = m_0 + m_1 (P_{em} + P_{acc} - P_{batt}) \quad (4.60)$$

Note that, while this equation has the same form as (4.45), its meaning is slightly different: (4.45) is an affine relation between the engine fuel consumption and the engine power, while (4.60) shows an affine relation between the engine fuel consumption and the net electric power at the generator.

### 4.4 Parallel between the two case studies

Since the two case studies share the same architecture and the problem definition is very similar, the control strategies are applied using the same methodology,

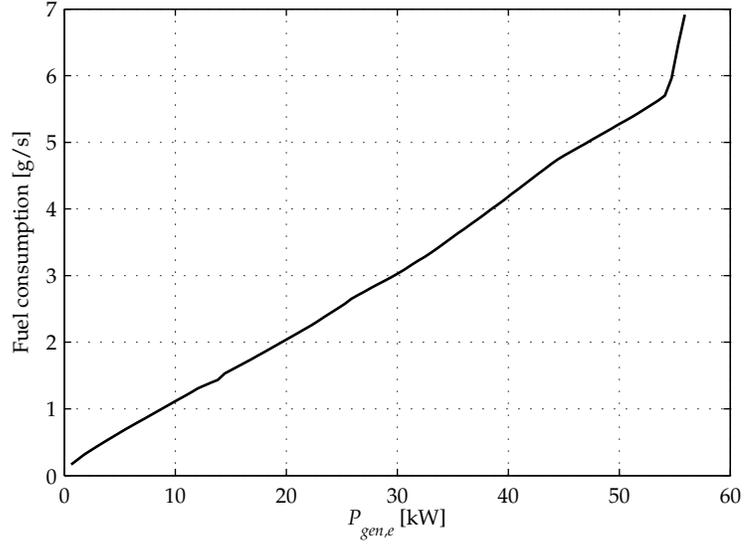


Figure 4.14: Fuel consumption of the genset as a function of the electrical power output

when possible. In order to uniform the notation between the two case studies, in the rest of this chapter both capacitor power and battery power (for case study 1 and 2 respectively) are denoted with  $P_{ress}$ , where the subscript *ress* refers to the acronym *RESS*, i.e. *rechargeable electrical storage device*, the generic definition of both capacitors and batteries. The system dynamic equation is then

$$\dot{\zeta}(t) = -\varepsilon_{ress}(\zeta, P_{ress}) \frac{P_{ress}(t)}{E_{ress}} \quad (4.61)$$

in both cases.

## 4.5 Simulation setup

ECMS and PMP are *causal* energy management strategies, i.e. they can be applied using an input-output formulation and a forward-dynamics vehicle model. The instantaneous power demand  $P_{req}(t)$  (determined by the vehicle model, using a closed-loop speed controller) is split by the energy management strategy

between the genset and the energy storage device, and applied to the powertrain. The acceleration at the current time is then computed and integrated to obtain the speed, which is fed back to the speed controller to determine the power demand at the following time step.

Dynamic programming, on the other hand, is applied in a static manner because the optimal path of SOE can be determined only after the entire cycle has been simulated and the all the admissible SOE paths have been evaluated. This means that the driving cycle (in terms of power demand) must be completely known *before* calculating the optimal power split. This makes dynamic programming an *a-causal* control algorithm, which cannot be applied in real time.

The only method of comparing the three algorithms on the same basis is to implement all of them as off-line optimization, using a pre-determined sequence of power demand (computed using the vehicle simulator described in Chapter 2). ECMS and PMP will perform their instantaneous minimization step by step, while DP will use all the information at once; all strategies, however, will simply split the same total power between the two electrical power sources. The comparison of the different power split and resulting SOE profile will allow to assess the similarities and differences among the three approaches.

## 4.6 Driving cycles

The velocity profile is the sequence of desired vehicle velocity  $V_{veh,des}(t)$ . The vehicle simulator is used to compute the electric power needed by the traction motors. This is obtained assuming the presence of an ideal energy buffer, and thus corresponds to the case in which the traction motors can deliver the entire tractive power request and recuperate the maximum possible amount of braking energy; only the power and torque limitation of the motors themselves is taken into account. When using this information in the implementation of the energy management strategies, the power limits of the storage devices are considered, and therefore the total power request might not be satisfied, especially in the negative power phase (for example if the batteries or capacitors are full and cannot accept more energy). The mechanical brakes are assumed to intervene if the effective

braking power generated by the traction motors is lower than the request, which means that the electric machines are not able to decelerate the vehicle as needed.

#### 4.6.1 Refuse truck

For the first case study, the definition of meaningful driving cycles is relatively easy. In fact, a refuse collection vehicle is operated along specific routes every day. Typical operation of such vehicles in the U.S. include three phases:

1. a trip (mostly on the highway) from the deposit to the city (*Approach*);
2. a shift (8 hours) of refuse collection operation, in urban or suburban conditions (*Routes* with different characteristics);
3. a trip to a dump to discharge the refuse collected, then to the deposit (*Return*).

The approach and return phases have a total duration of 1 to 2 h, depending on the city. The routes can have different characteristics depending on the city or the neighborhood, but in general they are constituted by stop-and-go cycles at low speed (the truck is stopping very frequently to collect refuse from the dumpsters).

Using statistical analysis of data collected on vehicles during operation in several U.S. cities [51], five artificial driving cycles have been created to analyze the vehicle behavior during these phases. The five cycles include one approach cycle, one return cycle, and three different routes, *Route 1*, *Route 2*, and *Route 3*, which have different characteristics. The five test cycles are a statistically representative synthesis of many real driving cycles measured on the field. Each cycle is composed by a velocity profile and a load profile, which is the sequence of power request by the accessory loads and the PTO loads (hydraulic mechanisms for loading, packing, dumping). In terms of the variables defined in Section 4.2.1, the load profile defines the sequence of values of  $P_{acc}(t)$  and  $P_{pto}(t)$  during the cycle. The five cycles are shown in the figures 4.15 through 4.19, which are the same as Figure 2.22.

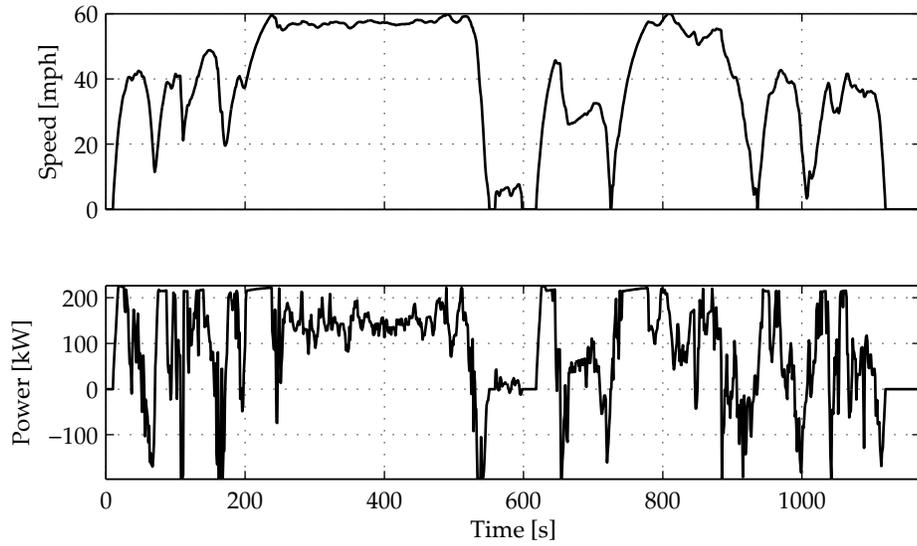


Figure 4.15: Velocity profile and power requests of cycle *Approach*

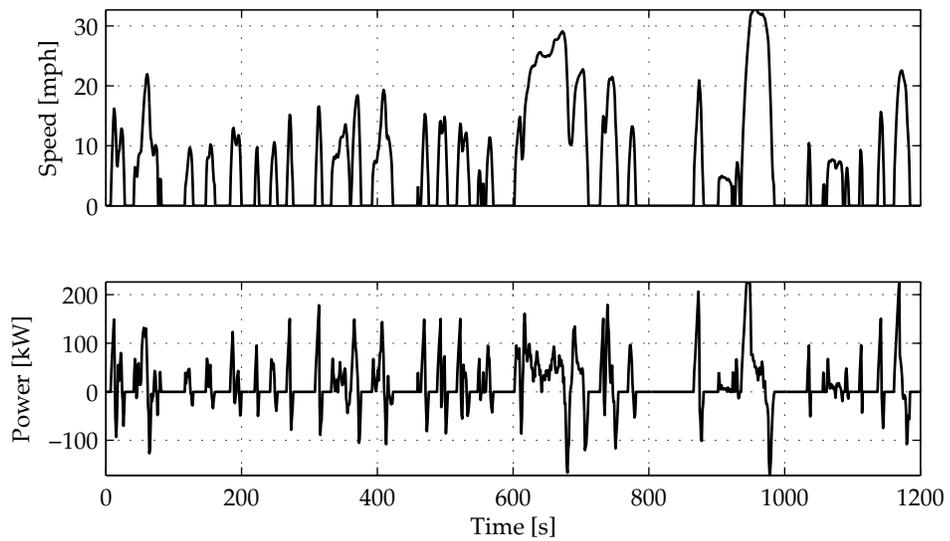


Figure 4.16: Velocity profile and power requests of cycle *Route 1*

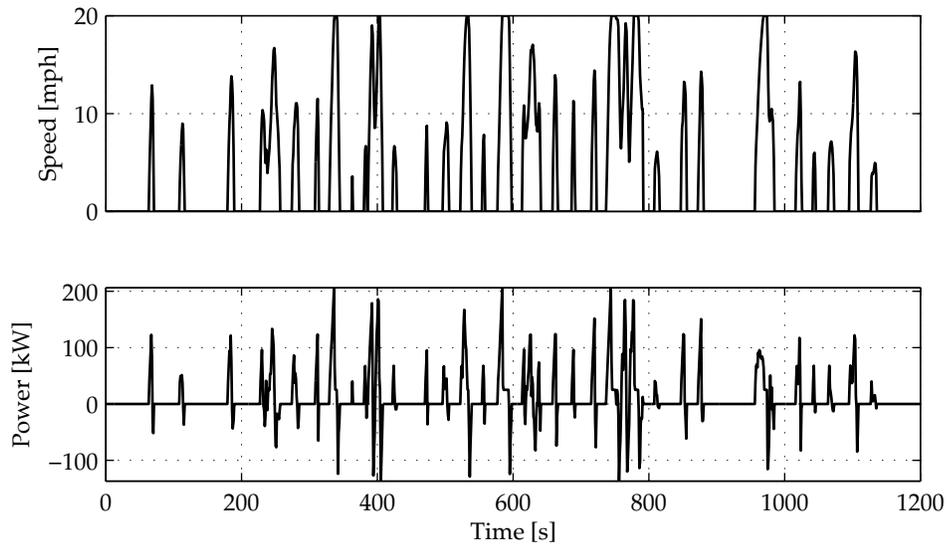


Figure 4.17: Velocity profile and power requests of cycle *Route 2*

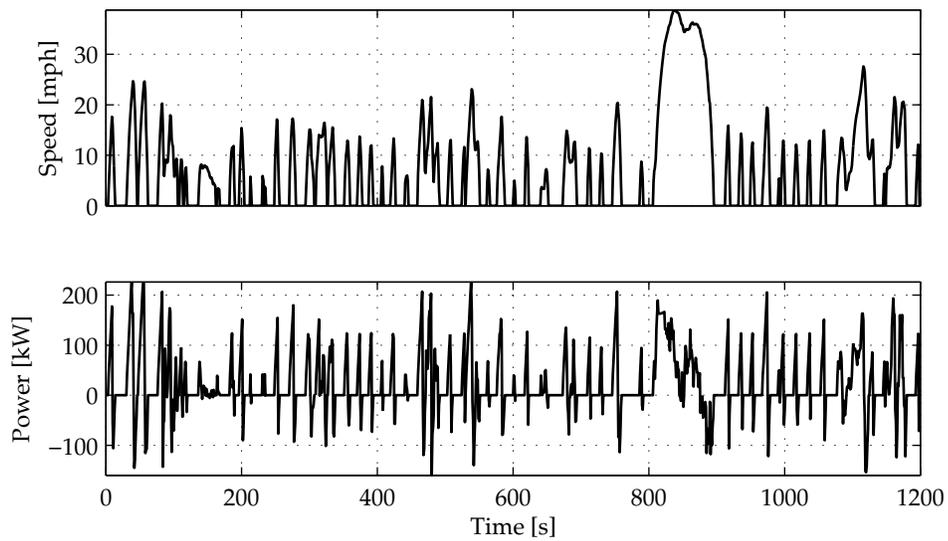


Figure 4.18: Velocity profile and power requests of cycle *Route 3*

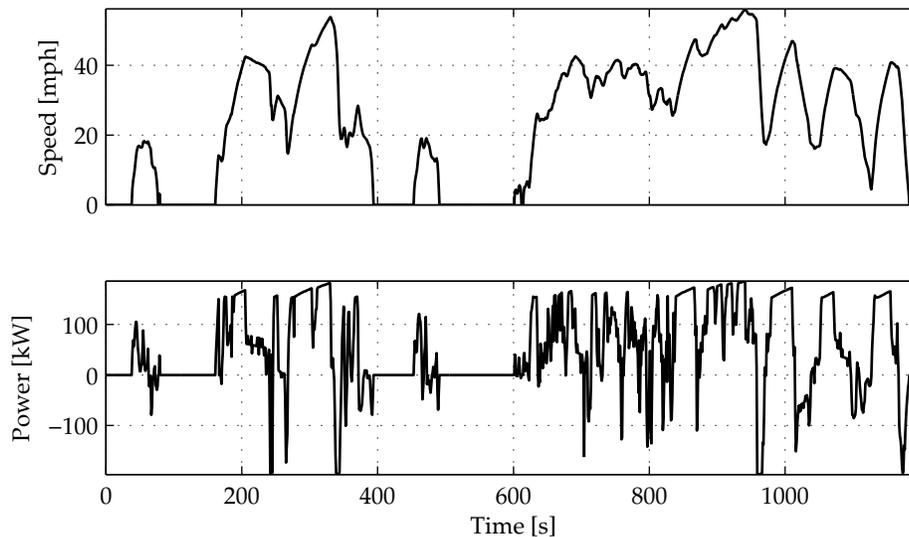


Figure 4.19: Velocity profile and power requests of cycle *Return*

#### 4.6.2 EcoCAR

In this case, there is no specific information about the intended use of the vehicle, since it is a general purpose SUV. Therefore, the simulations are performed using standard (regulatory) driving cycles, in particular the ones used in the U.S. by EPA to assess the fuel economy. Three cycles are considered: UDDS (urban driving dynamometer schedule, urban driving with mild acceleration, Figure 4.20), US06 (urban and suburban driving with higher speed and acceleration than UDDS, Figure 4.21), and FTP highway (highway cycle with almost constant speed, Figure 4.22).

For the charge-sustaining case, each of these cycles is considered independently. For the charge-depleting case, where it is necessary to use a longer driving cycle in order to see a significant decrease of the state of energy, a composite cycle created from a combination of these is used.

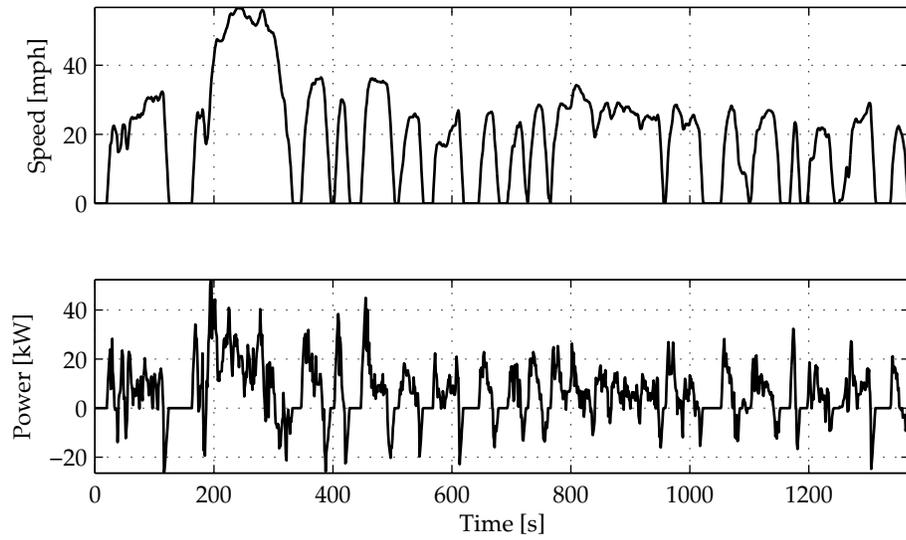


Figure 4.20: Velocity profile and power requests of cycle *UDDS*

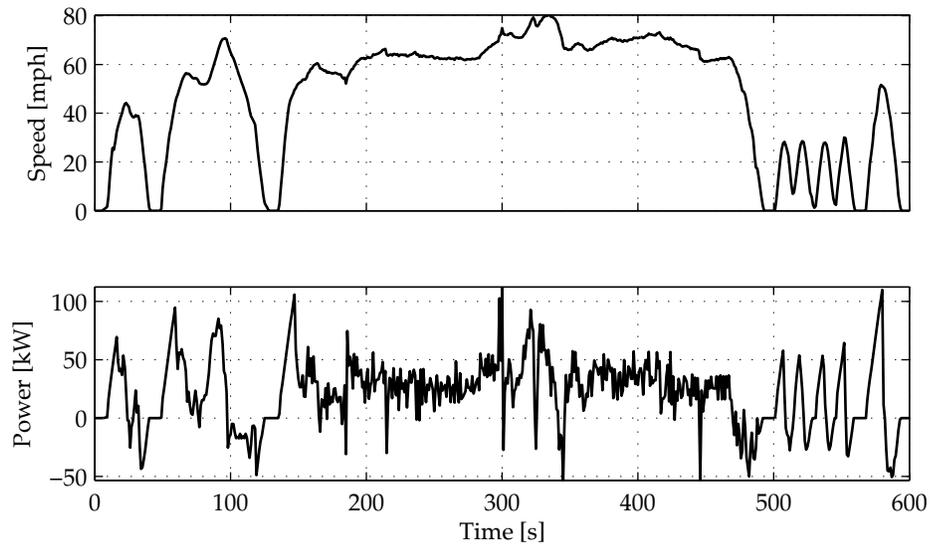


Figure 4.21: Velocity profile and power requests of cycle *US 06*

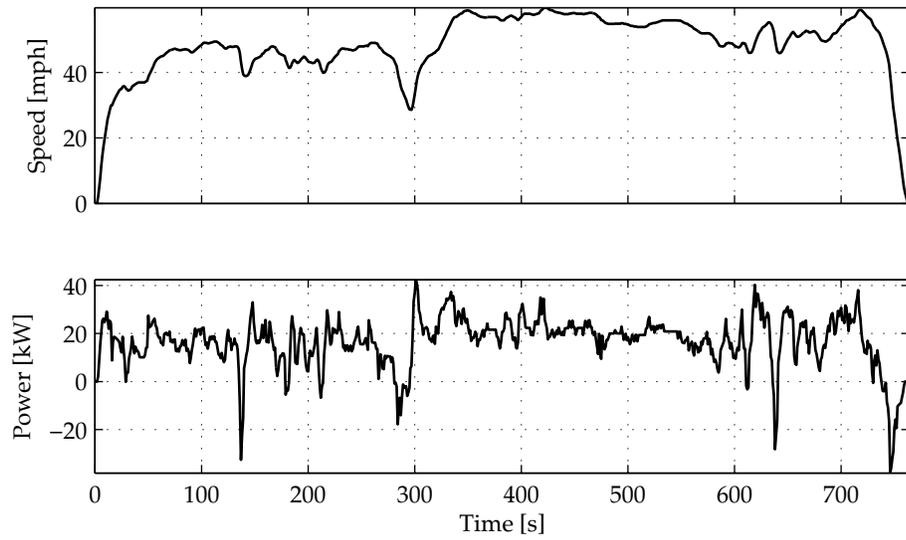


Figure 4.22: Velocity profile and power requests of cycle *FTP highway*

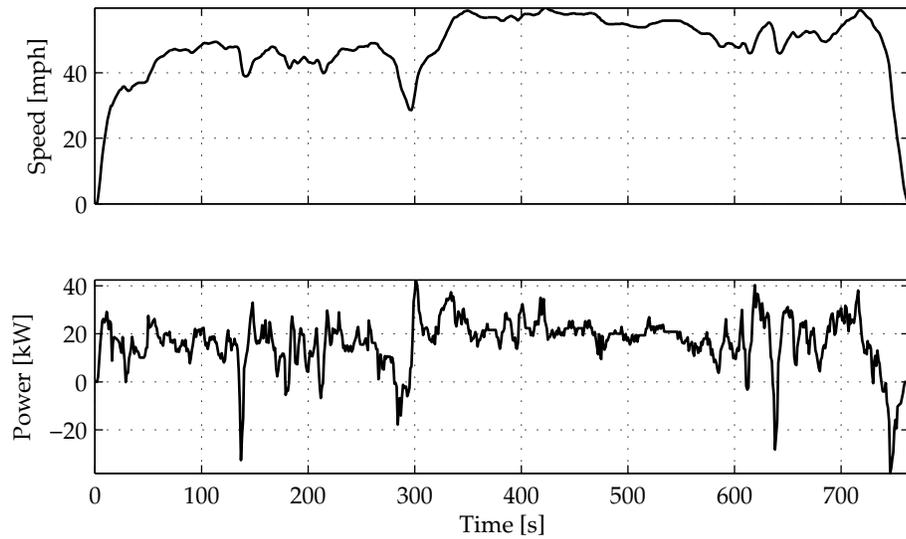


Figure 4.23: Velocity profile and power requests of cycle *FTP highway*

### 4.6.3 Fuel consumption correction

When evaluating the fuel consumption of a charge-sustaining hybrid vehicle, it is necessary to consider the fact that the variation of state of charge in the energy storage device can affect the actual fuel consumption. Therefore, the comparisons between vehicles or control strategies are based on a corrected fuel consumption. The correction assumes that a decrease in SOE can be compensated by running the genset for some time, thus using extra fuel; on the other hand, an increase in SOE can save some fuel since the genset needs to deliver less energy. The amount of extra fuel (or fuel savings) is estimated assuming that the genset operates at average efficiency ( $\bar{\eta}_{genset}$ ) to produce the amount of energy needed to compensate the SOE variation (assuming average RESS efficiency  $\bar{\epsilon}_{ress}$ ):

$$m_{f,corr} = \frac{1}{\bar{\epsilon}_{ress}} (\zeta(t_0) - \zeta(t_f)) \frac{E_{ress}}{Q_{lhv} \bar{\eta}_{genset}} \quad (4.62)$$

In these case studies, the correction term can be expressed using the relations (4.45) and (4.60), which give directly the fuel consumption as a function of the genset power. Therefore, for the refuse truck:

$$m_{f,corr} = \frac{1}{\bar{\epsilon}_{ress}} m_1 g_{10} (\zeta(t_0) - \zeta(t_f)) E_{ress} \quad (4.63)$$

and for the EcoCAR:

$$m_{f,corr} = \frac{1}{\bar{\epsilon}_{ress}} m_1 (\zeta(t_0) - \zeta(t_f)) E_{ress} \quad (4.64)$$

In both cases, this term is added to the value of fuel consumption obtained by integration of the fuel flow rate.

In the charge-depleting case, the total cost already accounts for the SOE variation and the correction term is replaced by the terminal cost.

## 4.7 Dynamic programming

Applying dynamic programming to HEV energy management control means finding the optimal sequence of the appropriate decision variables. The problem

setup for dynamic programming requires a discrete-time description of the system, and a discrete set of values for the decision variable. The procedure is described in Section 3.3.2, and can be applied to the case studies as follows.

Consider the discrete-time system described by discretized version of (4.22) or (4.56):

$$\zeta_{k+1} = \zeta_k - t_s \varepsilon_{ress} (\zeta_k, P_{ress,k}) \frac{1}{E_{ress}} P_{ress,k}, \quad k = 1, \dots, N_t - 1 \quad (4.65)$$

where  $t_s$  is the sampling time,  $N_t$  the length of the optimization horizon (in number of samples), and the subscript  $k$  indicates the value of the variable at the  $k$ -th time step:  $\zeta_k = \zeta(t_k)$  and  $P_{ress,k} = P_{ress}(t_k)$ .

The state of energy of the system is discretized and can only assume one of  $N_\zeta$  values between the minimum and the maximum; the set of values is defined as

$$\zeta^j = \zeta_{min} + (j - 1) \frac{\zeta_{max} - \zeta_{min}}{N_\zeta - 1}, \quad j = 1, \dots, N_\zeta. \quad (4.66)$$

The control policy  $\pi$  is the sequence of state value indices during the optimization horizon:  $\pi = \{j_1, j_2, \dots, j_{N_t-1}\}$ , and defines the state sequence  $\zeta = \{\zeta_1, \zeta_2, \dots, \zeta_{N_t-1}\} = \{\zeta^{j_1}, \zeta^{j_2}, \dots, \zeta^{j_{N_t-1}}\}$ . The control problem is to minimize the total cost

$$J_1(\zeta_1, \zeta_N, \pi) = L_N(\zeta_N) + \sum_{k=1}^{N_t-1} L_k(\zeta_k, P_{ress,k}) \quad (4.67)$$

with respect to the control policy  $\pi$ , i.e. to find the sequence  $\pi$  that generates the lowest cost  $J_1$ .

The arc cost  $L_k$  is defined as the cost incurred when moving from time step  $k$  to time step  $k + 1$ , with the exception of  $L_N$ , which is not actually an arc cost but rather a terminal cost, associated with the final value of the state variable (it has the same role as  $\phi(x_f)$  in the continuous optimal control problem defined in Section 4.3). In the case study 1 (refuse truck), it is assumed  $L_N = 0$  for all values of final state of energy; the final value of state of energy is not defined and the algorithm determines it based on the cost minimization criterion. In case study 2 (EcoCAR) there are two possibilities: in the charge-depleting case,  $L_N$  is defined by (4.53) and depends on the final state  $\zeta_N$ ; again, the algorithm determines the final value

based only on cost minimization. In the charge-sustaining case, the initial and final value of the SOE,  $\zeta_1$  and  $\zeta_N$ , are both pre-defined and assumed to be equal to each other (perfect charge-sustainability): this makes the terminal cost  $L_N$  equal to zero.

The dynamic programming algorithm works by calculating the sequence of minimal cost-to-go backwards in time (i.e., starting from the final instant of the driving cycle), based on Bellman's principle of optimality (§3.3.1). In order to do so, all the arc costs between feasible states must be evaluated (§3.3.2).

The arc cost  $L_k$  is the fuel consumption at time step  $t_k$ , which is calculated using (4.45) or (4.60) (for the refuse truck and the EcoCAR respectively). The fuel consumption depends on the capacitor or battery power and on the loading conditions; these are inputs to the system and are determined given the driving cycle. The RESS power is the control variable to be optimized. However, the actual decision variable of the algorithm is the system state,  $\zeta$ , rather than the control input. That is, the dynamic programming algorithm determines the optimal sequence of state of energy, and then, as a consequence, the power that produces it. The reason for choosing the state of energy as the decision variable lies in the fact that it is easier to implement the state constraints in this way, since only the range of admissible state values is considered and therefore it is impossible, by construction, to exceed the state boundaries. Given the relatively simple dynamic equation (4.65), the relation between state variation and control variable is immediate and allows for this particular formulation of the problem.

#### 4.7.1 State discretization

The decision variable, i.e. the state of energy  $\zeta$ , can assume values in a set defined as a finite number of elements,  $N_\zeta$ , between the lower and upper bounds. At each time instant, any of these values is, in principle, admissible. If  $N_t$  is the number of time instants that compose the driving cycle, the overall domain of admissible state values can be depicted as a matrix with  $N_t$  columns and  $N_\zeta$  rows. Each column represents all the state values admissible at a given time. The matrix representing the state discretization during the cycle can be visually represented as in Figure 4.24. If the initial and final value of the state, i.e.  $x_1$  and  $x_{N_t}$ , are given

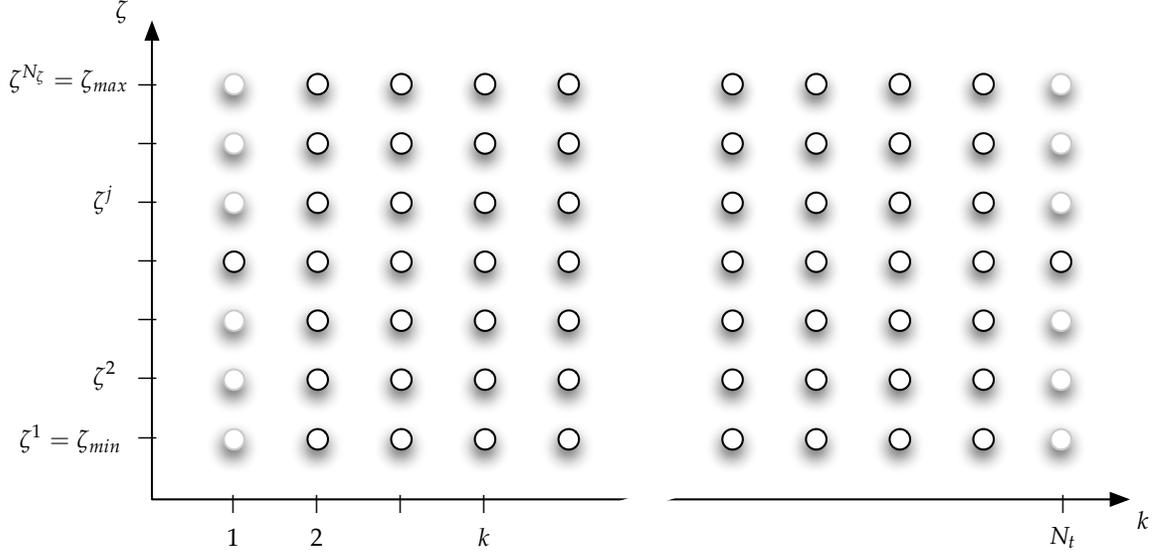


Figure 4.24: SOE discretization for dynamic programming

(as it is the case in most optimization problems), then the first and last column of the matrix have only one admissible value, corresponding to the assigned terminal condition. The subscript indicates the time index (or column index), and the superscript indicates the value of the variable (or row index). Thus,  $\zeta_k^j$  indicates the  $j$ -th value of  $\zeta$  at time  $k$ , as defined by (4.66).

The algorithm determines the sequence of SOE values that minimizes the assigned cost (fuel consumption), or, in other words, decides a path along the matrix of admissible values (as shown in Figure 4.25).

#### 4.7.2 Arc cost and cost-to-go

Moving from one value of SOE at time  $k$  to another (possibly equal) value at time  $k + 1$  implies a value of RESS power, which depends on the SOE variation and is calculated inverting (4.65). If the value at time  $k$  is  $\zeta^i$  and the value at time  $k + 1$  is  $\zeta^j$ , the RESS power that moves the system state from  $\zeta_k^i$  to  $\zeta_{k+1}^j$  is

$$P_{ress,k} \left( \Delta \zeta_k^{ij}, \zeta_k \right) = - \frac{E_{ress}}{t_s \varepsilon_{ress} (\zeta_k, P_{ress,k})} \Delta \zeta_k^{ij} = - \frac{E_{ress}}{t_s \varepsilon_{cap} (\zeta_k, P_{ress,k})} \left( \zeta_{k+1}^j - \zeta_k^i \right) \quad (4.68)$$

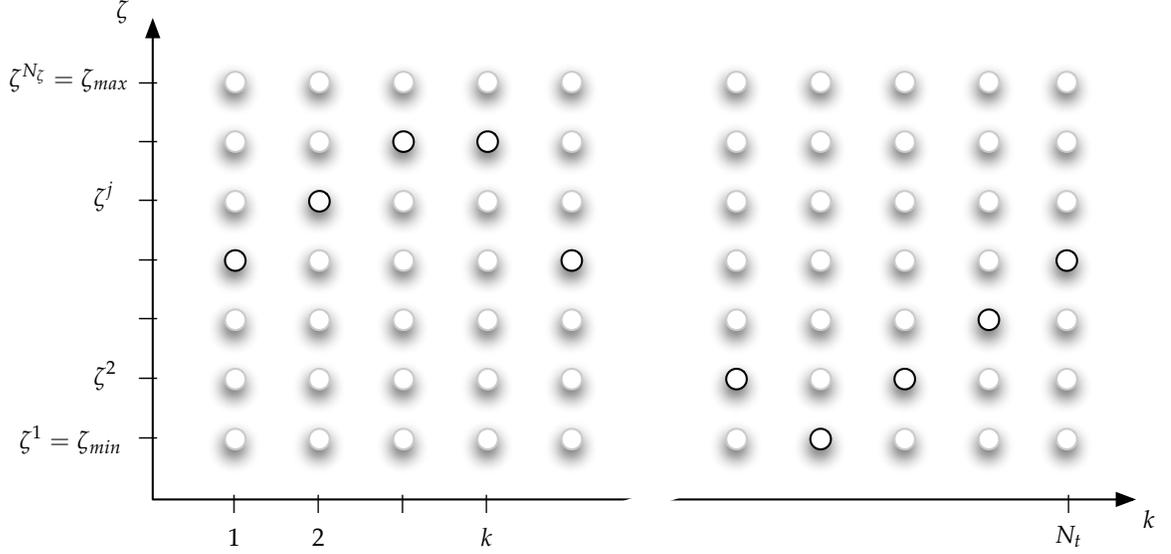


Figure 4.25: Example of optimal SOE sequence

where  $\Delta\zeta_k^{ij}$  is defined as the variation of SOE from time  $k$  to time  $k + 1$ , and indicates the fact that the SOE values changes from  $\zeta^i$  at time  $k$  to  $\zeta^j$  at time  $k + 1$ . The map  $\varepsilon_{ress}(\zeta_k, P_{ress,k})$  depends on the RESS power, i.e. on the result of (4.68), but in practice it is possible to use an approximation and assume an ideal RESS power, defined as  $P_{ress,k}^{ideal} = E_{ress}\Delta\zeta_k^{ij}/t_s$ , as the input for the map  $\varepsilon_{ress}(\zeta_k, P_{ress,k})$ : the result is approximated but acceptable, and can be refined using an iterative procedure (i.e., guess  $P_{ress,k}$  as  $P_{ress,k}^{ideal}$ , find  $P_{ress,k}$  using (4.68); replace this value of  $P_{ress,k}$  in the map  $\varepsilon_{ress}(\zeta_k, P_{ress,k})$  and solve (4.68) again for an improved estimate of  $P_{ress,k}$ ; repeat until two successive iterations give very close values of  $P_{ress,k}$ ).

The arc cost associated with this transition is the fuel consumption, calculated using (4.60); following the same convention used for the SOE variation, the arc cost incurred when moving from  $\zeta_k^i$  to  $\zeta_{k+1}^j$  is denoted as  $L_k^{ij}$ .

The cost-to-go at time step  $k$  is denoted as  $J_k$ .  $J_k$  represents the cost of moving the state from time step  $k$  to the final time step,  $N_t$ , following the optimal path. The cost-to-go at a specific time  $k$  depends, obviously, on the value of the state at that time; this is indicated by the notation  $J_k^i$ , which means the cost-to-go incurred in moving from state  $\zeta^i$  at time  $k$  to the terminal condition  $\zeta_{N_t}$  at time  $N_t$ .

At the end of the optimization horizon ( $k = N_t$ ), the cost-to-go is equal to the terminal cost:  $J_{N_t} = L_{N_t}$  (it does not depend on  $i$  because there is only one admissible  $i$  at the final stage); for any other value of  $k = 1, \dots, N_t - 1$ , the cost-to-go is defined by the following recursive relation, arising from Bellman's principle of optimality:

$$J_k^i = \min_j \left( L_k^{ij} + J_{k+1}^j \right) \quad (4.69)$$

This recursive formula can be explained as follows: the cost-to-go  $J_k^i$ , i.e. the lowest possible cost incurred while moving from state  $\zeta_k^i$  to the end of the optimization horizon (to the final state  $\zeta_{N_t}$ ), depends on the value of the state at the next time step,  $\zeta_{k+1}^j$ , and on the corresponding cost-to-go  $J_{k+1}^j$ . The lowest cost-to-go at time  $k$  is achieved by choosing the value  $\zeta_{k+1}^j$  (i.e. by choosing  $j$ ) such that the sum of the arc cost from  $\zeta_k^i$  to  $\zeta_{k+1}^j$  and of the cost to go from  $\zeta_{k+1}^j$  to the end is minimal. This is why the cost-to-go of each node  $i$  at time  $k$  depends on the index  $j$  of the state at the following time  $k + 1$ ; this dependency is represented by the arc cost  $L_k^{ij}$ , and by the fact that the minimization is performed with respect to the index  $j$  that defines the next time step.

The optimal path  $\pi^* = \{j_1^*, j_2^*, \dots, j_{N_t-1}^*\}$  is defined as the sequence of indices  $j$  that generate the cost-to-go (4.69) at each time step  $k$ , i.e.

$$\pi_k^* = \arg \min_j \left( L_k^{ij} + J_{k+1}^j \right), \quad k = 1, \dots, N_t - 1 \quad (4.70)$$

### 4.7.3 Implementation issues

As seen in the previous section, the optimal sequence of control actions is calculated once the arc cost  $L_k^{ij}$  is defined for each time step  $k$  and each pair of SOE indices  $i, j$ . The problem is set up in a quasi-static fashion: the instantaneous values of the parameters  $P_{em1,e}(t_k)$ ,  $P_{em2,e}(t_k)$ ,  $P_{pto}(t_k)$ , and  $P_{acc}(t_k)$  are derived from the definition of the driving and loading cycle, using the simulator described in Chapter 2. Since the driving cycle is defined using a velocity profile discretized

with a sampling time of 1 s, the same value is used for the dynamic programming algorithm, i.e.  $t_s = 1$  s in (4.65).

The implementation of the recursive minimization (4.69) uses a matrix formulation of the cost to go and arc cost, in which the superscripts  $i, j$  are the indices of the element in the corresponding matrix.

The number  $N_\zeta$  of discrete values of the state of energy  $\zeta$  can be seen as a tuning parameter for the dynamic programming algorithm: a higher number generates a finer quantization and therefore a solution closer to the continuous optimal solution, but also increases the total computational time of the algorithm. This can be estimated as follows:

$$t_{comp} = N_t \cdot N_\zeta^2 \cdot t_{comp,L} \quad (4.71)$$

where  $t_{comp,L}$  is the time necessary to evaluate the arc cost for one SOE variation  $\Delta\zeta^{ij}$ . The total time is linearly increasing with the length of the optimization horizon,  $N_t$ , and proportional to the square of the discretization parameter  $N_\zeta$ . The reason for the square is that, at any time step, all the combinations of arc costs  $\Delta\zeta^{ij}$  should be evaluated, with both  $i$  and  $j$  ranging from 1 to  $N_\zeta$ .

The memory requirement for the algorithm can be more limiting than the computational time: in fact, it is necessary to build (column by column) and store a cost-to-go matrix  $J$  of dimension  $(N_\zeta \times N_t)$ , and to keep in memory an arc cost matrix  $L_k^{ij}$  of dimension  $(N_\zeta \times N_\zeta)$  at each time  $k$ . The arc cost matrix is recalculated at each time step and only the matrix corresponding to the current time step  $k$  needs to be accessed, while the matrices corresponding to the time steps  $k + 1, \dots, N_t - 1$  (used in the preceding optimization steps) can be discarded.

In order to accelerate the execution of the algorithm and reduce its memory requirements, the most obvious solution is to reduce the number  $N_\zeta$  of discrete SOE values considered, which reduces the quality of the solution (evaluating fewer possible solutions). However, several expedients can be implemented to optimize the execution time and memory requirements without reducing the number of SOE levels:

1. consider that the arc cost is mainly a function of the SOE *variation*, more than a function of the SOE *value*: the cost of arcs characterized by the same  $\Delta\zeta$  is very similar, the only difference being due to the effect of  $\zeta_k$  on  $\varepsilon_{cap,k}$ , which is relatively small (see Figure 4.3). Thus, it is possible to reduce the number of evaluations necessary to build the arc cost matrix by lumping together the elements characterized by the same  $\Delta\zeta$  and similar initial values: that is, rather than considering  $N_\zeta$  different values of initial SOE for each value of SOE variation ( $N_\zeta^2$  combinations), it is possible to consider a reduced number of “zones”, in which the effect of  $\zeta_k$  on  $\varepsilon_{cap}(\zeta_k, \Delta\zeta_k)$  can be assumed to be constant. If  $N_\varepsilon$  is the number of zones ( $N_\varepsilon \ll N_\zeta$ , for example  $N_\varepsilon = 10$  while  $N_\zeta = 500$ ), the number of different combinations in the matrix  $L_k^{ij}$  is  $N_\varepsilon \times (2N_\zeta - 1)$ , where  $2N_\zeta - 1$  is the total number of different  $\Delta\zeta$  values. Thus, the number of evaluations (and of meaningful elements) for the arc cost matrix is reduced from  $N_\zeta^2$  to  $N_\varepsilon \times (2N_\zeta - 1)$ : if  $N_\varepsilon = 10$  and  $N_\zeta = 500$ , this means a reduction from 250000 to 9990 function evaluations.
2. use integer numbers rather than double-precision real numbers to describe numerically each element in the SOE matrix and in the matrix  $\pi$  of the indices corresponding to the optimal cost-to-go: since these elements are discretized, using just their integer indices rather than actual values does not reduce the accuracy in any way, but decreases the memory usage for large matrices. Going a step further, it is also possible to use single-precision real numbers or a custom quantization to describe the elements of the cost-to-go and arc cost matrices, with a very slight reduction in their accuracy. The accuracy reduction has no effect on the algorithm results as long as the difference between close values is preserved, since what is important for both matrices is to find their minimum value, not the exact value of each element.

#### 4.7.4 Simulation results, refuse truck

The results obtained from dynamic programming implementation are analyzed in this section. The *Approach* cycle, shown in Figure (4.15), is used as example.

The dynamic programming solution is represented in Figure 4.26, which shows the state of energy of the capacitors during the cycle. In order to better understand this solution, part of the cycle is shown in detail in Figure 4.27. In this case, the figure also shows the power delivered by the capacitors and by the genset, compared to the total power request. The capacitors absorb all the negative power made available by the traction machines, then deliver it gradually during the acceleration phase. Using the capacitors gradually, at low power, rather than discharge them completely keeping the engine at idle, may appear counter-intuitive. However, it is coherent with the problem definition, and in particular with the shape of the fuel consumption characteristic and the RESS effectiveness. The fuel consumption is an affine function of the genset power, which means that the genset efficiency increases for increasing output power; conversely, the RESS effectiveness is closest to 1 (maximum efficiency) for lower power levels (Figure 4.3). Thus, the best policy is to discharge the capacitors (using the energy stored in them during regenerative braking), but do so slowly, in order to minimize the losses. The rate of discharge selected by dynamic programming is the lowest power that can be delivered while discharging the RESS enough to leave room for the recharge occurring during the subsequent braking event. The reason for operating in the region of high SOE resides, once again, in the charge-effectiveness function, which is closer to the unit value for higher state of charge. Similar considerations hold for the other driving cycles, which are not shown here for brevity.

Dynamic programming, being the closest approximation of the optimal solution to the energy management problem, is often used as a method to benchmark other strategies: this is done in Section 4.10. It is also used to determine the theoretical improvements of hybridization for a given vehicle (theoretical because based on the assumption of a perfect energy management). As an example of this, the energy flow diagram for the hybrid refuse truck and the conventional version of the same vehicle are shown in Figure 4.28. This kind of diagram shows the amount of fuel energy introduced into the vehicle and the way it is used. In Figure 4.28, it is clear how the net energy generated at the wheels, i.e. the sum of kinetic energy, rolling resistance and aerodynamic resistance, is the same for both vehicles (except

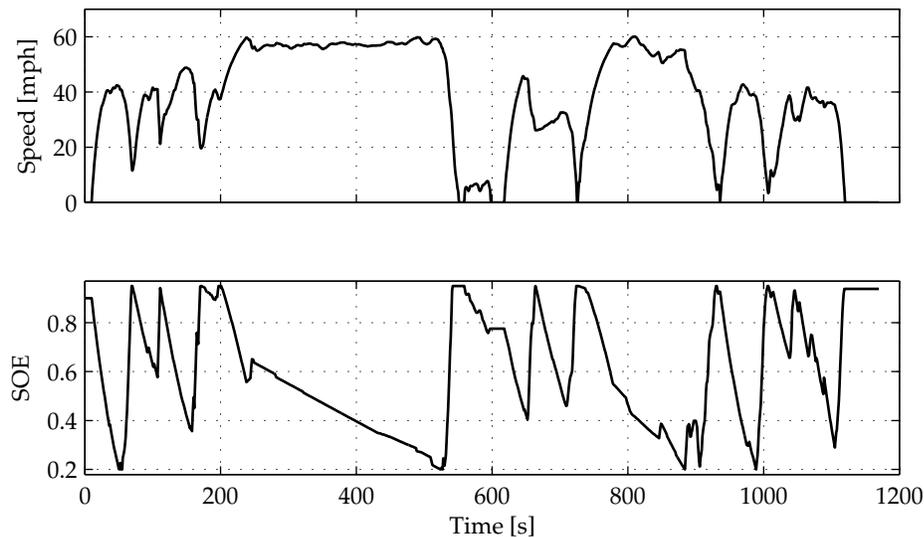


Figure 4.26: Dynamic programming solution for refuse truck, cycle *Approach*

for a small difference in kinetic energy due to the slightly higher mass of the hybrid version); however, the fuel energy needed to generate it is much higher in the conventional vehicle, for two reasons: higher powertrain inefficiencies (mainly due to the presence of a torque converter) and recuperation of kinetic energy using regenerative braking in the hybrid case, which reintroduces in the powertrain almost half of the kinetic energy generated at the wheels, which is lost in the conventional case. The overall reduction in fuel consumption with the hybrid powertrain is over 30 %, justifying the interest in hybridization of this kind of trucks.

#### 4.7.5 Simulation results, EcoCAR, charge-sustaining

In the charge-sustaining case, the dynamic programming algorithm is set to find a solution with identical initial and final SOE. The SOE at the terminal points of the optimization interval is set to the average between the minimum and maximum SOE values, set to 0.32 and 0.38 respectively.

The solution obtained for the UDDS cycle is shown in Figure 4.29 and a detail with the power split is reported in Figure 4.30. The behavior is very similar to what

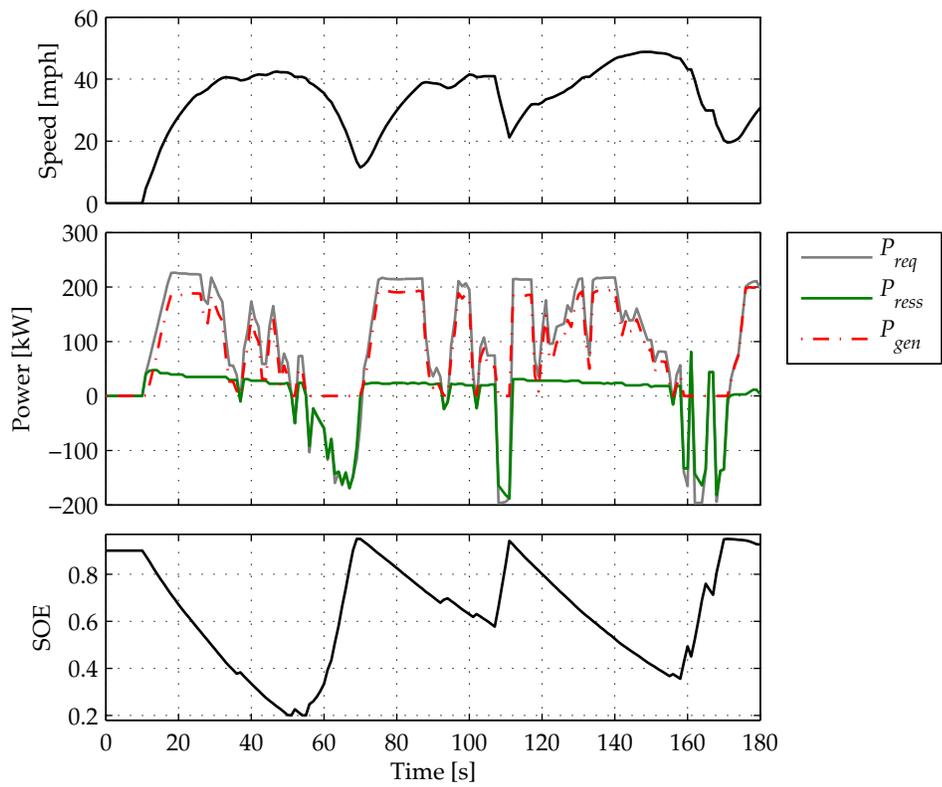
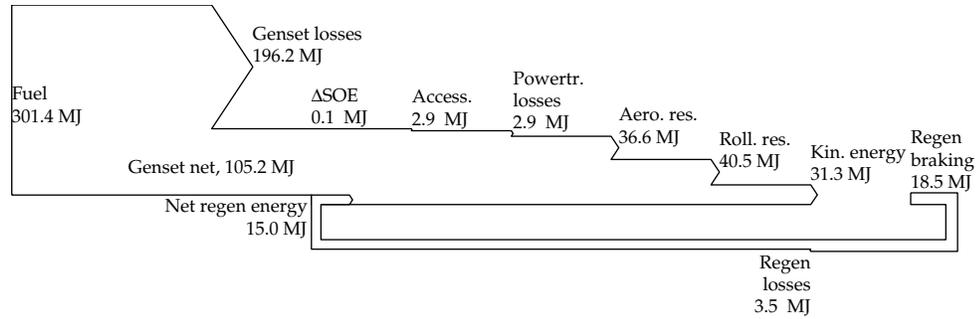
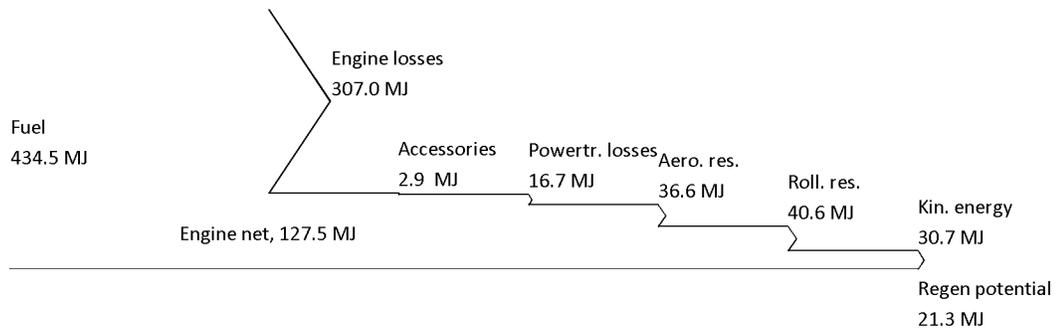


Figure 4.27: Dynamic programming solution for refuse truck, cycle *Approach* (detail of the first 180 s)



(a) Hybrid vehicle, dynamic programming solution



(b) Conventional vehicle

Figure 4.28: Energy flow diagram corresponding to the dynamic programming solution for refuse truck, cycle *Approach*, compared to the results obtained in the same cycle by the conventional version of the same vehicle.

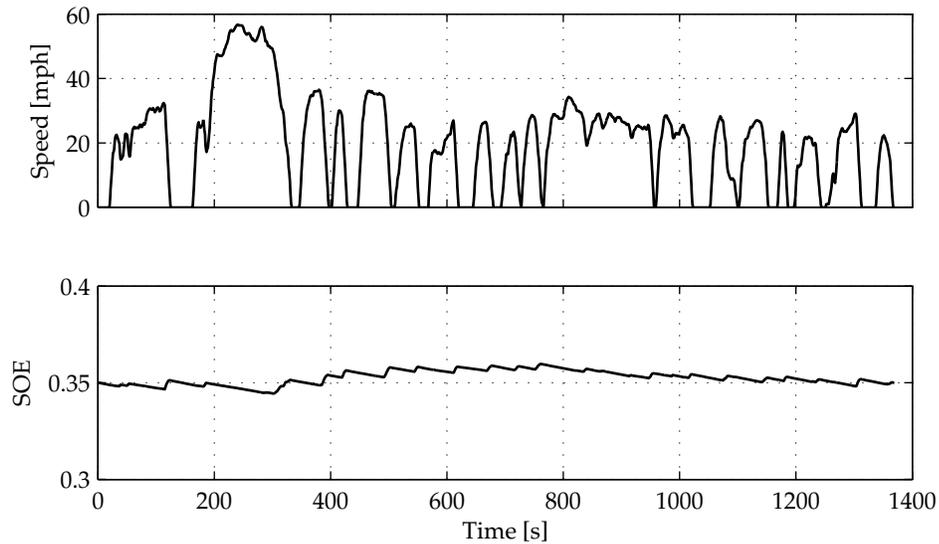


Figure 4.29: Dynamic programming solution for EcoCAR, cycle *UDDS*

was observed for the first case study, with the batteries being slowly discharged during acceleration and charged during deceleration with regenerative braking.

## 4.8 Pontryagin's minimum principle

In this section, Pontryagin's minimum principle, described in Section 3.2, is formulated for the two case studies, given the control problem defined in Section 4.2 and Section 4.3. The two cases are the same except for the fact that in the first case there is no constraint on the final value of the state of energy, nor a terminal cost is defined, while in the second case one of these is present.

The minimum principle<sup>2</sup> states that the optimal control law  $u^*(t)$  must satisfy the following necessary conditions [34]:

<sup>2</sup>see Section 3.2.2 for more detailed background information

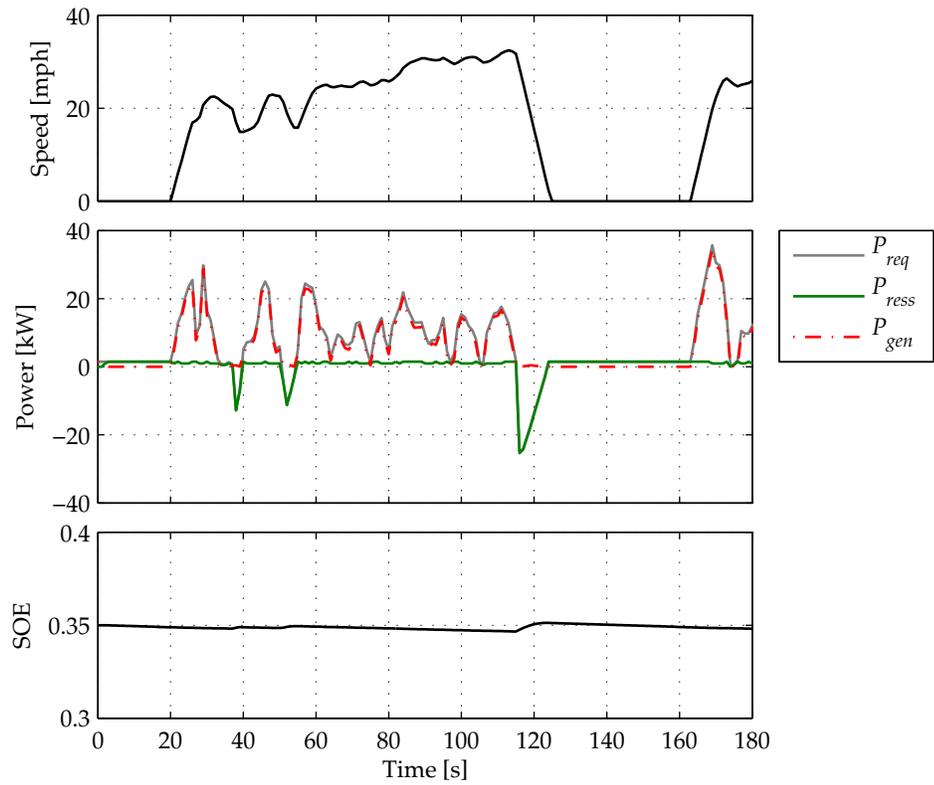


Figure 4.30: Dynamic programming solution for EcoCAR, cycle *UDDS* (detail of the first 180 s)

1.  $u^*(t)$  minimizes at each instant of time the augmented Hamiltonian of the system

$$H(t, u(t), x(t), \lambda(t)) = \lambda(t)' \cdot f(x(t), u(t), t) + L(u(t), t) + \mu \Gamma(x(t), t) \quad (4.72)$$

where:  $\lambda(t) \in \mathbb{R}^n$  is a vector of adjoint state variables (with the same dimension as the state vector  $x(t)$ );  $L(u(t), t)$  is the instantaneous cost (fuel consumption); and  $\mu \Gamma(x(t), t)$  is a penalty for reaching the boundary conditions of the state of energy, with  $\mu$  a constant and

$$\Gamma(x(t), t) = \begin{cases} 0 & \text{if } G(x(t), t) < 0 \text{ (constraints not active)} \\ \frac{\partial G(x,t)}{\partial x} \cdot f(x(t), u(t), t) & \text{if } G(x(t), t) \geq 0 \text{ (constraints active)} \end{cases} \quad (4.73)$$

2. the co-state dynamic equation is  $\dot{\lambda} = -\nabla_x H|_{u^*, x^*}$ , and the co-state jumps by the quantity  $\mu_0$  any time the state reaches one of the limits (i.e., in the instant in which  $G(x(t), t)$  becomes zero).
3. the terminal conditions on the state are  $x(t_f) = x_f$  if the final state is assigned (otherwise they are not defined)
4. the terminal conditions on the co-state are given by  $\lambda^*(t_f) = \left. \frac{\partial \phi(x(t_f), t_f)}{\partial t_f} \right|_{*, t_f}$  if the terminal cost  $\phi(x(t_f), t_f)$  is present (otherwise they are not defined).

In the system described, the state equation is (4.22) or (4.56), written in terms of the factor  $\varepsilon_{ress}$  and of the control variable  $u = \{P_{ress}\}$ :

$$\dot{\zeta}(\zeta, P_{ress}) = -\varepsilon_{ress}(\zeta, P_{ress}) \frac{P_{ress}(t)}{E_{ress}}. \quad (4.74)$$

The Hamiltonian of the system is

$$H(t, P_{ress}, \zeta, \lambda) = -\lambda(t) \varepsilon_{ress}(\zeta, P_{ress}) \frac{P_{ress}(t)}{E_{ress}} + \dot{m}_f(P_{ress}(t), t) + \mu \Gamma(\zeta(t), t) \quad (4.75)$$

where the function  $\dot{m}_f(P_{ress}(t), t)$  is the fuel consumption expressed in terms of the control input  $P_{ress}(t)$  and of the time-varying parameters  $P_{acc}(t)$ ,  $P_{pto}(t)$ ,  $P_{em1,e}(t)$

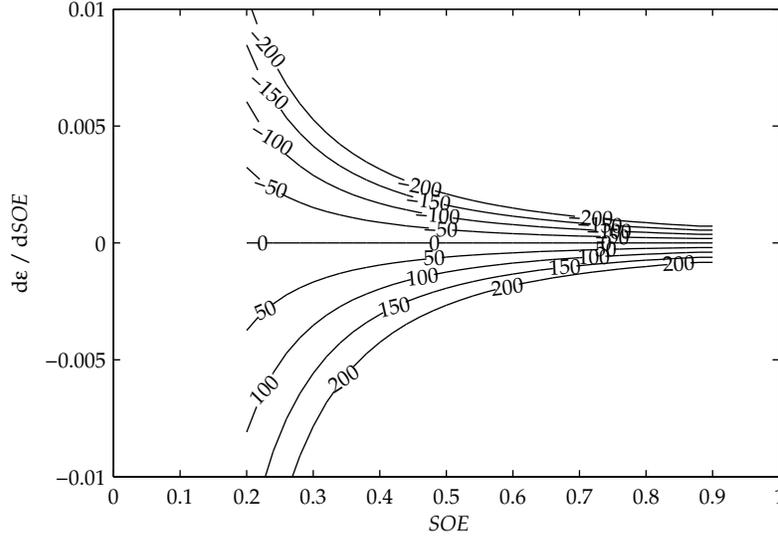


Figure 4.31: Derivative of  $\varepsilon_{cap}(\zeta, P_{cap})$  with respect to  $\zeta$ , for case study 1

and  $P_{em2,e}(t)$  or  $P_{em,e}(t)$ . The fuel consumption along the maximum efficiency line is expressed using the analytical form (4.45) (case study 1) or (4.60) (case study 2).  $\varepsilon_{ress}(\zeta, P_{ress})$  is tabulated as in figures 4.3 and 4.12.

The co-state dynamic equation is

$$\dot{\lambda}(t) = -\frac{\partial H}{\partial \zeta} = -\lambda(t) \frac{P_{ress}(t)}{E_{ress}} \frac{\partial \varepsilon_{ress}(\zeta, P_{ress})}{\partial \zeta} \quad (4.76)$$

where the term  $\frac{\partial \varepsilon_{ress}(\zeta, P_{ress})}{\partial \zeta}$  can also be tabulated, and is shown in Figure 4.31 for case study 1 (capacitors, refuse truck), and in Figure 4.32 for case study 2 (Li-Ion batteries, EcoCAR).

The terminal conditions are defined only for the second case study, the EcoCAR. In the charge sustaining case, they are referred to the final value of state of energy:

$$\zeta^*(t_f) = \zeta(t_0) = \zeta_0, \quad (4.77)$$

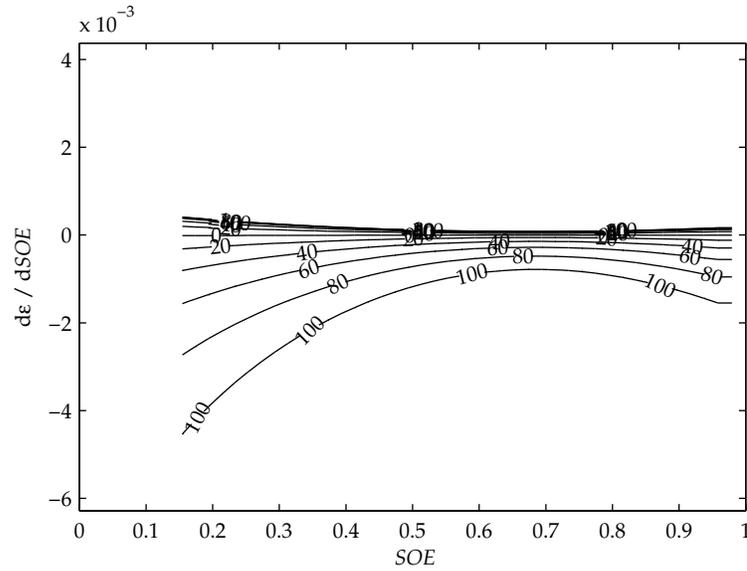


Figure 4.32: Derivative of  $\varepsilon_{batt}(\zeta, P_{batt})$  with respect to  $\zeta$ , for case study 2

while in the charge-depleting case they are defined for the co-state and derived from the terminal cost (4.53), as follows:

$$\lambda^*(t_f) = \left. \frac{\partial \phi(\zeta(t_f), t_f)}{\partial \zeta} \right|_{*, t_f} = - \frac{E_{batt, max}}{Q_{lhv}} \frac{C_{elec}}{C_{fuel}} \quad (4.78)$$

Note how this value of  $\lambda(t_f)$  is invariant with respect to the driving cycle and the solution.

The minimization of the Hamiltonian function with respect to the control variable  $P_{ress}$  can be done numerically. Given the admissible range of control values defined by (4.59), the value that minimizes the Hamiltonian can be easily found by enumeration of a finite number of admissible control values<sup>3</sup>. In other words, at each instant, the Hamiltonian (4.75) is evaluated for each of the elements in a

<sup>3</sup>In simulation, evaluating a simple algebraic function like the Hamiltonian even for a large number of cases is done with a negligible computational effort, hence the enumerative technique is the simplest and easiest way to approach the minimization. The situation is different in microcontroller implementation, where the limitations in terms of processing hardware and memory become much more important than they are on a personal computer.

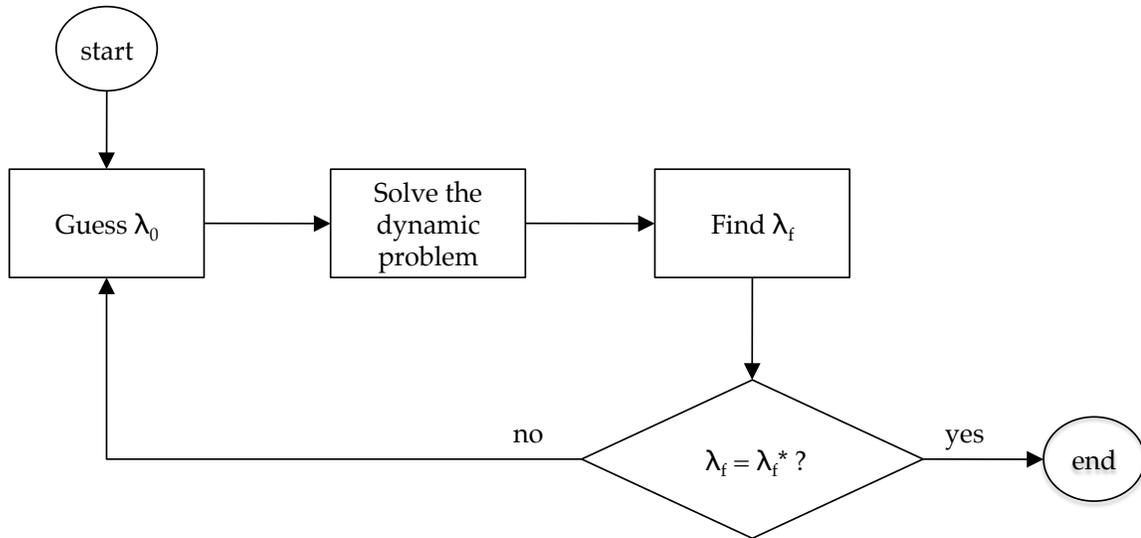


Figure 4.33: Flow chart of iterative solution for Pontryagin's minimum principle

set of  $N_u$  values of RESS power, equispaced in the interval  $[P_{ress,inf}, P_{ress,sup}]$  (as defined by (4.31) or (4.59), for the two cases respectively). The value of power that minimizes the Hamiltonian is chosen and applied to the system, and also used to update the numerical solution of the co-state dynamic equation (4.76).

The evolution of the co-state  $\lambda(t)$  obtained by integration of (4.76) is used in the definition of the Hamiltonian function; in addition to this, the final co-state value  $\lambda(t_f)$  must satisfy the terminal condition (4.78) if it is required.

If terminal conditions are imposed, on either the state or co-state variable, an iterative procedure is necessary in order to find a solution that satisfies them. The procedure, represented in Figure 4.33, can be outlined as follows:

1. assume an arbitrary value of the initial value of the co-state,  $\lambda(t_0) = \lambda_0$  (this is always a free variable);
2. solve the problem finding the sequence of controls that minimize the Hamiltonian, which depends on  $\lambda_0$ ;
3. integrate the dynamic equations in order to calculate the final values  $\zeta(t_f)$  and  $\lambda(t_f)$

4. compare the terminal value of the state or co-state variable with its reference value; if different (by more than some tolerance), define a new value of  $\lambda_0$  and repeat. The new value of  $\lambda_0$  can be determined using any algorithm; the simplest method is to increment  $\lambda_0$  at each iteration by a fixed amount.

#### 4.8.1 Simulation results, refuse truck

The solution method outlined was applied to the driving cycles described in Section 4.6. The power demand generated during each cycle is used as an input for the strategy, and the sequence of RESS power is calculated. In this case study there is no terminal condition, thus the problem has a free parameter, the initial value of the co-state variable,  $\lambda_0$ . This parameter affects the behavior of the solution because it changes the values of  $\lambda(t)$  during the entire optimization horizon, and therefore the Hamiltonian, since the co-state  $\lambda(t)$  has the function of relative weight of the terms in the Hamiltonian function. Since in this problem there are no terminal constraints,  $\lambda_0$  can be chosen freely and is selected as the value that minimizes the overall fuel consumption. Figure 4.34 shows the effect of  $\lambda_0$  on fuel consumption, for the cycle *Approach*. Two values of fuel consumption are shown: one is the actual fuel consumption, the other is corrected with a term that accounts for the variation in state of energy, as explained in Section 4.6.3. Because of the small size of the energy buffer, the two curves are almost superimposed, which justifies the assumption of neglecting the SOE variation in this application. Both curves show a minimum for  $\lambda_0 = 103$  g, which is selected as the optimal value. The units of  $\lambda$  are grams since the fuel consumption is computed in g/s using (4.45). If fuel power were chosen as the instantaneous cost, then the units of  $\lambda$  would be Joules. Thus, when expressed in grams,  $\lambda$  can be interpreted as the energy equivalent of that mass of fuel. The value of 103 g that gives the optimal results is equivalent to roughly twice the size of the energy buffer (which is about 2 MJ, equivalent to 47 g of diesel fuel).

In order to see the effect of the initial value of co-state on the solution, the SOE profiles corresponding to three different values of  $\lambda_0$  are shown in Figure 4.35. The three values are the extremes of the domain considered in Figure 4.34 and the

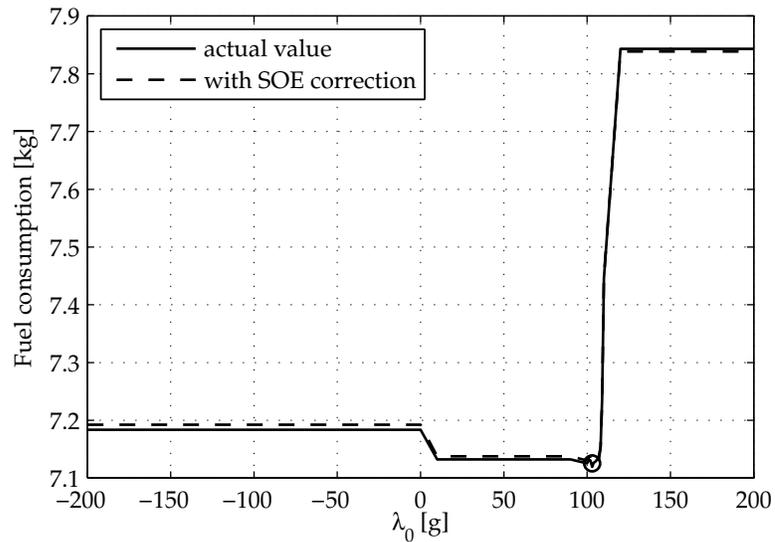


Figure 4.34: Application of the minimum principle to the refuse truck: effect of  $\lambda_0$  on fuel consumption (cycle *Approach*)

value selected as optimal. The variation of  $\lambda(t)$  during the cycle is also reported in the figure, and can be seen how its variation is very small, thus the initial value defines it for the entire optimization horizon. A high value of  $\lambda_0$  (200 g) makes the strategy to charge the capacitors to the maximum and use them only partially; a very low value (-200 g) corresponds to always using the capacitors up to complete discharge, charging them only with regenerative braking. The optimal solution ( $\lambda_0 = 103$  g), which gives a slightly better fuel consumption, recuperates more braking energy. This is visible in the power split for the three cases, shown in Figure 4.36 (which is limited to a fraction of the time horizon, for clarity). In general, the optimal solution found using Pontryagin's principle consists in charging the capacitors using the entire regenerative braking power, then discharge them fully during the subsequent acceleration phase, and start using the genset when the capacitors are discharged. This is true also for the other driving cycles examined. The optimal value of  $\lambda_0$  is the same in all driving cycles examined, despite the fact that they are sensibly different from each other.

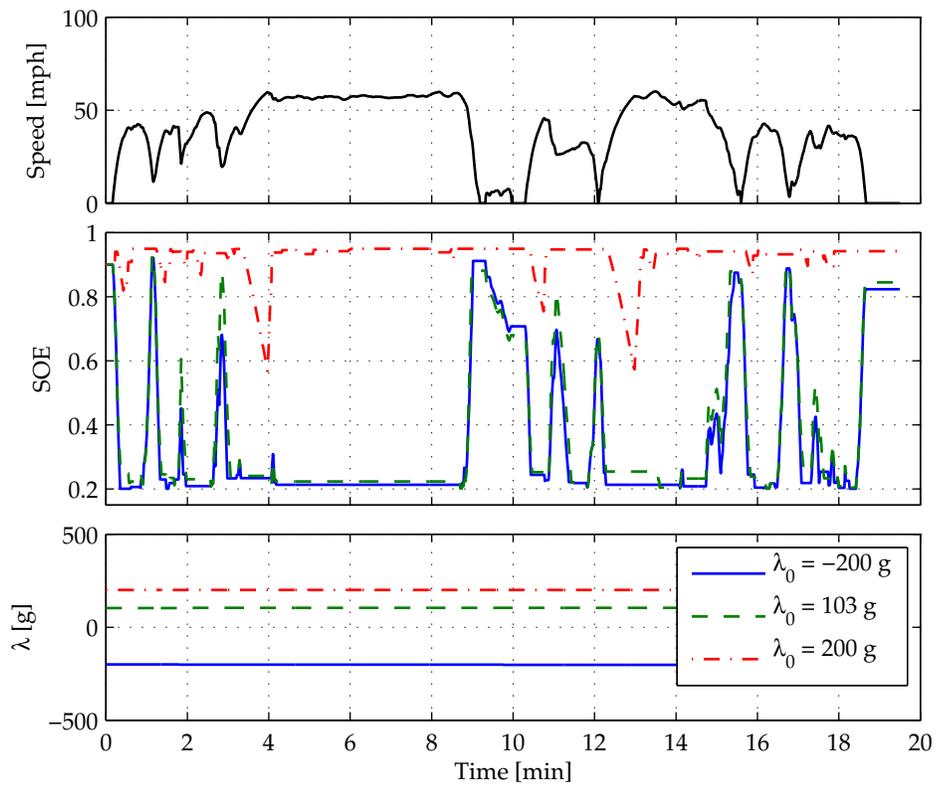


Figure 4.35: Application of the minimum principle to the refuse truck: effect of  $\lambda_0$  on SOE profile (cycle *Approach*)

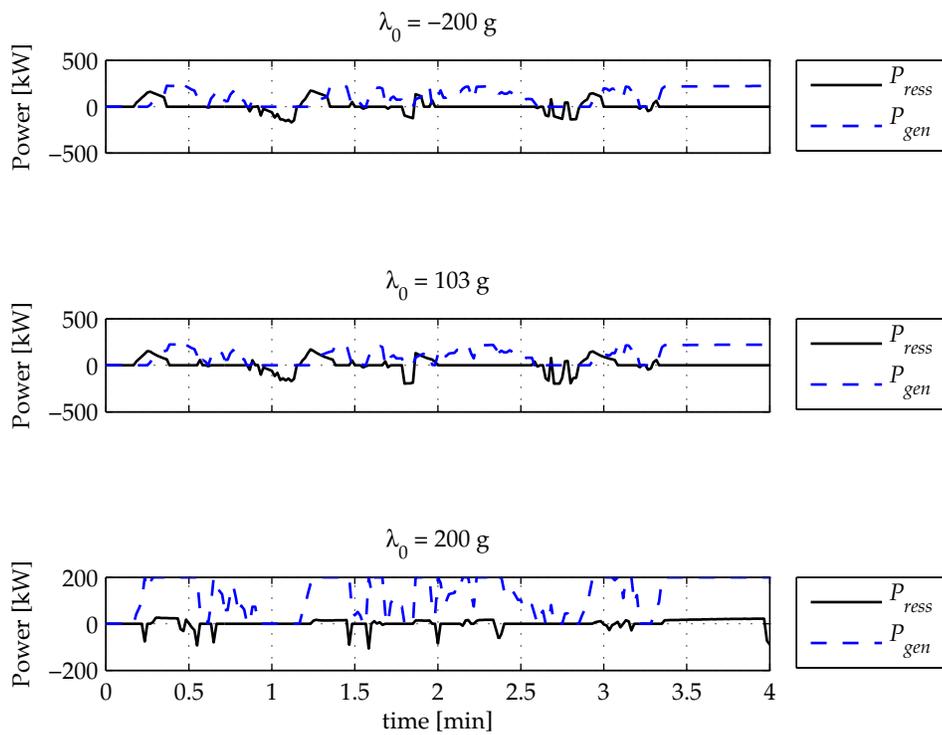


Figure 4.36: Application of the minimum principle to the refuse truck: effect of  $\lambda_0$  on power split (cycle *Approach*)

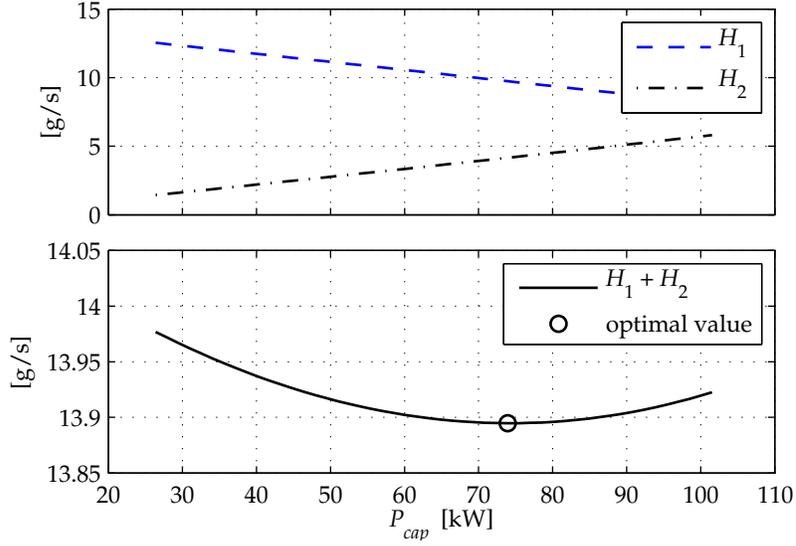


Figure 4.37: Visualization of the Hamiltonian at one instant ( $t = 628$  s in driving cycle *Approach*)

A way to illustrate the minimization of the Hamiltonian is to visualize it at a given instant, as done in Figure 4.37. In the figure,  $H_1$  and  $H_2$  indicate the two components of the Hamiltonian function (4.75):  $H_1 = \dot{m}_f(P_{cap}, t)$  and  $H_2 = -\lambda(t)\varepsilon_{cap}(\zeta, P_{cap})\frac{P_{cap}(t)}{E_{cap}}$  (the third component, the penalty due to reaching the SOE limits, is zero because the constraints are not active). Both terms are plotted as a function of the admissible control values, i.e. the range of capacitor power compatible with the current operating conditions.  $H_1(P_{cap})$  and  $H_2(P_{cap})$  appear nearly linear, but in fact they are not, because of the nonlinearity introduced by  $\varepsilon_{cap}(P_{cap}, \zeta)$  in  $H_2(P_{cap})$ ; thus, their sum, i.e. the Hamiltonian itself, is a convex function which, in this example, has a minimum for  $P_{ress} = 73$  kW. In many other instances, the nonlinearity of the Hamiltonian is extremely small and its minimum corresponds to one of the extreme values of the RESS power.

## 4.8.2 Simulation results, EcoCAR, charge-sustaining

The same procedure is applied to case study 2 as well. In the cycle UDDS, the effect of  $\lambda_0$  on the corrected fuel consumption is reported in Figure 4.38. The effect on the SOE variation during the cycle (synthetically represented by the quantity  $\Delta\zeta = \zeta(t_f) - \zeta(t_0)$ ) is shown in Figure 4.39. The solution is very sensitive to variations of  $\lambda_0$  and only a very narrow range of values give rise to a solution in which the final state of energy is close to the desired value (i.e.  $\Delta\zeta = 0$ ). The value of  $\lambda_0$  corresponding to the lowest value of  $|\Delta\zeta|$  is selected as the optimal value. Note that, since this point corresponds to the condition  $\Delta\zeta \approx 0$ , the actual and corrected fuel consumption in Figure 4.38 coincide. In general, since in this case the size of the energy buffer is much higher than in the refuse truck, the difference between the actual fuel consumption and the value corrected to account for SOE variation is important, and the effect of the final SOE variation is much more noticeable on the actual fuel consumption than it is on the corrected value. It is still clearly visible that there exists an optimal value of  $\lambda_0$ , which in this case is much higher than in the previous case study, thus suggesting a relation between the size of the energy buffer and the optimal co-state value. The effect of  $\lambda_0$  on the SOE profile and the power split is presented in Figure 4.40 and Figure 4.41 respectively.

The qualitative effect of  $\lambda_0$  on the solution is the same in all the driving cycles examined, but the optimal value varies slightly for each driving cycle, unlike the case of the refuse truck. In fact, in this case the charge-sustainability condition  $\Delta\zeta \approx 0$  is important and this introduces a difference between cycles with different energy characteristics (in particular, amount of potential regenerative braking relative to the total tractive energy necessary to follow the cycle).

## 4.9 ECMS

### 4.9.1 Basic formulation

Solving the problem using ECMS means defining an equivalent consumption function, and minimize it at every instant of time. The definition of equivalent fuel consumption is (from Section 3.5):

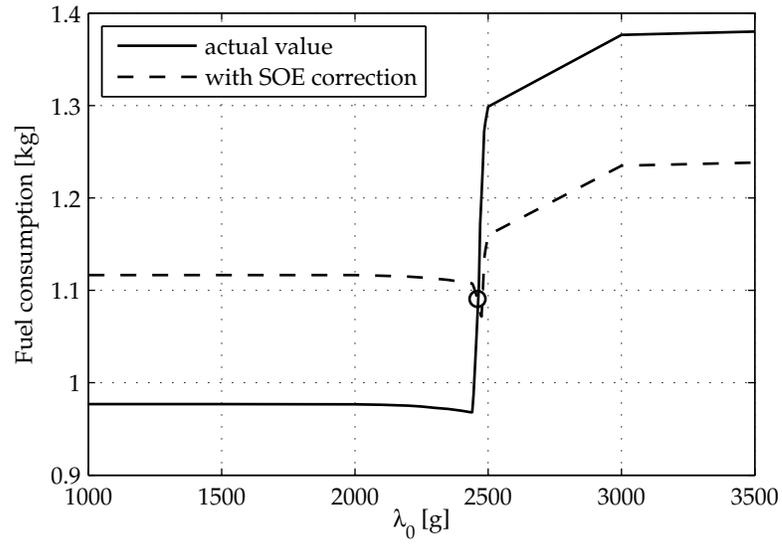


Figure 4.38: Application of the minimum principle to the EcoCAR: effect of  $\lambda_0$  on fuel consumption (cycle *UDDS*)

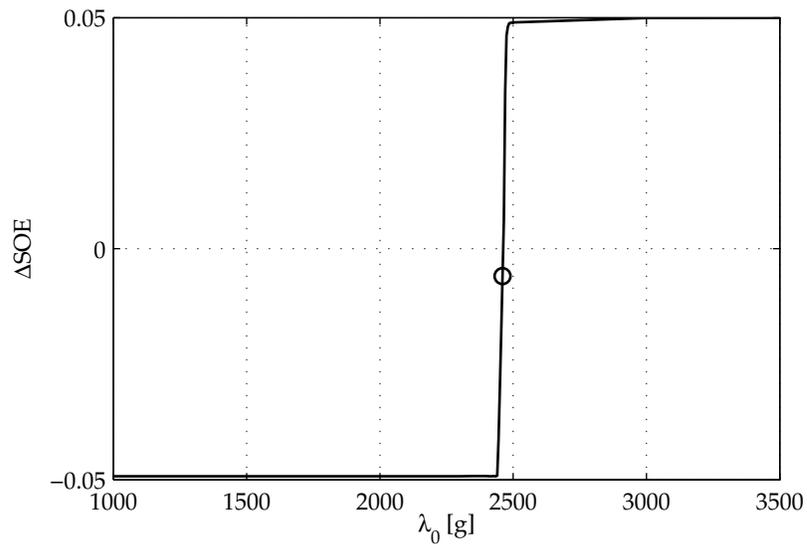


Figure 4.39: Application of the minimum principle to the EcoCAR: effect of  $\lambda_0$  on SOE variation, defined as  $\Delta SOE = SOE(t_f) - SOE(t_0)$  (cycle *UDDS*)

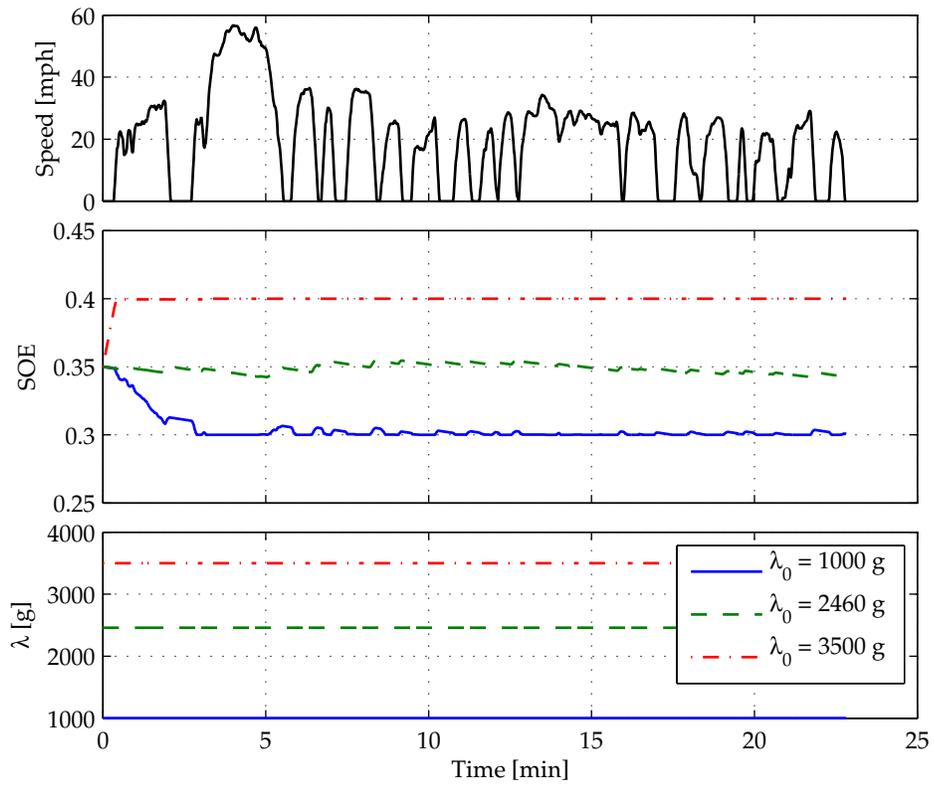


Figure 4.40: Application of the minimum principle to the EcoCAR: effect of  $\lambda_0$  on SOE profile (cycle *UDDS*)

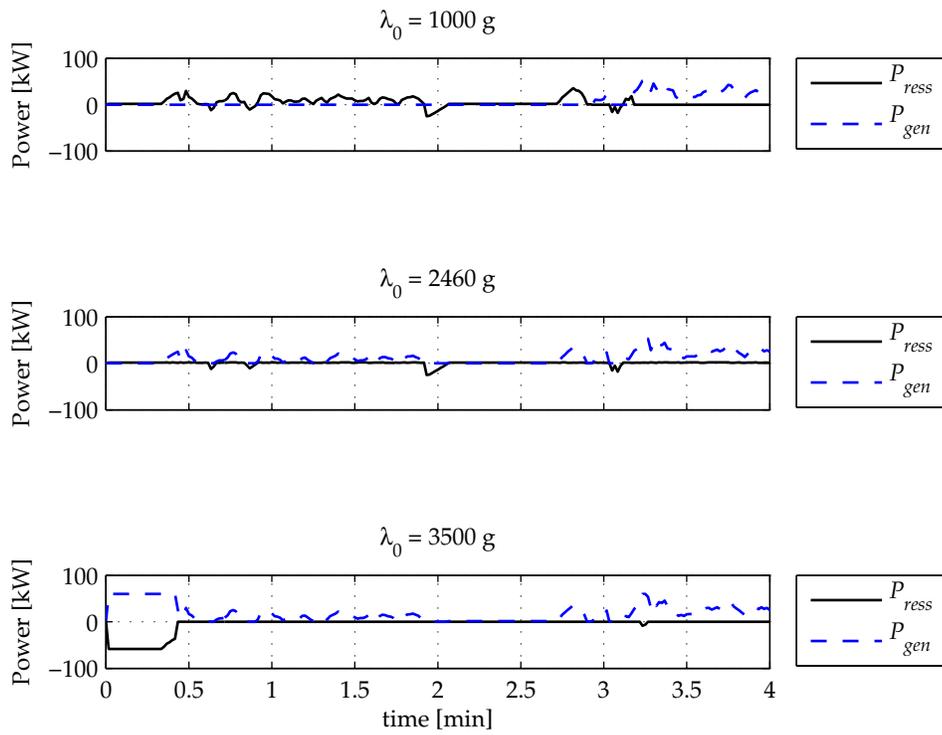


Figure 4.41: Application of the minimum principle to the EcoCAR: effect of  $\lambda_0$  on power split (cycle *UDDS*)

$$\dot{m}_{eqv}(P_{ress}(t), \zeta(t), t) = \dot{m}_f(P_{ress}(t), t) + \dot{m}_{ress}(P_{ress}(t)), \quad (4.79)$$

which can be rewritten, putting  $\dot{m}_{ress}$  in explicit form, as

$$\dot{m}_{eqv}(P_{ress}(t), \zeta(t), t) = \dot{m}_f(P_{ress}(t), t) + s \cdot p(\zeta(t)) \cdot \frac{1}{Q_{lhv}} P_{ress}(t) \quad (4.80)$$

where:

- $\dot{m}_f(P_{ress}(t), t)$  is the engine fuel consumption, and depends on the net power delivered by the genset, which is the difference of the total power request and the RESS power: since the total power request is a time-varying parameter independent from the optimization algorithm, its effect is the same as an explicit time dependence of  $\dot{m}_f$ ;
- $s$  is the equivalence factor, different for charge and discharge:  $s = \begin{cases} s_{dis} & \text{if } P_{ress} \geq 0 \\ s_{ch} & \text{if } P_{ress} < 0 \end{cases}$ ;
- $p(\zeta(t))$  is a penalty function that is used to keep the state of the system  $\zeta(t)$  within its boundaries.

At each time, the equivalent fuel consumption is calculated using (4.80) for several candidate values of the control variable  $P_{ress}(t)$ ; the value that gives the lowest equivalent fuel consumption is selected.

The element that constitute (4.80) are examined in detail in the following sections.

#### 4.9.2 Admissible control set

At each time  $t$ , the admissible range of control values is calculated using (4.59); the interval between the minimum and maximum admissible values is subdivided into a finite number of values  $P_{cap}^i(t)$  ( $i = 1, \dots, N_u$ ), and the equivalent fuel consumption is calculated for each of these values according to (4.80). The admissible control set is thus

$$P_{ress}^i(t) = P_{ress,inf}(t) + (i - 1) \frac{P_{ress,sup}(t) - P_{ress,inf}(t)}{N_u - 1}, \quad i = 1, \dots, N_u \quad (4.81)$$

where the lowest admissible value  $P_{ress,inf}(t)$  and the highest admissible value  $P_{ress,sup}(t)$  are calculated based on component limitations and current operating conditions according to (4.31) and (4.59).

### 4.9.3 Fuel consumption

The fuel consumption is calculated as a function of the instantaneous power demand, accessory load etc. using (4.45) and (4.60) for the two case studies respectively. The fuel consumption is calculated for each of the admissible control values  $P_{ress}^i$  in the range  $[P_{ress,inf}, P_{ress,sup}]$ .

### 4.9.4 Penalty function and difference between charge-sustaining and charge-depleting case

The penalty function is defined by (3.38), and is rewritten here in terms of the state variable  $\zeta(t)$ :

$$p(\zeta) = \begin{cases} \left(1 + \left(\frac{\zeta_{ref} - \zeta(t)}{\zeta_{ref} - \zeta_{min}}\right)^{2n_{p1}+1}\right) & \text{if } \zeta(t) < \zeta_{ref} \\ \left(1 - \left(\frac{\zeta(t) - \zeta_{ref}}{\zeta_{max} - \zeta_{ref}}\right)^{2n_{p2}+1}\right) & \text{if } \zeta(t) \geq \zeta_{ref} \end{cases} \quad (4.82)$$

where  $\zeta(t)$  is the instantaneous value of state of energy,  $\zeta_{ref}$  is the desired nominal value, and  $\zeta_{min}$ ,  $\zeta_{max}$  are the minimum and maximum admissible values;  $n_{p1}$  and  $n_{p2}$  are integer numbers. This function is represented as in Figure 4.42.

The correction function  $p(\zeta)$  defined in this way multiplies the equivalence factors and artificially increases or decreases their value near the boundaries of the desired SOC interval. The definition of  $p(\zeta)$  changes depending on whether the value of  $\zeta(t)$  is above or below  $\zeta_{ref}$ , thus allowing for asymmetric penalization of the state of energy, which is useful for keeping the average value at a level closer to one of the boundaries.

The definition of the penalty function (4.82) keeps the state of energy around a constant reference value. This is the most common case in a charge-sustaining hybrid, in which the battery state of energy should be maintained around a constant value.

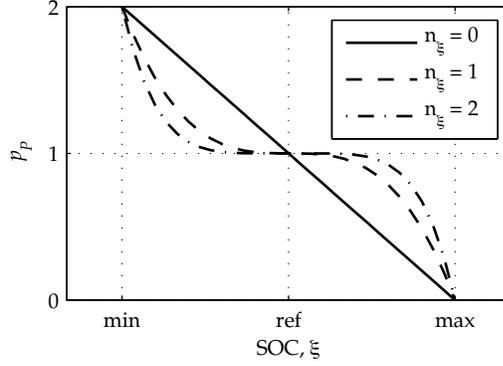


Figure 4.42: Representation of (4.82) in the case in which the penalty is symmetrical, i.e.  $\zeta_{ref} = \frac{1}{2}(\zeta_{min} + \zeta_{max})$  and  $n_{\zeta 1} = n_{\zeta 2} = n_{\zeta}$

In the charge-depleting case, there is no constant value to be tracked; rather, one can define a target SOE profile to be tracked during the driving cycle (perhaps using the results of dynamic programming optimization to define the optimal discharge profile), defining a time-varying<sup>4</sup>  $\zeta_{ref}(t)$  in (4.82). An alternative approach is to not define a penalty function (i.e., leave its value to 1) and let the SOE decrease freely in the acceptable range. Note that, even in the plug-in vehicles, the charge-depleting phase is followed by a charge-sustaining phase when a low SOE level is reached.

#### 4.9.5 Equivalence factors

The equivalence factors  $s_{chg}$  and  $s_{dis}$  that appear in (4.80) are constant in the standard ECMS implementation. However, their optimal value, which minimizes the total fuel consumption while maintaining the vehicle substantially charge sustaining, depends on the specific driving cycle. Similar cycles have similar values of optimal equivalence factors. Given the driving cycle, the optimal equivalence factors can be found by a search procedure or an extensive set of simulations in

<sup>4</sup>In fact, it should be dependent on distance traveled, not time, but the practical effect is the same

which several pairs of equivalence factors are compared. An example of this procedure is presented in the following sections, in which simulation results relative to the two case studies are presented.

#### 4.9.6 Simulation results, refuse truck

The first task to be performed when implementing ECMS is to define the values of equivalence factors most appropriate for the driving cycle. In order to do this, the procedure of instantaneous minimization is repeated for several values of the equivalence factors  $s_{dis}$  and  $s_{chg}$ , with the objective to find the best pair. In this case study, since the charge-sustainability is not an issue (in the sense that the final value of SOE is not important), the best pair of equivalence factors is the one that minimizes fuel consumption (including the correction term due to SOE variation). The analysis is presented for the cycle *Approach*, but the results are similar for all the driving cycles.

The effect of the equivalence factors on the solution calculated by the ECMS is visible in Figure 4.43, which shows how changing the equivalence factors modifies the SOE profile. The corresponding fuel consumption is visible in Figure 4.44. The solution with  $s_{dis} = 2.5$ ,  $s_{chg} = 1.5$  (case B) is such that the capacitors are almost never used (the SOE remains at its initial value), and gives the highest fuel consumption. Varying only the discharge factor from 2.5 to 1.5 (case D) changes completely the behavior and increases greatly the use of the capacitors, reducing the fuel consumption.

To examine the behavior of a large number of equivalence factors combinations, it is possible to visualize the fuel consumption and the SOE variation, defined as  $\Delta\zeta = \zeta(t_f) - \zeta(t_0)$ , for all the combinations of equivalence factors in a given range. This information is shown Figure 4.44 and Figure 4.45. From observation of the surface plots, it appears that the lowest fuel consumption corresponds to very small values of the both equivalence factors. The same procedure is repeated for the other cycles, obtaining the optimal pairs of equivalence factors shown in Table 4.6: in fact, the same values work for all the driving cycles.

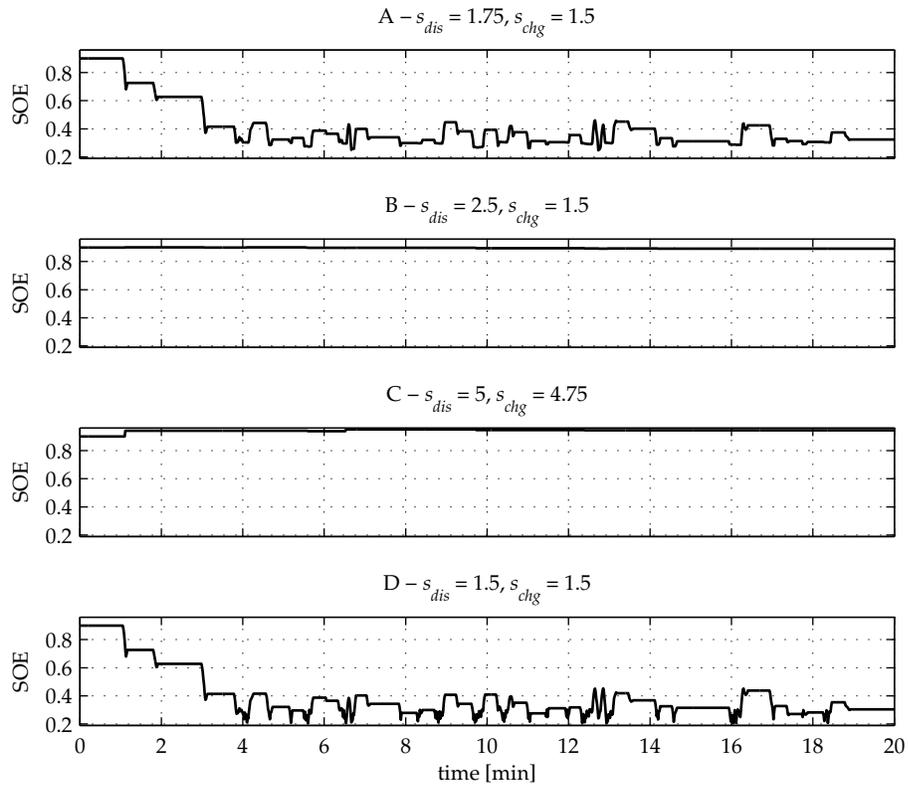


Figure 4.43: Application of ECMS to the refuse truck: SOE profiles obtained for four combinations of equivalence factors (cycle *Approach*)

Table 4.6: Fuel consumption and best equivalence factors for case study 1

Cycle	$\Delta SOE$	$s_{dis}$	$s_{chg}$
Approach	-0.04	1.5	1.5
Route 1	-0.46	1.5	1.5
Route 2	-0.66	1.5	1.5
Route 3	-0.65	1.5	1.5
Return	-0.01	1.5	1.5

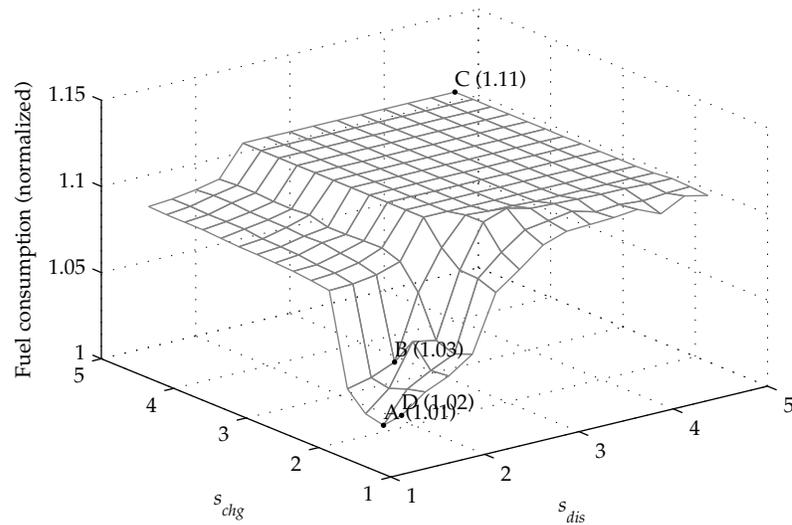


Figure 4.44: Effect of equivalence factors on fuel consumption for the Refuse truck, cycle *Approach*. Fuel consumption values are corrected including the effect of SOE variation, and normalized with respect to the optimal solution obtained with dynamic programming. The points correspond to the cases shown in Figure 4.43

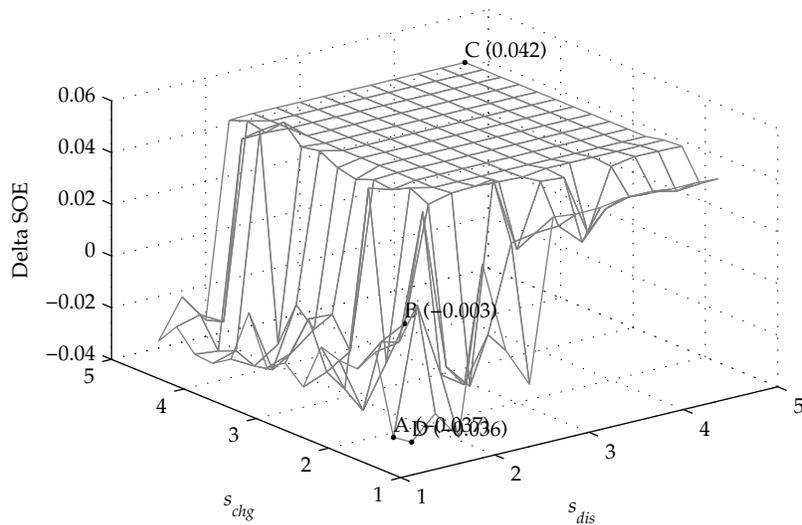


Figure 4.45: Effect of equivalence factors on SOE variation (positive values indicate an increasing trend, negative values a decreasing one) (cycle *Approach*)

#### 4.9.7 Simulation results, EcoCAR, charge-sustaining

In the charge-sustaining case, ECMS should minimize the fuel consumption while maintaining the state of charge (or state of energy) of the battery around a nominal value. The penalty function is helpful in this respect, but the combination of equivalence factors can change the behavior of the strategy with respect to the SOE profile as well as the fuel consumption. In order to investigate the effect of the equivalence factors, a set of simulations were run with various combinations of  $s_{chg}$  and  $s_{dis}$  for the three driving cycles considered in this case study.

The *UDDS* cycle is considered. The velocity profile and corresponding power request are shown in Figure 4.46. The ECMS solution, in terms of battery power and consequent SOE profile, is shown in 4.47 for several combinations of  $s_{chg}$  and  $s_{dis}$ , arbitrarily chosen. It can be observed how the values of these factors have a great effect on the strategy behavior, as it is expected since they change the relative cost of the battery usage with respect to the fuel consumption. In particular, higher values of the discharge equivalence factor tend to discourage the battery use (making higher its cost in terms of future equivalent fuel consumption), while lower values make discharge easier. The dual situation is true for the charge equivalence factor.

The effect of the equivalence factors can be quantified and visualized by simulating all their combinations in a given range. Figure 4.48 shows the effect of the equivalence factors on fuel consumption. For a more fair comparison, the corrected fuel consumption defined in Section 4.6.3 is reported. The values are normalized with respect to the optimal result obtained with dynamic programming. As it is evident, there exists a relatively wide range of values of equivalence factors that generate a solution very close to the optimum. This is a qualitative difference between Figure 4.48 and the corresponding surface for the refuse truck in Figure 4.44, in which the minimum is localized in a smaller region.

Figure 4.49 shows the effect of the equivalence factors on the state of energy trend, expressed using the difference between the final and initial state of energy. The plane  $(s_{dis}, s_{chg})$  can be subdivided in regions with different behavior in this

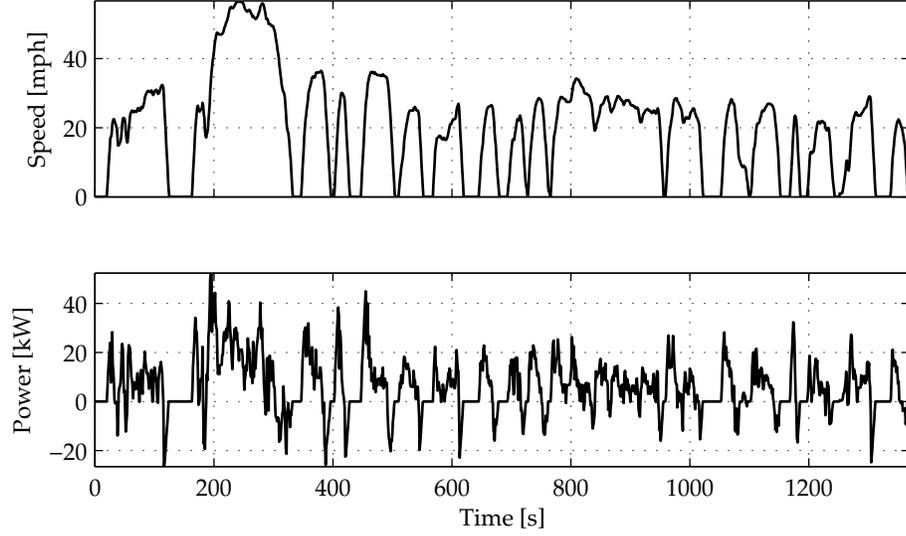


Figure 4.46: Velocity profile and power requests of UDDS cycle

respect: in particular, only a subset of the plane generates a trend close to zero, which indicates charge-sustainability.

Using these maps it is possible to select the most appropriate values of equivalence factors for this cycle as  $s_{dis} = 4.25$ ,  $s_{chg} = 1.75$ . The corresponding results are presented in Figure 4.50 and Figure 4.51 (detail of the previous). It can be noted how the strategy makes a large use of the battery in discharge, using only the battery to propel the vehicle; and how it tries to charge the battery using the genset at full load during regenerative braking phase, in order to maximize the recharge. This behavior is due to the much larger value of  $s_{chg}$  with respect to  $s_{dis}$ . The optimal values of the equivalence factors for this vehicle are shown in Table 4.7.

To better understand how the instantaneous minimization works, it is possible to show the values of the terms  $\dot{m}_f$  and  $\dot{m}_{elec} = s \cdot p(\zeta(t)) \cdot \frac{1}{Q_{lhw}} P_{ress}(t)$  appearing in (4.80) at one particular instant. This is done in Figure 4.52, which refers to an arbitrary time instant, in which the power request is  $P_{req} = 23.7$  kW, and the admissible range of values for the battery power is  $[-36.1$  kW,  $23.7$  kW] (according to (4.59)). The values of  $\dot{m}_f$  and  $\dot{m}_{ress}$  for this range of battery power are shown

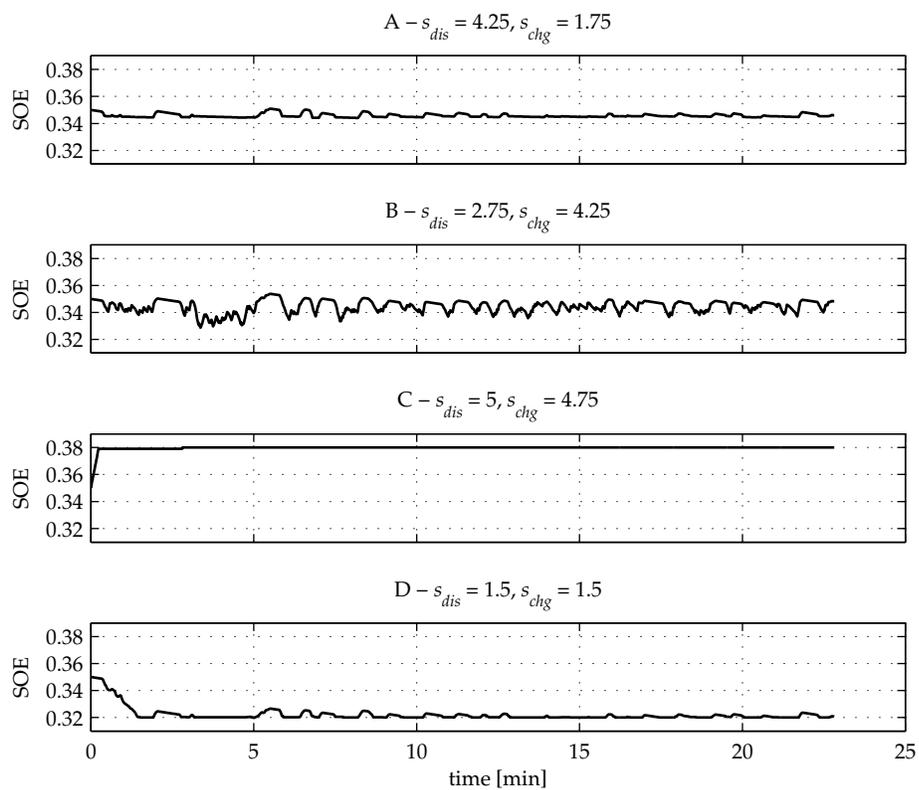


Figure 4.47: SOE profile in UDDS cycle, for various combinations of equivalence factors (each letter represent a pair  $s_{dis}, s_{chg}$  and the respective values are shown in Figure 4.48)

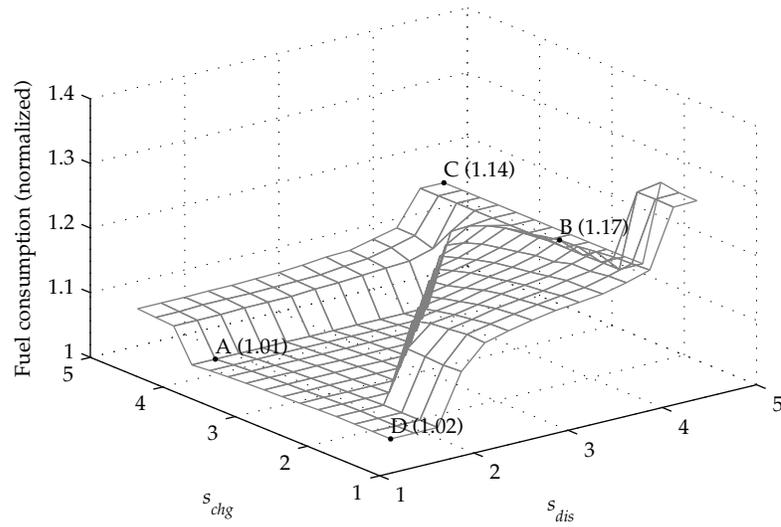


Figure 4.48: Effect of equivalence factors on fuel consumption (fuel consumption is corrected to account for SOE variation and the result is normalized with respect to the optimal solution obtained with dynamic programming)

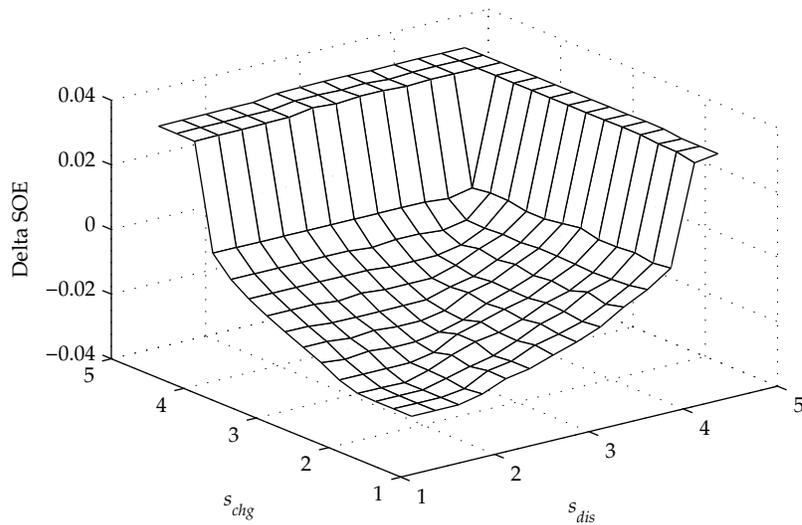


Figure 4.49: Effect of equivalence factors on SOE variation (positive values indicate an increasing trend, negative values a decreasing one)

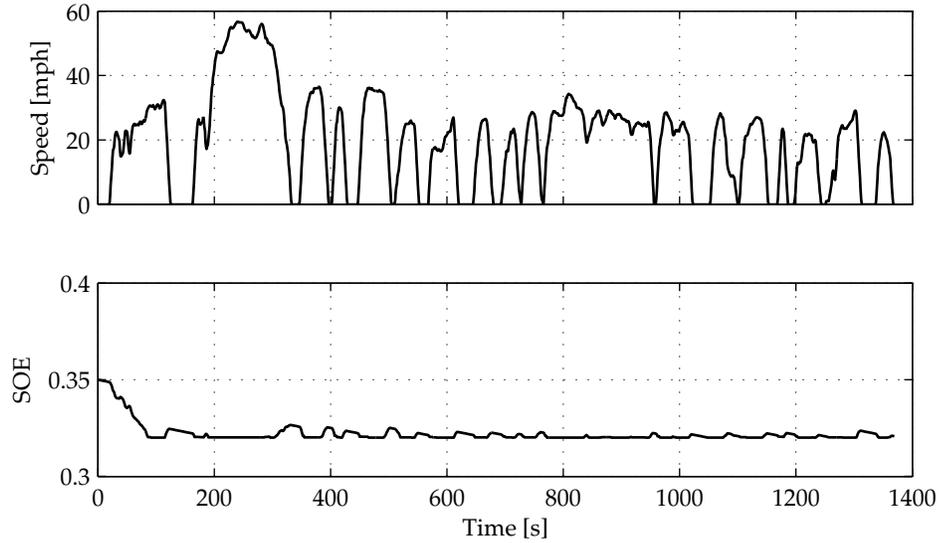


Figure 4.50: Results of ECMS implementation on UDDS cycle ( $s_{dis} = 1.75$ ,  $s_{chg} = 4.25$ )

in Figure 4.52: the engine fuel consumption  $\dot{m}_f$  increases as the battery power decreases, obviously, since the engine must compensate for the lower availability of battery power (or provide the power needed to charge it when  $P_{batt}$  is negative). On the other hand, the electric-equivalent fuel consumption  $\dot{m}_{ress}$  increases with increasing battery power. The different slopes of the curve  $\dot{m}_{ress}(P_{ress})$  for positive or negative values of  $P_{ress}$  are due to the different equivalence factors.  $\dot{m}_{ress}$  is negative when the RESS is being charged, and positive when it is being discharged. The resulting equivalent fuel consumption  $\dot{m}_{eqv}$  is the sum of the two and, in this case, it is minimized when the battery power assumes a value of zero.

## 4.10 Strategy comparison

In this section, the outputs of each of the strategies implemented are compared to each other to point out the similarities and differences. The EcoCAR case study is considered. The solutions obtained with the three strategies for the cycle UDDS are shown in Figure 4.53 (entire SOE profile) and figures 4.54 and 4.55 (detail of

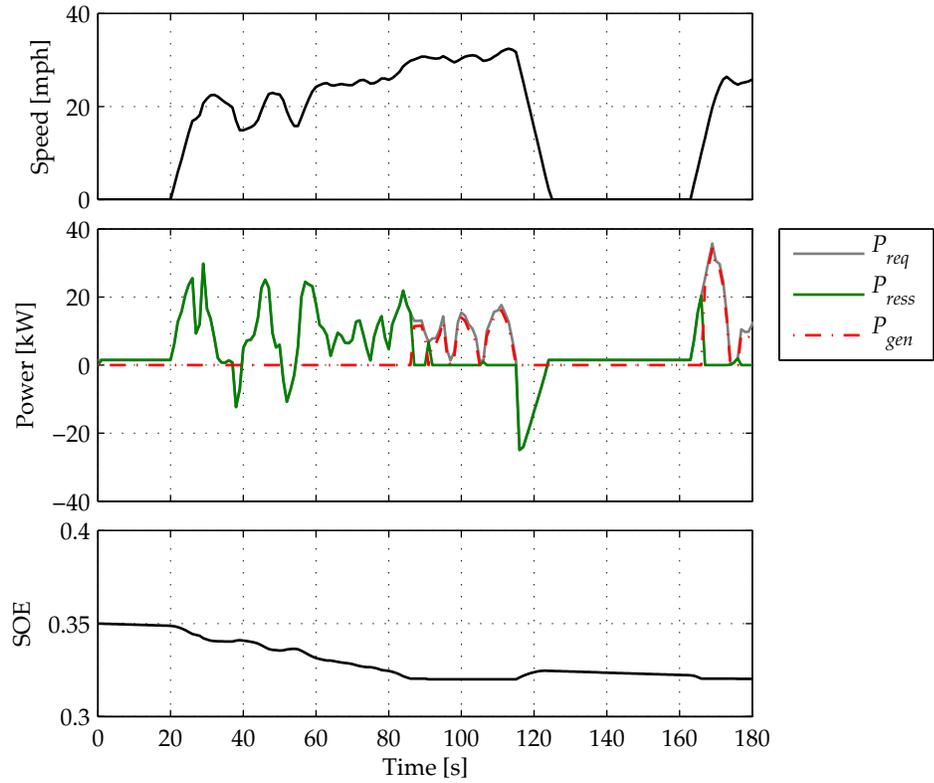


Figure 4.51: Results of ECMS implementation on UDDS cycle, detail of Figure 4.50

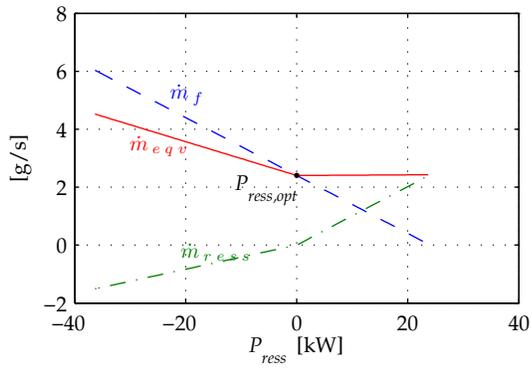


Figure 4.52: The terms of the ECMS instantaneous cost function at one instant

Table 4.7: Fuel consumption and best equivalence factors for case study 2

Cycle	$\Delta SOE$	$s_{dis}$	$s_{chg}$
UDDS	-0.03	4.25	1.75
US 06	-0.01	4.25	2.75
FTP highway	+0.02	5	3

power split in two separate time windows). The dynamic programming solution is, as expected, the one that generates the lowest total cost and is regarded as the benchmark optimal solution; the solution obtained with Pontryagin’s minimum principle is very close to the dynamic programming solution, almost identical, as it is expected from the theory. The ECMS solution, using opportune equivalence factors, is also very close to the others, at least in terms of fuel consumption, even if the SOE profile appears different. The solution obtained with ECMS shows some apparent differences with respect to the others: the SOE profile, shown in Figure 4.53, does not show any long-term trend, unlike the other two solutions, in which the SOE first increases and then decreases, with local ups and down due to acceleration and braking. The long-term oscillations are due to the different properties of the driving cycle in the initial region (higher average power demand, as seen in Figure 4.20), which means that it is optimal to charge the battery from  $t = 300$  s to  $t = 500$  s in order to use it later. ECMS, being a local optimization strategy, does not account for this kind of information regarding the global driving cycle behavior, and instead discharges the battery during acceleration and recharges it using regenerative braking. This is clearly visible in Figure 4.54, where the power split is shown: while dynamic programming (DP) and Pontryagin’s minimum principle (PMP) tend to discharge the battery at low power for an extended period, always supplementing the battery with the generator, ECMS uses only the battery initially, then, when this reaches a low SOE, it stops using it and switches to the generator. This is the effect of the instantaneous minimization approach, in which the future driving conditions are not known and are taken into account only indirectly, and approximately, by the use of an optimized equivalence factor.

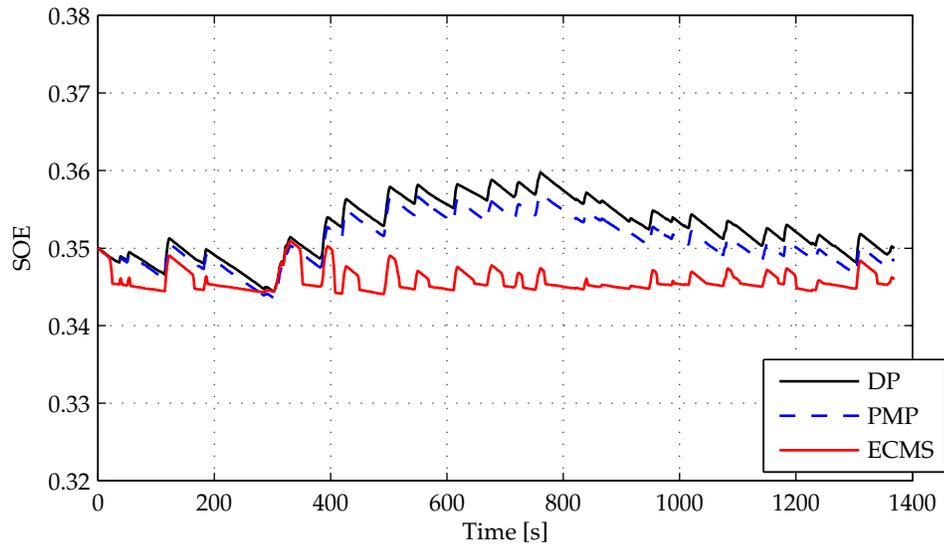


Figure 4.53: Comparison of the SOE profile obtained with the three strategies for the EcoCAR case, cycle UDDS.

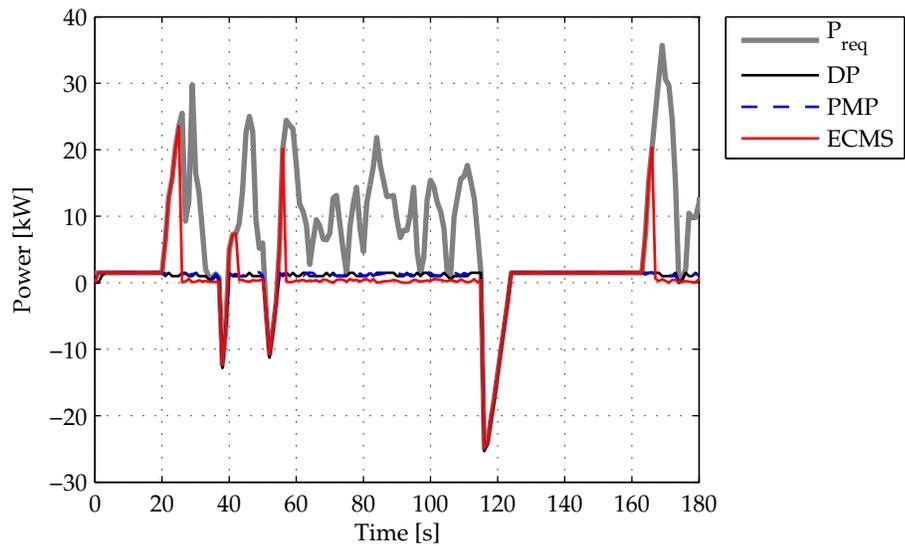


Figure 4.54: Comparison of the RESS power obtained with the three strategies for the EcoCAR case, cycle UDDS, detail #1.

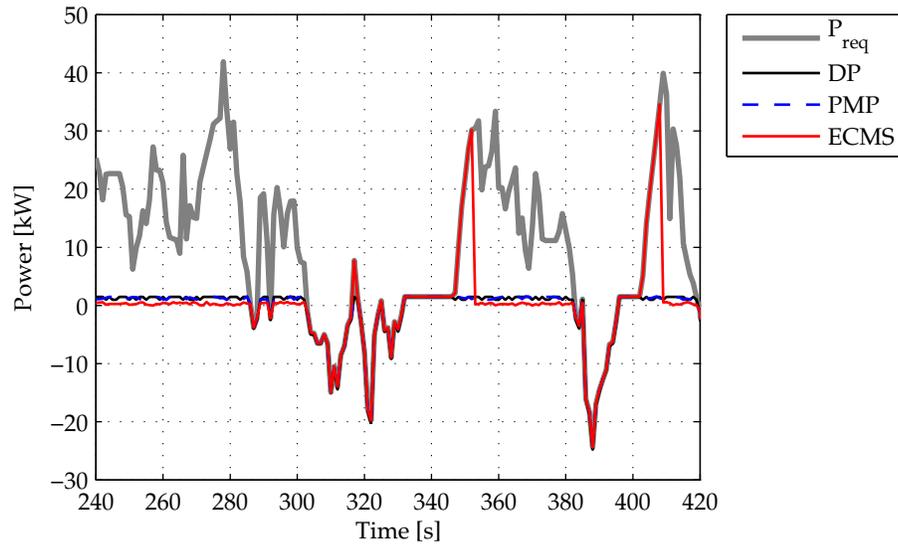


Figure 4.55: Comparison of the RESS power obtained with the three strategies for the EcoCAR case, cycle UDDS, detail #2.

Table 4.8: Fuel consumption for the three strategies, case study EcoCAR. All values are normalized with respect to the DP solution

Driving cycle	DP	PMP	ECMS
UDDS	1	1.000	1.017
US06	1	1.001	1.017
FTP highway	1	1.000	1.009

Table 4.9: Fuel consumption for the three strategies, case study refuse truck. All values are normalized with respect to the DP solution

Driving cycle	DP	PMP	ECMS
Approach	1	1.005	1.021
Route 1	1	1.003	1.033
Route 2	1	1.005	1.030
Route 3	1	1.005	1.035
Return	1	1.008	1.076

Dynamic programming and the minimum principle, on the other hand, are very close to each other. Their similarity derives directly from the optimal control theory: when the Hamiltonian of the system is a convex function, with a unique minimum in the admissible control set, the necessary conditions given by the minimum principle (§4.8) are also sufficient conditions for optimality, therefore the only solution that is obtained using the conditions is the optimal solution, and must correspond to the optimal solution found with dynamic programming. This is true assuming that the problem has a unique optimal solution. The existence of the solution has not been shown formally, but is intuitively true: being this a physical system, there must be one way of splitting the power request between the generator and RESS in such a way that the fuel consumption is minimized. The uniqueness of this power split sequence, however, is not guaranteed, as there may be several solutions that lead to the same result. In this case, the convexity of the Hamiltonian function means that there is only one solution satisfying the minimum principle, and therefore the optimal solution is unique. The minor differences between the solutions obtained with DP and PMP can be attributed to the different approximation levels of the two cases. DP is based on a discretized SOE grid and therefore the RESS power can only assume one of a finite number of values, which depend on the number of SOE points considered in the grid. Because of limitations in the available memory and computational time, the power discretization in DP is in the order of 1 kW. For Pontryagin's principle, there is no state discretization; however, the Hamiltonian is evaluated at a finite number of points (in terms of RESS power) and one of them is chosen as the optimal solution at each instant. The number of points is such that the power discretization is about 1 kW in this case as well, but clearly the numerical values of power in the two cases are different.

The computational time required by DP and PMP is a major difference between the two: PMP requires the optimization of the co-state variable using an iterative method, but the procedure is completed in 5-10 minutes (for a driving cycle lasting 20 minutes, discretized at 1 second intervals); in the same driving cycle, computing the optimal solution with DP, even using the optimized procedure described in

Section 4.7.3, requires several hours. Given the identity between the two methods (when the conditions given at the beginning of this section are satisfied), it is clear that PMP allows to obtain the optimal solution, to be used as a benchmark, much faster. Both strategies work only off-line because they require knowledge of the entire driving cycle.

## 4.11 ECMS as an implementable quasi-optimal strategy: i-ECMS

ECMS was originally proposed as a strategy implementable on-line without a-priori knowledge of the driving cycle. In fact, this is not completely true, since it does require appropriate tuning of the equivalence factors using iterative simulations, just like PMP. Unlike PMP, however, it is more robust to change in the tuning parameters, thanks to the presence of a penalty function on SOE.

As formally shown in Section 3.6, ECMS can represent a realization of Pontryagin's minimum principle, and therefore it can be an optimal solution as well. The fact that the equivalence factors are constant and the presence of the penalty function introduce some differences with respect to the minimum principle and make ECMS sub-optimal, but only marginally, as shown by the simulation results in Section 4.10. The fact that ECMS is an implementation of the optimal solution obtained with the minimum principle is extremely important for at least two reasons: it means that it is possible to implement a causal strategy that is also formally optimal (or at least sub-optimal), and gives some tools that can be used for more effective tuning and implementation.

The way in which the equivalence between PMP and ECMS can be exploited is to use the correspondence between the Hamiltonian (4.75) and the ECMS equivalent fuel consumption (4.80) to relate the charge and discharge equivalence factors to the co-state. Assuming that the SOE is within the boundaries and thus neglecting the effect of the penalty function, the virtual fuel consumption of ECMS in (4.80) is equal to the term of the Hamiltonian that depends on the co-state:

$$H = \dot{m}_{eqv} \Leftrightarrow -\lambda(t)\varepsilon_{ress}(\zeta, P_{ress}) \frac{P_{ress}(t)}{E_{ress}} = s \cdot \frac{1}{Q_{lhv}} P_{ress}(t) \quad (4.83)$$

Therefore, the equivalence factor can be expressed, at each instant of time, as:

$$s(t) = -\lambda(t)\varepsilon_{ress}(\zeta, P_{ress}) \frac{Q_{lhv}}{E_{ress}} \quad (4.84)$$

The fact that  $\varepsilon_{ress} < 1$  for negative values of  $P_{ress}$  (charge) and  $\varepsilon_{ress} > 1$  for positive  $P_{ress}$  (discharge) generates the difference between the charge and discharge equivalence factors that has always been recognized in the literature on ECMS. In addition to this, (4.84) shows that the equivalence factor depends on the co-state  $\lambda(t)$ , on the actual battery efficiency at a given instant, which can be easily modeled using a map of  $\varepsilon_{ress}(\zeta, P_{ress})$ , and on the ratio of fuel energy content and RESS energy capacity. The latter two are constant vehicle parameters, while the co-state is a function of both vehicle and driving cycle. Despite being formally a time-varying function, it has been seen from the simulation results that  $\lambda(t)$  changes very slowly during a driving cycle, and can be approximated by its initial value  $\lambda_0$ .

If the equivalence factors in the ECMS formulation are expressed<sup>5</sup> by (4.84), using the assumption that  $\lambda(t) \simeq \lambda_0 \forall t$  and assuming to know the optimal value of  $\lambda_0$  for the specific cycle, then the results obtained with ECMS are almost identical to those achieved by PMP, as shown in figures 4.56, 4.57 and 4.58 (which are the equivalent of figures 4.53, 4.54 and 4.55). This “ideal ECMS”, or *i-ECMS*, can be regarded as an implementable solution that is potentially equivalent to the optimal solution, assuming a proper tuning of the parameter  $\lambda_0$ .

If a short-term prediction of the future driving conditions is available, for example using statistical methods combined with information about the recent driving conditions, then the iterative solution to Pontryagin’s minimum principle can be obtained for the prediction horizon, thus deriving the optimal value of  $\lambda_0$  for the current and near-future conditions. The PMP solution is not directly implementable, of course, because of the prediction inaccuracies and of the computational time that prevents a solution of the problem quickly enough to implement it as a receding horizon approach. However, the value of  $\lambda_0$  found in this way can be

<sup>5</sup>both charge and discharge equivalence factor have the same expression, since the term  $\varepsilon_{ress}(\zeta, P_{ress})$  accounts for the sign of the power

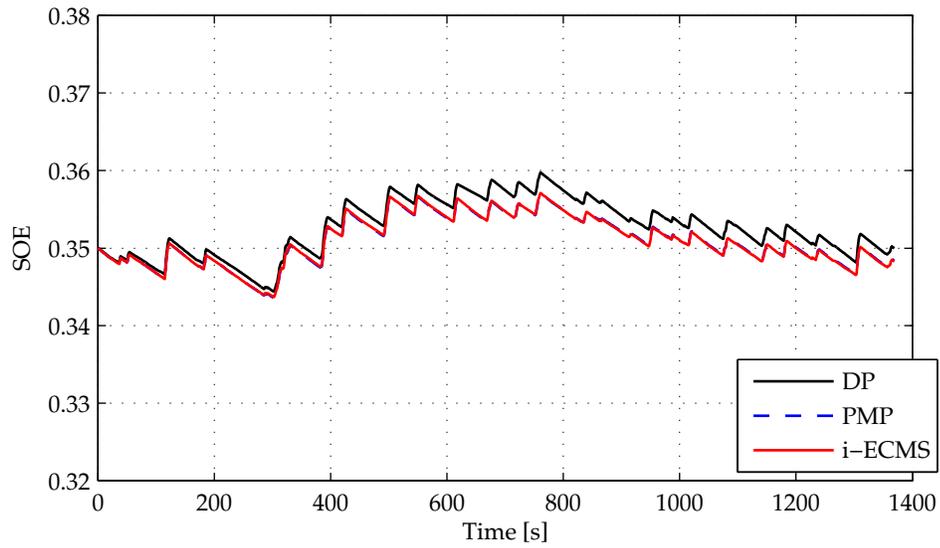


Figure 4.56: Comparison of the SOE profile obtained with the three strategies for the EcoCAR case, cycle UDDS.

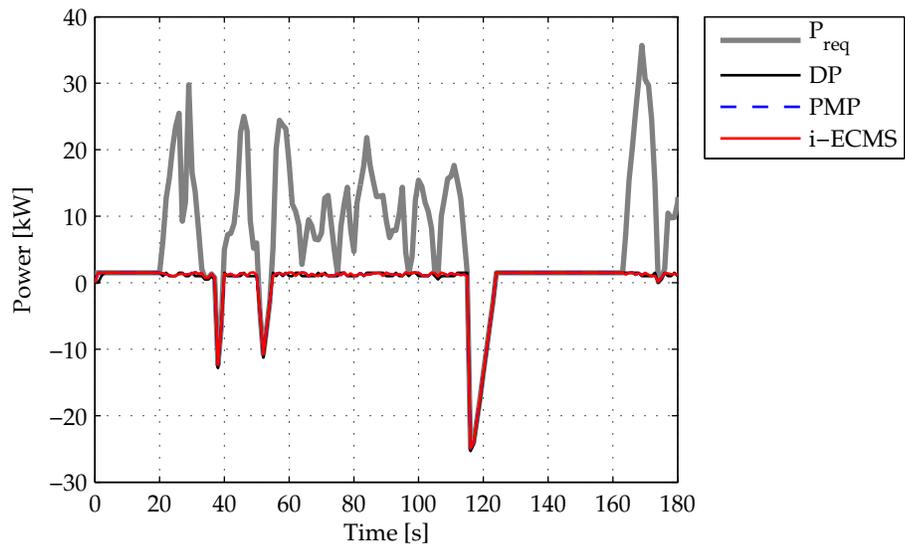


Figure 4.57: Comparison of the RESS power obtained with the three strategies for the EcoCAR case, cycle UDDS, detail #1.

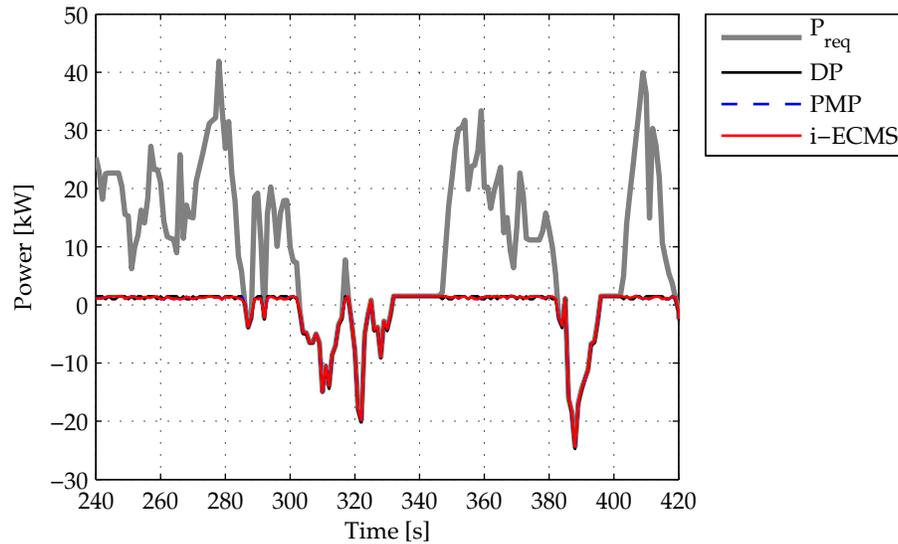


Figure 4.58: Comparison of the RESS power obtained with the three strategies for the EcoCAR case, cycle UDDS, detail #2.

used in (4.84) to estimate the equivalence factor of i-ECMS, which is then used to define the instantaneous value of equivalent fuel consumption. The frequency at which  $\lambda_0$  is updated, and therefore the frequency of adaptation of the equivalence factor, can be as slow as needed for the prediction algorithm and the PMP iterative solution to be executed, which can be estimated to be a few minutes, once the code is optimized for real-time implementation. In this way, the ideal ECMS introduced here is the foundation of an adaptive ECMS based on optimal control theory.

## 4.12 On the stability of ECMS and PMP

In order to be implementable on-line, energy management controllers need to satisfy some basic stability properties. In this section, qualitative considerations on the stability of the solution to Pontryagin's minimum principle and the ECMS (both conventional and ideal, or adaptive) are proposed. Dynamic programming, being a numerical optimization technique that solves the problem at once for the entire driving horizon, is not subject to stability issues.

The first point to be addressed is the definition of stability for an energy management strategy. In this context, a strategy is defined stable if it drives the system state around a reference value, or keeps it in the neighborhood of that value, in presence of a constant external input (power request). In other words, if the power request remains constant for an arbitrarily long time, the SOE should not drift toward the upper and lower bounds, but rather converge to its nominal (reference) value (indicated as  $\zeta_{ref}$  in (4.82)), or steadily oscillate around it. This definition is implicitly assuming the case of a charge-sustaining vehicle, but it can be adapted to a charge depleting vehicle as well, if the reference value is set to the average SOE at which the vehicle operates in charge-sustaining mode after the battery has been discharged. From a qualitative point of view, a HEV subject to constant power demand for an indefinite time is considered in the same way as an autonomous system, i.e., a system with no external input acting on it, which should remain in the vicinity of its equilibrium point: the equilibrium point is defined as the SOE taking its nominal (reference) value  $\zeta_{ref}$ . Intuitively, this equilibrium point is stable if the SOE does not drift away from the reference value, asymptotically stable if the SOE tends to move toward the reference value.

Stability, in this sense, is important for an energy management strategy because it implies the fact that the system state is kept at a value such that the correct operation of the hybrid powertrain is guaranteed (if the reference value is chosen appropriately). For example, if the SOE moved to the upper bound and remain there, there would be no possibility of regenerative braking, and therefore the results would be sensibly sub-optimal. The fact that the SOE tends to remain around the appropriate reference value, ensures the fully exploitation of the potential of the HEV system.

In formal terms, given the system dynamic equation (4.61)

$$\dot{\zeta}(t) = -\varepsilon_{ress}(\zeta, P_{ress}) \frac{P_{ress}(t)}{E_{ress}}, \quad (4.85)$$

and a constant external input  $P_{req}(t) = P_{req,0} \forall t \in [t_0, t_f]$ , the control sequence

$$P_{ress}(t) = \arg \min_{P_{ress}(t)} J(P_{ress}(t), \zeta(t), P_{req}(t)), \quad t \in [t_0, t_f] \quad (4.86)$$

is asymptotically stable if the state  $\zeta(t)$  tends to its reference value:  $\zeta(t) - \zeta_{ref} \rightarrow 0$  as  $t \rightarrow t_f$ . It is stable in the bounded-input, bounded-output (BIBO) sense if the state of the system remains bounded (i.e., it does not differ from the reference value for more than a tolerance  $\zeta_{tol}$ ) in presence of bounded input, that is, if there exists a maximum value  $P_0$  that the power request does not exceed:  $|P_{req}(t)| \leq P_0 \implies |\zeta(t) - \zeta_{ref}| \leq \zeta_{tol}$ .

A theoretical proof of the stability properties of ECMS and PMP is not among the objectives of this dissertation, however the implementation of these strategies is such that, by construction, BIBO stability is achieved. In particular, the power request  $P_{req}(t)$  is always a finite value that does not exceed the physical limitations of the electric machines, therefore the input is always bounded. The output, i.e. the state of energy, is also bounded in implementation because the limits on the control variable  $P_{ress}(t)$  are set in order to prevent the state to exceed the bounds  $\zeta_{min}$  and  $\zeta_{max}$ : if the SOE reaches its maximum admissible value, then the RESS can only be discharged, and, conversely, it can only be charged once SOE reaches the lower bound.

In PMP, the fact that the co-state “jumps” (i.e., it has a discontinuity) every time a bound is hit means that, in presence of constant power request, the SOE solution is either constant ( $P_{ress} = 0$ ) or steadily oscillating between the minimum and the maximum (the discriminating variable between these two cases are the value of power request and the efficiency characteristics of the engine and RESS [20, 36]).

In ECMS, an asymptotic stabilization effect is achieved with the penalty function (4.82), which modifies the local cost function (equivalent fuel consumption) so that, whenever the SOE deviates from the reference value, solutions that tend to move the SOE back to the reference value are encouraged (because they assume a lower cost). This tends to ensure a notion of practical stability in the sense described. The same correction is implemented in i-ECMS, and leads to the same result.

As an example, consider the case of ECMS with constant equivalence factors, and examine in detail the two components of the equivalent fuel consumption (4.80) (repeated here for convenience):

$$\dot{m}_{eqv}(P_{ress}, \zeta(t), t) = \dot{m}_f(P_{ress}, t) + s \cdot p(\zeta(t)) \cdot \frac{1}{Q_{lhv}} P_{ress}. \quad (4.87)$$

The first term is the fuel consumption expressed as a function of the control variable  $P_{ress}$ ; it depends explicitly on time in the general case in which the power request is time-varying. In the case of constant power request, it is a function only of the control:  $\dot{m}_f = \dot{m}_f(P_{ress})$ . The second term, i.e. the RESS virtual fuel consumption, is proportional to the RESS power through the factor  $s \cdot p(\zeta(t))$ . Therefore, during a period in which the power request is constant, the instantaneous cost  $\dot{m}_{eqv}$  (to be minimized with respect to  $P_{ress}$ ) is the sum of two terms:

- $\dot{m}_f(P_{ress})$ , which is invariant with time; and
- $\dot{m}_{ress}(P_{ress}, \zeta(t)) = s \cdot p(\zeta(t)) \cdot \frac{1}{Q_{lhv}} P_{ress}$ , which varies according to the SOE variation.

In the standard ECMS implementation, where  $s$  is constant with time (even if it takes different values for charge or discharge), the effect of  $\zeta(t)$  appears as a variation of slope in the curve  $\dot{m}_{ress}(P_{ress})$ , due to the factor  $p(\zeta)$ . In particular, the variation of slope with time can be expressed as

$$\frac{d}{dt}(s \cdot p(\zeta)) = s \cdot \frac{\partial}{\partial \zeta} p(\zeta) \cdot \dot{\zeta}. \quad (4.88)$$

Using the expression  $p(\zeta) = \left( \frac{\zeta(t) - \zeta_{ref}}{\zeta_{max} - \zeta_{min}} \right)^{2n_p + 1}$  for the penalty function<sup>6</sup>, the factor  $\frac{\partial}{\partial \zeta} p(\zeta)$  is

$$\frac{\partial}{\partial \zeta} p(\zeta) = (2n_p + 1) \left( \frac{\zeta(t) - \zeta_{ref}}{\zeta_{max} - \zeta_{min}} \right)^{2n_p} \quad (4.89)$$

and is always positive because  $n_p$  is a positive integer number. Therefore, since the equivalence factor is also positive, the sign of  $\frac{d}{dt}(s \cdot p(\zeta))$  is the same as the sign of  $\dot{\zeta}$ , which means that, as the state of energy increases, the slope of the curve  $\dot{m}_{ress}(P_{ress})$  increases as well. This has a stabilizing effect on the SOE (in the sense

<sup>6</sup>this is the same expression given in Section 3.5, and is identical to the version given by (4.82) if the reference SOE is the average between  $\zeta_{min}$  and  $\zeta_{max}$

explained at the beginning of this section), as can be seen by examining the possible cases:

1.  $\zeta(t) > \zeta_{ref}$ , increasing SOE (i.e.,  $\dot{\zeta} > 0$  and  $P_{ress,opt} < 0$ ). In this case, the curve  $\dot{m}_{ress}(P_{ress})$  is such that the optimal solution, corresponding to the minimum of  $\dot{m}_{eqv}(P_{ress})$ , is a negative value of  $P_{ress}$ . In order to move the SOE toward its reference value, it is necessary to reverse the SOE tendency, and changing the slope of the curve  $\dot{m}_{ress}(P_{ress})$  in such a way that the minimum of  $\dot{m}_{eqv}(P_{ress})$  is found at a positive value of  $P_{ress}$ . This is obtained by increasing the slope of  $\dot{m}_{ress}(P_{ress})$ , that is by increasing the value of  $s \cdot p(\zeta(t))$ , which is in fact happening because  $\text{sign}(s \cdot p(\zeta)) = \text{sign} \dot{\zeta}$ . The slope itself is proportional to  $p(\zeta)$ , that is to the difference between the current SOE value and the reference value.
2.  $\zeta(t) > \zeta_{ref}$ , decreasing SOE (i.e.,  $\dot{\zeta} < 0$  and  $P_{ress,opt} > 0$ ). This case is happening if the SOE is converging to the reference value from above. The slope of the curve  $\dot{m}_{ress}(P_{ress})$  keeps decreasing, moving the solution toward the value  $P_{ress} = 0$ , as the reference value is being approached.
3.  $\zeta(t) < \zeta_{ref}$ , decreasing SOE (i.e.,  $\dot{\zeta} < 0$  and  $P_{ress,opt} > 0$ ). This is the dual of case 1: the SOE is below the reference value and keeps decreasing, therefore it is necessary to move the solution to the positive range. This is in fact realized, because the slope of  $\dot{m}_{ress}(P_{ress})$  is decreasing, according to the fact that  $\text{sign}(s \cdot p(\zeta)) = \text{sign} \dot{\zeta}$ .
4.  $\zeta(t) < \zeta_{ref}$ , increasing SOE (i.e.,  $\dot{\zeta} > 0$  and  $P_{ress,opt} < 0$ ). This is the dual of case 2, and in this case as well the SOE tends to converge to the reference value, this time from below.

As an example, the behavior of ECMS with a constant power request of 30 kW is shown in Figure 4.59: it is clearly visible the tendency to stabilize at an SOE level close to the reference value. Three points are shown in the figure: A corresponds to case 2 of the previous list, B to case 3, and C to case 4. The respective cost functions are shown in Figure 4.60, where it is possible to note how the minimum of the

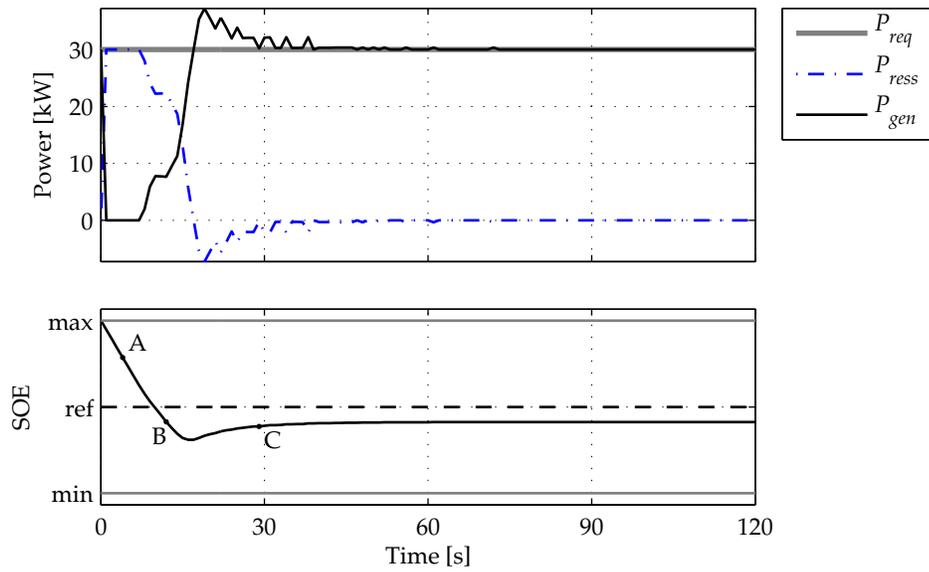
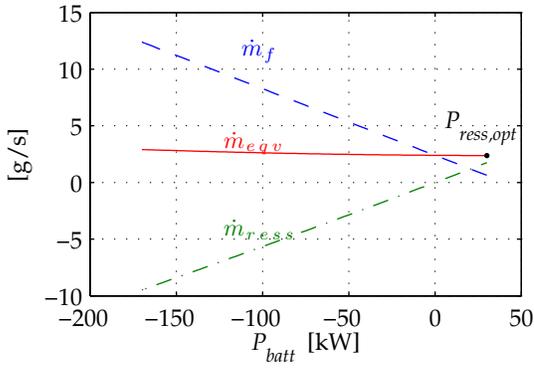
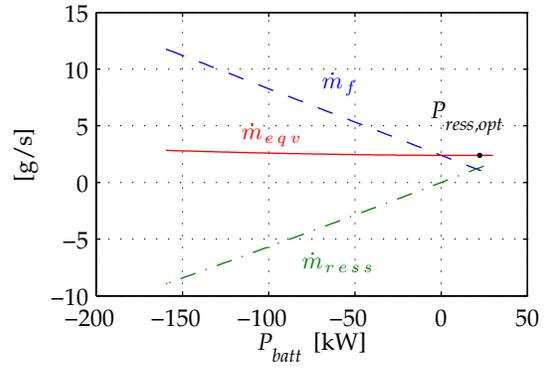


Figure 4.59: Behavior of ECMS under a constant power request  $P_{req} = 30$  kW

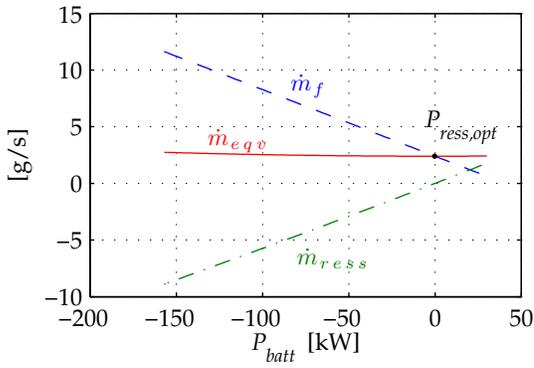
equivalent fuel consumption  $\dot{m}_f(P_{ress})$  is different in the three cases; the location of this minimum only depends on the value of the penalty function, because the equivalence factor and the curve  $\dot{m}_f(P_{ress})$  are identical in the three cases. Note that the SOE does not converge exactly to the reference value, but to a slightly lower value: this is due to the fact that the penalty function is a cubic polynomial in  $\zeta$  (i.e.,  $n_p = 1$ ) and is very close to zero for a range of SOE values around the reference value. In order to show this, the points A, B, and C are also represented on the curve  $p(\zeta)$  in Figure 4.60.d.



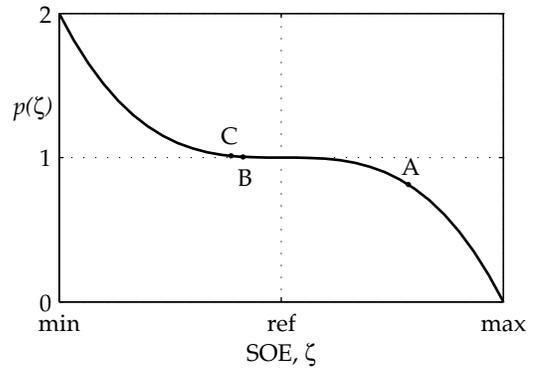
(a) point A,  $t = 4$  s,  $p(\zeta) = -0.31$



(b) point B,  $t = 12$  s,  $p(\zeta) = 8 \cdot 10^{-4}$



(c) point C,  $t = 29$  s,  $p(\zeta) = 0.01$



(d) points A, B, and C on the curve  $p(\zeta)$

Figure 4.60: ECMS cost function for the three cases of Figure 4.59

## Chapter 5

### CONCLUSION

Energy management is of fundamental importance in hybrid electric vehicles, for exploiting the advantages deriving from the availability of a rechargeable energy buffer. With the increasing interest in plug-in hybrid electric vehicles and their future commercial availability, energy management is becoming even more important, since the use of the electric power distribution grid is also to be taken into account.

Several energy management techniques have been proposed in the literature, but there is still the need for their deeper understanding and more organic formalization. The objective of this dissertation is to provide an organic review of existing energy management strategies and provide new insights into some of them, showing interconnections and potential for improvements. Using an analytical approach, the optimal control problem of energy management in hybrid electric vehicles is formulated formally. This is the basis for a better understanding of the available techniques and for mathematical insights that, complementing experience and intuition, allow for designing and implementing more effective energy management controllers.

An appropriate model of the vehicle powertrain is necessary to study the effect of any control strategy. Such a model is presented in Chapter 2 and is used to compute the vehicle power request and the energy characteristics of a prescribed driving cycle. The chapter provides a detailed description of modeling assumptions and techniques for the most relevant powertrain components; the vehicle model is validated using a comparison of the simulation results with experimental

data, showing good agreement between them. A modular approach is used for implementing the simulator, to facilitate the reuse of component models and the creation of simulators for various hybrid architectures.

The review of existing energy management strategies proposed in Chapter 3 demonstrates how the problem is complex and how different methods can be used to solve it. These methods can be quite different from each other, but most of them rely on the formulation of an optimal control problem. Optimal control theory is used throughout the dissertation to uniform notation and concepts among the various strategies investigated, and to point out the similarities and interconnections between existing strategies. The problem formulation in formal terms and the use of results from optimal control theory is useful to derive control strategies that are based on algorithms, unlike traditional strategies based on rules. A control strategy based on a parameterized algorithm can be more effective, robust and easy to implement than one based on rules. The effectiveness is due to the fact that the algorithm is derived using mathematical results that guarantee an optimal solution, or a solution close to the optimal. The robustness is high if the algorithm is derived following general procedures and built in a way that allows to account for variations in the vehicle parameters. Finally, while a rule-based controller requires the definition of numerous rules of the type if-then-else and the appropriate tuning of the relative thresholds, a controller based on optimal control theory can be implemented as a sequence of mathematical operations with only a few parameters to be tuned after implementation. Moving the same algorithm to a different vehicle requires only to change the relevant parameters to reflect the new vehicle configuration, rather than rewriting the rules or change all their parameters.

Chapter 4 provides a detailed explanation of how three techniques (dynamic programming, Pontryagin's minimum principle and equivalent consumption minimization strategy, or ECMS) are implemented on two case studies. Dynamic programming provides the global optimal solution in a numerical way and represents a benchmark for the other strategies; Pontryagin's minimum principle represents the analytical solution to the problem based on optimal control theory and is

shown to be substantially equivalent to the numerical solution obtained with dynamic programming. Unlike the previous two techniques, ECMS does not require knowledge of the entire driving cycle in advance and is implementable on-line; however, with appropriate formulation of the optimal control problem, it is shown that ECMS is, in fact, an implementation of the solution based on Pontryagin's minimum principle and therefore it is very close to the optimal solution (the difference being due to some implementation details). The links among the three strategies allow to justify the use of ECMS as a sub-optimal control and are used to propose improvements to it. In particular, a more effective method of implementing an adaptive ECMS is proposed in Section 4.11. Similar ideas can be applied to other strategies that are presented in Chapter 3 but are not studied in detail, such as model predictive control and stochastic dynamic programming. Both these techniques, in fact, have been developed for being easily implementable on-line, and show promising results.

The point of view of this dissertation is rather general, and focused on the essence of the energy management problem. However, implementing a strategy on a vehicle presents many more challenges in addition to choosing the optimal control algorithm: these issues, which may also be related to limitations in available memory and computational power, are not considered here but may be part of successive efforts.

The framework presented here allows to formulate the optimization problem using any cost function, and to derive energy management controls applicable in any hybrid vehicle. In particular, the results presented are valid for charge-sustaining as well as charge-depleting (plug-in) hybrid vehicles, in which the battery is allowed to discharge completely. In addition to the possibility of a decreasing state of charge, plug-in hybrids are characterized by the fact that, unlike in charge-sustaining hybrids, fuel consumption is not necessarily the main minimization objective. Other quantities, such as total energy consumption, total emissions (including those generated in the production of the electrical energy used), actual cost, battery wear etc. can be considered as minimization goals.

In fact, the tools and methodologies for energy flow optimization presented in this dissertation are general and can be extended to other engineering applications. The relevance of such tools is significant at a time in which energy optimization and reduction of oil consumption make headlines almost every day.

## Bibliography

- [1] EPA Driving Schedules. [Online]. Available: <http://www.epa.gov/nvfel/testing/dynamometer.htm>
- [2] P. Pisu, C. Cantemir, N. Dembski, G. Rizzoni, L. Serrao, J. Josephson, and J. Russell, "Evaluation of powertrain solutions for future tactical truck vehicle systems," *Proceedings of SPIE*, vol. 6228, 2006.
- [3] R. Bornatico, A. Storti, L. Mandrioli, A. Zappavigna, Y. Guezennec, and G. Rizzoni, "Ni-MH battery characterization and state-of-charge estimation for HEV applications," *Proceedings of the 2007 ASME International Mechanical Engineering Congress and Exposition*, 2007.
- [4] EATON Hydraulics, "420 Mobile Piston Pump Technical Catalog Manual," April 2004.
- [5] A. Bryson, *Dynamic optimization*. Menlo Park, CA: Addison Wesley Longman, 1999.
- [6] J. How. Principles of optimal control. Class Notes, MIT OpenCourseWare. [Online]. Available: <http://ocw.mit.edu/OcwWeb/Electrical-Engineering-and-Computer-Science/6-231Dynamic-Programming-and-Stochastic-ControlFall2002/CourseHome/>
- [7] G. Paganelli, T. Guerra, S. Delprat, J. Santin, M. Delhom, and E. Combes, "Simulation and assessment of power control strategies for a parallel hybrid car," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 214, no. 7, pp. 705–717, 2000.
- [8] K. Koprubasi, "Modeling and control of a hybrid-electric vehicle for drivability and fuel economy improvements," Ph.D. dissertation, The Ohio State University, 2008.
- [9] P. Pisu and G. Rizzoni, "A comparative study of supervisory control strategies for hybrid electric vehicles," *Control Systems Technology, IEEE Transactions on*, vol. 15, no. 3, pp. 506–518, 2007.

- [10] B. Gu and G. Rizzoni, "An adaptive algorithm for hybrid electric vehicle energy management based on driving pattern recognition," *Proceedings of the 2006 ASME International Mechanical Engineering Congress and Exposition*, 2006.
- [11] E. Camacho and C. Bordons, *Model Predictive Control*. London: Springer, 1999.
- [12] N. Jalil, N. Kheir, and M. Salman, "A rule-based energy management strategy for a series hybrid vehicle," *Proceedings of the 1997 American Control Conference*, vol. 1, 1997.
- [13] J. M. Miller, *Propulsion Systems for Hybrid Vehicles*. London (UK): The Institution of Electrical Engineers, 2003.
- [14] L. Guzzella and A. Sciarretta, *Vehicle Propulsion Systems: Introduction to Modeling and Optimization*. Springer, 2007.
- [15] Y. Guezennec and G. Rizzoni, "Hybrid electric vehicles," Lecture notes, The Ohio State University, 2007.
- [16] Q. Li, "Development and refinement of a hybrid electric vehicle simulator and its application in design space exploration," Ph.D. dissertation, The Ohio State University, 1998.
- [17] A. Holmes, "Modeling and analysis of electrically variable transmissions," Master's thesis, The Ohio State University, 2000.
- [18] C. Musardo and B. Staccia, "Energy management strategies for hybrid electric vehicles," Master's thesis, Politecnico di Milano, 2003.
- [19] B. Gu, "Supervisory control strategy development for a hybrid electric vehicle," Master's thesis, The Ohio State University, 2006.
- [20] X. Wei, "Optimal control of hybrid electric vehicles," Ph.D. dissertation, The Ohio State University, 2004.
- [21] J. Morbitzer, "High-level modeling, supervisory control strategy development, and validation for a proposed power-split hybrid-electric vehicle design," Master's thesis, The Ohio State University, 2005.
- [22] M. D. Arnett, "Development and implementation of an electric all-wheel drive system," Master's thesis, The Ohio State University, 2007.
- [23] K. Sevel, "Development of a start-stop system for a hybrid-electric vehicle," Master's thesis, The Ohio State University, 2007.
- [24] (2008, January) Website of the U.S. Environmental Protection Agency (EPA). [Online]. Available: <http://www.epa.gov/fueleconomy/regulations.htm>

- [25] M. O’Keefe and T. Markel, “Dynamic programming applied to investigate energy management strategies for a plug-in hev,” *Proceedings of the 22nd International Battery, Hybrid and Fuel Cell Electric Vehicle Symposium & Exposition (EVS-22)*, 2006.
- [26] L. Johannesson, M. Åsbogård, and B. Egardt, “Assessing the potential of predictive control for hybrid vehicle powertrains using stochastic dynamic programming,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 8, no. 1, pp. 71–83, March 2007.
- [27] Q. Gong, Y. Li, and Z.-R. Peng, “Trip based near globally optimal power management of plug-in hybrid electric vehicles using gas-kinetic traffic flow model,” *Proceedings of the 17th IFAC World Congress*, 2008.
- [28] A. Brahma, Y. Guezennec, and G. Rizzoni, “Optimal energy management in series hybrid electric vehicles,” *Proceedings of the 2000 American Control Conference*, vol. 1, no. 6, pp. 60–64, 2000.
- [29] L. Pérez, G. Bossio, D. Moitre, and G. García, “Optimization of power management in an hybrid electric vehicle using dynamic programming,” *Mathematics and Computers in Simulation*, vol. 73, no. 1-4, pp. 244–254, 2006.
- [30] R. Bellman, “The theory of dynamic programming,” *Proceedings of the National Academy of Sciences*, vol. 38, no. 8, pp. 716–719, 1952.
- [31] H. Sussmann and J. Willems, “300 years of optimal control: from the brachystochrone to the maximum principle,” *IEEE Control Systems Magazine*, vol. 17, no. 3, pp. 32–44, 1997.
- [32] L. Pontryagin, V. Boltyanskii, R. Gamkrelidze, and E. Mishchenko, *The mathematical theory of optimal processes*. New York: Interscience Publishers, 1962.
- [33] D. Bertsekas, *Dynamic Programming and Optimal Control*. Belmont, MA: Athena Scientific, 1995.
- [34] H. P. Geering, *Optimal Control with Engineering Applications*. Berlin Heidelberg: Springer, 2007.
- [35] M. Anatone, R. Cipollone, and A. Sciarretta, “Control-oriented modeling and fuel optimal control of a series hybrid bus,” *SAE paper 2005-01-1163*, 2005.
- [36] X. Wei, L. Guzzella, V. Utkin, and G. Rizzoni, “Model-based fuel optimal control of hybrid electric vehicle using variable structure control systems,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 129, p. 13, 2007.
- [37] G. Rousseau, D. Sinoquet, and P. Rouchon, “Constrained optimization of energy management for a mild-hybrid vehicle,” *Oil and Gas Science and Technology - Rev. IFP*, vol. 62, no. 4, pp. 623–634, 2007.

- [38] L. Serrao and G. Rizzoni, "Optimal control of power split for a hybrid electric refuse vehicle," *Proceedings of the 2008 American Control Conference*, 2008.
- [39] R. Cipollone and A. Sciarretta, "Analysis of the Potential Performance of a Combined Hybrid Vehicle with Optimal Supervisory Control," *Proceedings of the 2006 IEEE International Conference on Control Applications*, pp. 2802–2807, 2006.
- [40] G. Paganelli, G. Ercole, A. Brahma, Y. Guezennec, and G. Rizzoni, "General supervisory control policy for the energy optimization of charge-sustaining hybrid electric vehicles," *JSAE Review*, vol. 22, no. 4, pp. 511–518, 2001.
- [41] C. Musardo, G. Rizzoni, Y. Guezennec, and B. Staccia, "A-ECMS: An adaptive algorithm for hybrid electric vehicle energy management," *European Journal of Control*, vol. 11, no. 4-5, pp. 509–524, 2005.
- [42] E. Cacciatori, N. Vaughan, and J. Marco, "Energy management strategies for a parallel hybrid electric powertrain: Fuel economy optimisation with driveability requirements," *Hybrid Vehicle Conference, IET The Institution of Engineering and Technology, 2006*, pp. 157–172, Dec. 2006.
- [43] X. He, M. Parten, and T. Maxwell, "Energy management strategies for a hybrid electric vehicle," *Proceedings of the 2005 IEEE Vehicle Power and Propulsion Conference*, pp. 536–540, 2005.
- [44] T. Hofman, M. Steinbuch, R. van Druten, and A. Serrarens, "Rule-based energy management strategies for hybrid vehicles," *Int. J. Electric and Hybrid Vehicles*, vol. 1, no. 1, pp. 71–94, 2007.
- [45] C. Lin, J. Kang, J. Grizzle, and H. Peng, "Energy management strategy for a parallel hybrid electric truck," *Proceedings of the 2001 American Control Conference*, vol. 4, pp. 2878–2883, 2001.
- [46] C. Lin, H. Peng, J. Grizzle, and J. Kang, "Power management strategy for a parallel hybrid electric truck," *IEEE Transactions on Control Systems Technology*, vol. 11, no. 6, pp. 839–849, 2003.
- [47] B. Wu, C. Lin, Z. Filipi, H. Peng, and D. Assanis, "Optimal power management for a hydraulic hybrid delivery truck," *Vehicle System Dynamics*, vol. 42, no. 1, pp. 23–40, 2004.
- [48] Y. Zhu, Y. Chen, G. Tian, H. Wu, and Q. Chen, "A four-step method to design an energy management strategy for hybrid vehicles," *Proceedings of the 2004 American Control Conference*, vol. 1, 2004.
- [49] L. Louca and U. Yildir, "Modeling and reduction techniques for studies of integrated hybrid vehicle systems," *Proceedings of the 4th International Symposium on Mathematical Modeling*, 2003.

- [50] NREL Advanced Heavy Hybrid Propulsion Systems. [Online]. Available: <http://www.nrel.gov/vehiclesandfuels/ahhps/>
- [51] N. Dembski, G. Rizzoni, A. Soliman, J. Fravert, and K. Kelly, "Development of refuse vehicle driving and duty cycles," *SAE paper 2005-01-1165*, 2005.
- [52] EcoCAR Challenge website. [Online]. Available: <http://www.ecocarchallenge.org/>
- [53] G. Rizzoni, Y. Guezennec, A. Brahma, X. Wei, and T. Miller, "VP-SIM: A Unified Approach to Energy and Power Flow Modeling Simulation and Analysis of Hybrid Vehicles," *Proceedings of the 2000 Future Car Congress, Arlington, VA*, 2000.
- [54] X. Wei, P. Pisu, G. Rizzoni, and S. Yurkovich, "Dynamic modeling of hybrid electric drivetrain for fuel economy, performance and drivability evaluations," *Proceedings of the 2003 ASME International Mechanical Engineering Congress and Exposition*, pp. 16–21, 2003.
- [55] Easy5 website. [Online]. Available: <http://www.mscsoftware.com/products/easy5.cfm>
- [56] MSC Software, "Dynamic system modeling, simulation, and analysis using easy5," *14th Korea VPD Conference*, 2004.
- [57] AMESim website. [Online]. Available: <http://www.amesim.com/>
- [58] Website of the Modelica Association. [Online]. Available: <http://www.modelica.org/>
- [59] H. Elmqvist, "A structured model language for large continuous systems," Ph.D. dissertation, Lund Institute of Technology, 1978.
- [60] M. Tiller, *Introduction to Physical Modeling With Modelica*. Kluwer Academic Publishers, 2001.
- [61] J. Heywood, *Internal combustion engine fundamentals*. New York: McGraw-Hill, 1988.
- [62] G. Rizzoni, L. Guzzella, and B. M. Baumann, "Unified modeling of hybrid electric vehicle drivetrains," *IEEE/ASME Transactions on Mechatronics*, vol. 4, no. 3, pp. 246–257, September 1999.
- [63] G. Rizzoni and K. Srinivasan, "Powertrain dynamics," Lecture notes, The Ohio State University.
- [64] A. J. Kotwicki, "Dynamic models for torque converter equipped vehicles," *SAE Technical Papers*, no. 820393, 1982.

- [65] H. B. Pacejka, *Tire and Vehicle Dynamics*. Oxford (UK): SAE and Butterworth-Heinemann, 2002.
- [66] L. Serrao, Z. Chehab, Y. Guezennec, and G. Rizzoni, "An aging model of Ni-MH batteries for hybrid electric vehicles," *Proceedings of the 2005 IEEE Vehicle Power and Propulsion Conference (VPP05)*, pp. 78–85, 2005.
- [67] Z. Chehab, L. Serrao, Y. Guezennec, and G. Rizzoni, "Aging characterization of nickel – metal hydride batteries using electrochemical impedance spectroscopy," *Proceedings of the 2006 ASME International Mechanical Engineering Congress and Exposition*, 2006.
- [68] S. Drouilhet and B. Johnson, "A battery life prediction method for hybrid power applications," *Proceedings of the 35th AIAA Aerospace Sciences Meeting & Exhibit, Reno, NV*, 1997.
- [69] M. Dubarry, V. Svoboda, R. Hwu, and B. Liaw, "Capacity and power fading mechanism identification from a commercial cell evaluation," *Journal of Power Sources*, vol. 165, no. 2, pp. 566–572, 2007.
- [70] —, "Capacity loss in rechargeable lithium cells during cycle life testing: The importance of determining state-of-charge," *Journal of Power Sources*, vol. 174, no. 2, pp. 1121–1125, 2007.
- [71] A. Sciarretta and L. Guzzella, "Control of hybrid electric vehicles," *IEEE Control Systems Magazine*, pp. 60–70, April 2007.
- [72] A. Sciarretta, M. Back, and L. Guzzella, "Optimal control of parallel hybrid electric vehicles," *IEEE Transactions on Control Systems Technology*, vol. 12, no. 3, pp. 352–363, 2004.
- [73] F. Lewis and V. Syrmos, *Optimal Control*. Wiley-Interscience, 1995.
- [74] C. Lin, Z. Filipi, Y. Wang, L. Louca, H. Peng, D. Assanis, and J. Stein, "Integrated, feed-forward hybrid electric vehicle simulation in simulink and its use for power management studies," *SAE Paper 2001-01-1334*, 2001.
- [75] I. Kolmanovsky, I. Siverguina, and B. Lygoe, "Optimization of powertrain operating policy for feasibility assessment and calibration: stochastic dynamic programming approach," *Proceedings of the 2002 American Control Conference*, vol. 2, pp. 1425–1430, 2002.
- [76] W. Li, G. Xu, H. Tong, and Y. Xu, "Design of optimal, robust energy management strategy for a parallel hev," *Proceedings of the 2007 IEEE International Conference on Robotics and Biomimetics*, pp. 1894–1899, 2007.
- [77] C. Lin, H. Peng, and J. Grizzle, "A stochastic control strategy for hybrid electric vehicles," *American Control Conference*, 2004.

- [78] C. Lin, M. Kim, H. Peng, and J. Grizzle, "System-level model and stochastic optimal control for a pem fuel cell hybrid vehicle," *Journal of Dynamic Systems, Measurement, and Control*, vol. 128, p. 878, 2006.
- [79] C. Musardo, B. Staccia, S. Midlam-Mohler, Y. Guezennec, and G. Rizzoni, "Supervisory control for nox reduction of an HEV with a mixed-mode HCCI/CIDI engine," *Proceedings of the 2005 American Control Conference*, 2005.
- [80] M. Back, M. Simons, F. Kirschbaum, and V. Krebs, "Predictive control of drivetrains," *Proceedings of the IFAC 15th Triennial World Congress*, 2002.
- [81] E. Nuijten, M. Koot, J. Kessels, B. de Jager, M. Heemels, W. Hendrix, and P. van den Bosch, "Advanced energy management strategies for vehicle power nets," *Proceedings of EAEC 9th Int. Congress: European Automotive Industry Driving Global Changes*, 2003.
- [82] D. De Vito, A. Miotti, and R. Scattolini, "Power flow management with predictive capabilities for a hybrid fuel cell vehicle," *Proceedings of the 5th IFAC Symposium on Advances in Automotive Control*, 2007.
- [83] M. Salman, N. Schouten, and N. Kheir, "Control strategies for parallel hybrid vehicles," *Proceedings of the 2000 American Control Conference.*, vol. 1, no. 6, pp. 524–528, 2000.
- [84] N. Schouten, M. Salman, and N. Kheir, "Fuzzy logic control for parallel hybrid vehicles," *IEEE Transactions on Control Systems Technology*, vol. 10, no. 3, pp. 460–468, 2002.
- [85] F. Salmasi, "Control strategies for hybrid electric vehicles: Evolution, classification, comparison, and future trends," *IEEE Transactions on Vehicular technology*, vol. 56, no. 5, pp. 2393–2404, 2007.
- [86] G. Passier, F. V. Conte, S. Smets, F. Badin, A. Brouwer, M. Alaküla, D. Santini, and M. Alexander, "Status overview of hybrid and electric vehicle technology," International Energy Agency (IEA), Tech. Rep., 2007.
- [87] T. Markel, A. Brooker, J. Gonder, M. O. Keefe, A. Simpson, and M. Thornton, "Plug-in hybrid vehicle analysis," National Renewable Energy Laboratory, Tech. Rep., 2006.
- [88] M. Tateno, "Emissions modeling for enhanced hybrid electric powertrain control," Master's thesis, The Ohio State University, 2001.
- [89] J. Kang, I. Kolmanovsky, and J. Grizzle, "Dynamic optimization of lean burn engine aftertreatment," *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 123, p. 153, 2001.

- [90] M. Morari, C. Garcia, and D. M. Prett, "Model predictive control: Theory and practice—a survey," *Automatica*, vol. 25, no. 3, pp. 335–348, 1989. [Online]. Available: <http://control.ee.ethz.ch/index.cgi?page=publications;action=details;id=2611>
- [91] T. Donato, L. Serrao, and G. Rizzoni, "Multi-objective optimization of a heavy duty hybrid electric vehicle," *Workshop on Electric, Hybrid and Solar Vehicles, University of Salerno (Italy)*, 2007.
- [92] C. Cantemir, G. Ursescu, L. Serrao, G. Rizzoni, J. Bechtel, T. Udvare, and M. Letherwood, "Concept design of a new generation military vehicle," *Proceedings of SPIE*, vol. 6201, 2006.
- [93] C.-G. Cantemir, G. Ursescu, L. Serrao, G. Rizzoni, J. Bechtel, T. Udvare, and M. Letherwood, "Island concept evt," *SAE Paper 06FFL-250*, 2006.
- [94] G. Rizzoni, J. Josephson, A. Soliman, C. Hubert, C. Cantemir, N. Dembski, P. Pisu, D. Mikesell, L. Serrao, and J. Russell, "Modeling, simulation, and concept design for hybrid-electric medium-size military trucks," *Proceedings of SPIE*, vol. 5805, 2005.
- [95] D. Kirk, *Optimal control theory*. Prentice-Hall Englewood Cliffs, N. J, 1970.
- [96] H. Pesch, "A practical guide to the solution of real-life optimal control problems," *Control and Cybernetics*, vol. 23, no. 1/2, pp. 7–60, 1994.
- [97] A. Bemporad, M. Morari, and N. L. Ricker, *Model Predictive Control Toolbox 3 - User's Guide*, The Mathworks, Inc., Natick, MA.
- [98] A. Bemporad and M. Morari, "Robust model predictive control: A survey," *Lecture notes in control and information sciences*, pp. 207–226, 1999.
- [99] A. Bemporad, F. Borrelli, and M. Morari, "Model predictive control based on linear programming - the explicit solution," *IEEE Transactions on Automatic Control*, vol. 47, no. 12, pp. 1974–1985, 2002.
- [100] K. Chau and Y. Wong, "Overview of power management in hybrid electric vehicles," *Energy Conversion and Management*, vol. 43, no. 15, pp. 1953–1968, 2002. [Online]. Available: <http://linkinghub.elsevier.com/retrieve/pii/S0196890401001480>
- [101] O. Mangasarian, "Sufficient conditions for the optimal control of nonlinear systems," *SIAM Journal on Control*, vol. 4, p. 139, 1966.
- [102] A. V. Arutyunov, "The pontryagin maximum principle and sufficient optimality conditions for nonlinear problems," *Differential Equations*, vol. 39, no. 12, pp. 1671–1679, December 2003.

- [103] M. S. I. Arsie, G. Rizzo, "Optimal design of a hybrid solar vehicle," *Proceedings of the 8th International Symposium on Advanced Vehicle Control (AVEC 06)*, August 2006.
- [104] I. Arsie, M. Marotta, C. Pianese, G. Rizzo, and M. Sorrentino, "Optimal design of a hybrid electric car with solar cells," *Proceedings of the 1st AUTOCOM Workshop on Preventive and Active Safety Systems for Road Vehicles*, 2005.
- [105] L. Johannesson and B. Egardt, "Approximate dynamic programming applied to parallel hybrid powertrains," *Proceedings of the 17th IFAC World Congress*, 2008.
- [106] M. Ivarsson, J. Aslund, and L. Nielsen, "Optimal speed on small gradients - consequences of a non-linear fuel map," *Proceedings of the 17th IFAC World Congress*, 2008.
- [107] M. Cavalletti, J. Piovesan, C. T. Abdallah, S. Longhi, P. Dorato, and G. Ippoliti, "Statistical learning applied to the energy management in a fuel cell electric vehicle," *Proceedings of the 17th IFAC World Congress*, 2008.
- [108] I. Arsie, R. Di Martino, G. Rizzo, and M. Sorrentino, "Energy management for a hybrid solar vehicle with series structure," *Proceedings of the 17th IFAC World Congress*, 2008.
- [109] S. Kermani, S. Delprat, T. Guerra, and R. Trigui, "Real time control of hybrid electric vehicle on a prescribed road," *Proceedings of the 17th IFAC World Congress*, 2008.
- [110] E. Hellstrom, J. Aslund, and L. Nielsen, "Design of a well-behaved algorithm for on-board look-ahead control," *Proceedings of the 17th IFAC World Congress*, 2008.
- [111] J. Liu and H. Peng, "Automated modelling of power-split hybrid vehicles," *Proceedings of the 17th IFAC World Congress*, 2008.
- [112] O. Sundstrom, L. Guzzella, and P. Soltic, "Optimal hybridization in two parallel hybrid electric vehicles using dynamic programming," *Proceedings of the 17th IFAC World Congress*, 2008.
- [113] T. van Keulen, B. de Jager, and M. Steinbuch, "An adaptive sub-optimal energy management strategy for hybrid drive trains," *Proceedings of the 17th IFAC World Congress*, 2008.
- [114] R. Bartholomaeus, M. Klingner, and M. Lehnert, "Prediction of power demand for hybrid vehicles operating in fixed-route service," *Proceedings of the 17th IFAC World Congress*, 2008.

- [115] T. Kim, C. Manzie, and H. Watson, "Fuel economy benefits of look-ahead capability in a mild hybrid configuration," *Proceedings of the 17th IFAC World Congress*, 2008.
- [116] T. Hofman, M. Steinbuch, R. van Druten, and A. Serrarens, "Rule-based equivalent fuel consumption minimization strategies for hybrid vehicles," *Proceedings of the 17th IFAC World Congress*, 2008.
- [117] M. Suzuki, S. Yamaguchi, T. Araki, P. Raksincharoensak, M. Yoshizawa, and M. Nagai, "Fuel economy improvement strategy for light duty hybrid truck based on fuel consumption computational model using neural network," *Proceedings of the 17th IFAC World Congress*, 2008.
- [118] The Mathworks website. [Online]. Available: <http://www.mathworks.com>
- [119] N. Picciano, "Battery aging and characterization of nickel metal hydride and lead acid batteries," Master's thesis, The Ohio State University, 2007.
- [120] M. Miner, "Cumulative damage in fatigue," *Journal of Applied Mechanics*, vol. 12, pp. 159–164, 1945.
- [121] A. Palmgren, "The fatigue life of ball-bearings," *Zeitschrift des Vereins Deutscher Ingenieure*, vol. 68, pp. 339–341, 1924.
- [122] C. Suozzo, "Lead-acid battery aging and state of health diagnosis," Master's thesis, The Ohio State University, 2008.
- [123] S. Qin and T. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 11, no. 7, pp. 733–764, 2003.
- [124] M. Nikolaou, "Model predictive controllers: A critical synthesis of theory and industrial needs," *Advances in Chemical Engineering*, vol. 26, pp. 132–204, 2001.
- [125] J. Rawlings, "Tutorial overview of model predictive control," *IEEE Control Systems Magazine*, vol. 20, no. 3, pp. 38–52, 2000.
- [126] M. Ceraolo, A. di Donato, and G. Franceschi, "A general approach to energy optimization of hybrid electric vehicles," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 3, pp. 1433–1441, 2008.
- [127] M. Koot, "Energy management for vehicular electric power systems," Ph.D. dissertation, Technical University of Eindhoven, 2006.
- [128] K. Koprubasi, J. Morbitzer, E. Westervelt, and G. Rizzoni, "Toward a framework for the hybrid control of a multi-mode hybrid-electric driveline," *Proceedings of the 2006 American Control Conference*, 2006.

- [129] M. West, C. Bingham, and N. Schofield, "Predictive control for energy management in all/more electric vehicles with multiple energy storage units," *Proceedings of the IEEE International Electric Machines and Drives Conference (IEMDC '03)*, vol. 1, 2003.
- [130] P. Bosch and F. Lootsma, "Scheduling of power generation via large-scale nonlinear optimization," *Journal of Optimization Theory and Applications*, vol. 55, no. 2, pp. 313–326, 1987.
- [131] E. Tate, J. Grizzle, and H. Peng, "SP-SDP for fuel consumption and tailpipe emissions minimization in an EVT hybrid," *IEEE Transactions on Control Systems Technology*, 2007.
- [132] ———, "Shortest Path Stochastic Control for Hybrid Electric Vehicles," *International Journal of Robust and Nonlinear Control*, vol. 18, no. 14, pp. 1409 – 1429, 2008.
- [133] R. Zhang and Y. Chen, "Control of hybrid dynamical systems for electric vehicles," *Proceedings of the 2001 American Control Conference*, pp. 2884–2889, 2001.
- [134] V. D. Blondel and J. N. Tsitsiklis, "A survey of computational complexity results in systems and control," *Automatica*, vol. 36, no. 9, pp. 1249–1274, 2000.
- [135] L. Mandrioli and A. Zappavigna, "A state of charge estimation algorithm for NiMH batteries," Master's thesis, Politecnico di Milano, 2007.
- [136] Z. Chehab, "Aging characterization of nimh batteries for hybrid electric vehicles," Master's thesis, The Ohio State University, 2006.
- [137] M. Athans and P. Falb, *Optimal control*. McGraw-Hill New York, 1966.
- [138] R. Hartl, S. Sethi, and R. Vickson, "A survey of the maximum principles for optimal control problems with state constraints," *SIAM Review*, vol. 37, no. 2, pp. 181–218, 1995.
- [139] J. G. Kassakian, H.-C. Wolf, J. M. Miller, and C. J. Hurton, "Automotive electrical systems circa 2005," *IEEE Spectrum*, August 1996.
- [140] K. Ahn, S. Cho, W. Lim, Y. Park, and J. Lee, "Performance analysis and parametric design of the dual-mode planetary gear hybrid powertrain," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 220, no. 11, pp. 1601–1614, 2006.
- [141] J. Liu, H. Peng, and Z. Filipi, "Modeling and analysis of the toyota hybrid system," *Proceedings of the 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 134–139, 2005.

- [142] C. W. Ayers, J. S. Hsu, L. D. Marlino, C. W. Miller, G. W. Ott, and C. B. Oland, "Evaluation of 2004 toyota prius hybrid electric drive system," Oak Ridge National Laboratory, Tech. Rep., 2004.
- [143] C. Medin, "Modelling of a planetary gear train for simulation of a power split hybrid vehicle," Master's thesis, Chalmers University of Technology, 2003.
- [144] J. Meisel, "An analytic foundation for the toyota prius ths-ii powertrain with a comparison to a strong parallel hybrid-electric powertrain," *SAE paper 2006-01-0666*, 2006.
- [145] J. M. Miller, "Hybrid electric vehicle propulsion system architectures of the e-cvt type," *IEEE Transactions on Power Electronics*, vol. 21, no. 3, pp. 756–767, 2006.
- [146] P. Rodatz, G. Paganelli, A. Sciarretta, and L. Guzzella, "Optimal power management of an experimental fuel cell-supercapacitor powered hybrid vehicle," *Control Engineering Practice*, no. 13, pp. 41–53, 2005.
- [147] C. Musardo, B. Staccia, S. Bittanti, Y. Guezennec, L. Guzzella, and G. Rizzoni, "An adaptive algorithm for hybrid electric vehicles energy management," *Proceedings of the Fisita World Automotive Congress*, 2004.
- [148] J. Hansson, "Analysis and control of a hybrid vehicle powered by a free-piston energy converter," Master's thesis, Royal Institute of Technology, Stockholm, Sweden, 2006.
- [149] A. Piccolo, L. Ippolito, V. Galdi, and A. Vaccaro, "Optimization of Energy Flow Management in Hybrid Electric Vehicles via Genetic Algorithms," *Proceedings of the 2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Como, Italy, July*, 2001.
- [150] T. Donato, L. Serrao, and G. Rizzoni, "A two-step optimisation method for the preliminary design of a hybrid electric vehicle," *International Journal of Electric and Hybrid Vehicles*, vol. 1, no. 2, 2008.
- [151] V. H. Johnson, K. B. Wipke, and D. J. Rausen, "Hev control strategy for real-time optimization of fuel economy and emissions," *SAE paper 2000-01-1543*, 2000.
- [152] T. Markel and A. Simpson, "Energy storage systems considerations for grid-charged hybrid electric vehicles," *Proceedings of the 2005 IEEE Vehicle Power and Propulsion Conference (VPP05)*, pp. 344–349, 2005.
- [153] W. Kempton and J. Tomić, "Vehicle-to-grid power fundamentals: Calculating capacity and net revenue," *Journal of Power Sources*, vol. 144, no. 1, pp. 268–279, 2005.

- [154] M. Duvall, "Battery evaluation for plug-in hybrid electric vehicles," *Proceedings of the 2005 IEEE Vehicle Power and Propulsion Conference*, pp. 338–343, 2005.
- [155] J. Won and R. Langari, "Intelligent energy management agent for a parallel hybrid vehicle - part II: torque distribution, charge sustenance strategies, and performance results," *IEEE Transactions on Vehicular Technology*, vol. 54, no. 3, pp. 935–953, 2005.
- [156] R. Langari and J. Won, "Intelligent energy management agent for a parallel hybrid vehicle - part I: system architecture and design of the driving situation identification process," *IEEE Transactions on Vehicular Technology*, vol. 54, no. 3, pp. 925–934, 2005.
- [157] A. Kleimaier and D. Schroder, "Optimization strategy for design and control of a hybrid vehicle," *Proceedings of the 6th International Workshop on Advanced Motion Control*, pp. 459–464, 2000.
- [158] L. Buie, C. Mitts, M. Fry, and P. Fussey, "An application of cost-based power management control strategies to hybrid fuel cell vehicles," *SAE paper 2004-01-1299*, 2004.
- [159] S. W. W. Peter Kall, *Stochastic Programming*. New York: John Wiley & Sons, 1994.
- [160] J. Van Mierlo, P. Van den Bossche, and G. Maggetto, "Models of energy sources for ev and hev: fuel cells, batteries, ultracapacitors, flywheels and engine-generators," *Journal of Power Sources*, vol. 128, no. 1, pp. 76–89, 2004.
- [161] F. Badin, B. Jeanneret, R. Trigui, and F. Harel, "Hybrid vehicles, should we plug them to the grid or not?" *Proceedings of the 18th International Electric Vehicle Symposium*, 2004.
- [162] T. Hofman, R. van Druten, A. Serrarens, and J. van Baalen, "A fundamental case study on the prius and ima drivetrain concepts [eb]," *Technische Universiteit Eindhoven*, From Internet: <http://www.wtb.tue.nl>.
- [163] S. Lukic and A. Emadi, "Modeling of electric machines for automotive applications using efficiency maps," *Proceedings of the Electrical Insulation Conference and Electrical Manufacturing & Coil Winding Technology Conference*, pp. 543–550, 2003.
- [164] R. Juvinall and K. Marshek, *Fundamentals of machine component design*. New York: Wiley, 2000.
- [165] M. Dubarry, V. Svoboda, R. Hwu, and B. Liaw, "A roadmap to understand battery performance in electric and hybrid vehicle operation," *Journal of Power Sources*, vol. 174, no. 2, pp. 366–372, 2007.

- [166] M. Schwabacher, "A survey of data-driven prognostics," *Proceedings of Infotech@Aerospace Conference*, 2005.
- [167] P. Ruetschi, "Aging mechanisms and service life of lead–acid batteries," *Journal of Power Sources*, vol. 127, no. 1-2, pp. 33–44, 2004.
- [168] H. Wenzl, I. Baring-Gould, R. Kaiser, B. Liaw, P. Lundsager, J. Manwell, A. Ruddell, and V. Svoboda, "Life prediction of batteries for selecting the technically most suitable and cost effective battery," *Journal of Power Sources*, vol. 144, no. 2, pp. 373–384, 2005.
- [169] D. Pavlov, G. Petkova, M. Dimitrov, M. Shiomi, and M. Tsubota, "Influence of fast charge on the life cycle of positive lead–acid battery plates," *Journal of Power Sources*, vol. 87, no. 1-2, pp. 39–56, 2000.
- [170] M. Verbrugge and E. Tate, "Adaptive state of charge algorithm for nickel metal hydride batteries including hysteresis phenomena," *Journal of Power Sources*, vol. 126, no. 1-2, pp. 236–249, 2004.
- [171] R. Rao, S. Vrudhula, and D. Rakhmatov, "Battery modeling for energy aware system design," *Computer*, vol. 36, no. 12, pp. 77–87, 2003.
- [172] A. Cherif, M. Jraidi, and A. Dhouib, "A battery ageing model used in stand alone pv systems," *Journal of Power Sources*, vol. 112, no. 1, pp. 49–53, 2002.
- [173] O. Barbarisi, R. Canaletti, L. Glielmo, M. Gosso, and F. Vasca, "State of charge estimator for nimh batteries," *Proceedings of the 41st IEEE Conference on Decision and Control*, vol. 2, 2002.
- [174] D. Howell, "Progress report for energy storage research and development," US Department of Energy, Tech. Rep., 2006.
- [175] M. Ivarsson, P. Sahlholm, M. Blackenfelt, K. Johansson, H. Jansson, and L. Nielsen, "Vehicle control using preview information," 2006.
- [176] C. H. Chuang, T. D. Goodson, and L. A. Ledsinger, "Theory and computation of optimal low- and medium-thrust transfers," NASA, Tech. Rep. CR-197317, 1995.
- [177] M. Athans, "The status of optimal control theory and applications for deterministic systems," *IEEE Transactions on Automatic Control*, vol. 11, no. 3, pp. 580–596, 1966.
- [178] J. Pu and C. Yin, "Optimal control of fuel economy in parallel hybrid electric vehicles," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 221, no. 9, pp. 1097–1106, 2007.

- [179] I. Kolmanovsky, M. van Nieuwstadt, and J. Sun, "Optimization of complex powertrain systems for fuel economy and emissions," *Nonlinear Analysis: Real World Applications*, vol. 1, no. 2, pp. 205–221, 2000.
- [180] M. Salman, M. Chang, and J. Chen, "Predictive energy management strategies for hybrid vehicles," *Proceedings of the 2005 IEEE Vehicle Power and Propulsion Conference*, pp. 21–25, 2005.
- [181] S. Jeon, K. Kim, S. Jo, and J. Lee, "Driving simulation of a parallel hybrid electric vehicle using receding horizon control," *Proceedings of the 2001 IEEE International Symposium on Industrial Electronics (ISIE 2001)*, vol. 2, 2001.
- [182] A. Bryson Jr, "Optimal control: 1950 to 1985," *Control Systems Magazine, IEEE*, vol. 16, no. 3, pp. 26–33, 1996.
- [183] C. Lin, H. Peng, J. Grizzle, J. Liu, and M. Busdiecker, "Control system development for an advanced-technology medium-duty hybrid electric truck," *SAE Paper 03TB-45*, 2003.
- [184] M. Canova, K. Sevel, Y. Guezennec, and S. Yurkovich, "Control of the start/stop of a diesel engine in a parallel HEV: Modeling and experiments," *Proceedings of the 2006 ASME International Mechanical Engineering Congress and Exposition*, 2006.
- [185] S. Pagerit, A. Rousseau, and P. Sharer, "Global optimization to real time control of HEV power flow: Example of a fuel cell hybrid vehicle," *Proceedings of the 20th International Electric Vehicle Symposium (EVS20)*, 2005.
- [186] K. Oh, J. Kim, D. Kim, D. Choi, and H. Kim, "Optimal power distribution control for parallel hybrid electric vehicles," *Proceedings of IEEE International Conference on Vehicular Electronics and Safety*, pp. 79–85, 2005.
- [187] J. Kessels, M. Koot, and P. van den Bosch, "Optimal adaptive solution to powersplit problem in vehicles with integrated starter/generator," *Proceedings of the 2006 IEEE Vehicle Power and Propulsion Conference*, 2006.
- [188] M. Koot, J. Kessels, B. de Jager, W. Heemels, P. van den Bosch, and M. Steinbuch, "Energy management strategies for vehicular electric power systems," *IEEE Transactions on Vehicular Technology*, vol. 54, no. 3, pp. 771–782, 2005.
- [189] M. Koot, J. Kessels, and B. de Jager, "Fuel reduction of parallel hybrid electric vehicles," *Proceedings of the 2005 IEEE Vehicle Power and Propulsion Conference*, pp. 26–31, 2005.
- [190] M. P. O'Keefe and K. Vertin, "An analysis of hybrid electric propulsion systems for transit buses," National Renewable Energy Laboratory, Golden, CO, Tech. Rep. NREL/MP-540-32858, 2002.

- [191] L. Johannesson, "On energy management strategies for hybrid electric vehicles," Master's thesis, Chalmers University of Technology, Göteborg, Sweden, 2006.
- [192] J. Chen and M. Salman, "Learning energy management strategy for hybrid electric vehicles," *Vehicle Power and Propulsion, 2005 IEEE Conference*, pp. 427–432, 2005.
- [193] A. Kleimaier and D. Schroder, "An approach for the online optimized control of a hybrid powertrain," *Proceedings of the 7th International Workshop on Advanced Motion Control*, pp. 215–220, 2002.
- [194] S. Williamson and A. Emadi, "Comparative assessment of hybrid electric and fuel cell vehicles based on comprehensive well-to-wheels efficiency analysis," *IEEE Transactions on Vehicular Technology*, vol. 54, no. 3, pp. 856–862, 2005.
- [195] S. Barsali, C. Miulli, and A. Possenti, "A control strategy to minimize fuel consumption of series hybrid electric vehicles," *IEEE Transactions on Energy Conversion*, vol. 19, no. 1, pp. 187–195, 2004.
- [196] S. Delprat, T. Guerra, G. Paganelli, J. Lauber, and M. Delhom, "Control strategy optimization for an hybrid parallel powertrain," *Proceedings of the 2001 American Control Conference*, 2001.
- [197] S. Delprat, T. Guerra, and J. Rimaux, "Optimal control of a parallel powertrain: from global optimization to real time control strategy," *Proceedings of the IEEE 55th Vehicular Technology Conference*, 2002.
- [198] J. Luo, M. Namburu, K. Pattipati, L. Qiao, M. Kawamoto, and S. Chigusa, "Model-based prognostic techniques," *Update*.
- [199] M. T. Todinov, "Necessary and sufficient condition for additivity in the sense of the Palmgren–Miner rule," *Computational Materials Science*, vol. 21, no. 1, pp. 101–110, 2001.
- [200] J. Kessels, "Energy management for automotive power nets," Ph.D. dissertation, Technical University of Eindhoven, 2007.
- [201] E. Stromberg, "Optimal control of hybrid electric vehicles," Master's thesis, Linköping University, 2003.
- [202] T. Sandberg, "Heavy truck modeling for fuel consumption: Simulations and measurements," Master's thesis, Linköping University, 2001.
- [203] S. Delprat, J. Lauber, T. Guerra, and J. Rimaux, "Control of a parallel hybrid powertrain: optimal control," *IEEE Transactions on Vehicular Technology*, vol. 53, no. 3, pp. 872–881, 2004.

- [204] R. Beck, F. Richert, A. Bollig, D. Abel, S. Saenger, K. Neil, T. Scholt, and K. Noreikat, "Model predictive control of a parallel hybrid vehicle drivetrain," *Proceedings of the 44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference (CDC-ECC'05)*, pp. 2670–2675, 2005.
- [205] D. Assanis, W. Bryzik, N. Chalhoub, Z. Filipi, N. Henein, D. Jung, X. Liu, L. Louca, J. Moskwa, S. Munns *et al.*, "Integration and use of diesel engine, driveline and vehicle dynamics models for heavy duty truck simulation," *SAE paper 1999-01-0970*, 1999.
- [206] M. Asbogard, L. Johannesson, D. Angervall, and P. Johansson, "Improving system design of a hybrid powertrain using stochastic drive cycles and dynamic programming," *SAE Paper 2007-01-0304*, 2007.
- [207] P. Pisu, L. Serrao, C. Cantemir, and G. Rizzoni, "Hybrid-electric powertrain design evaluation for future tactical truck vehicle systems," *Proceedings of the 2006 ASME International Mechanical Engineering Congress and Exposition*, 2006.
- [208] D. Assanis, Z. Filipi, S. Gravante, D. Grohnke, X. Gui, L. Louca, G. Rideout, J. Stein, and Y. Wang, "Validation and use of simulink integrated, high fidelity, engine-in-vehicle simulation of the international class vi truck," *SAE Paper 2000-01-0288*, 2000.
- [209] L. Serrao, C. Hubert, and G. Rizzoni, "Dynamic modeling of heavy-duty hybrid electric vehicles," *Proceedings of the 2007 ASME International Mechanical Engineering Congress and Exposition*, 2007.
- [210] C. Lin, S. Jeon, H. Peng, and J. Moo Lee, "Driving pattern recognition for control of hybrid electric trucks," *Vehicle System Dynamics*, vol. 42, no. 1-2, pp. 41–58, 2004.
- [211] L. Johannesson, M. Asbogard, and B. Egardt, "Assessing the potential of predictive control for hybrid vehicle powertrains using stochastic dynamic programming," *Proceedings of the 2005 IEEE Conference on Intelligent Transportation Systems*, pp. 366–371, 2005.
- [212] J. Josephson, B. Chandrasekaran, M. Carroll, N. Iyer, B. Wasacz, G. Rizzoni, Q. Li, and D. Erb, "An architecture for exploring large design spaces," *Proc. of the National Conference of Artificial Intelligence (AAAI-98)*, Madison, Wisconsin, pp. 143–150, 1998.
- [213] R. Bellman, *Dynamic Programming*. New York: Courier Dover Publications, 2003.
- [214] F. Badin, O. Briat, S. Olivier, B. Jeanneret, R. Trigui, and B. Malaquin, "The use of batteries in hybrid vehicles," *IEEE International Symposium on Vehicular Power and Propulsion, Paris, France*, 2004.

- [215] D. Vladikova, "The technique of the differential impedance analysis. Part I: Basics of the impedance spectroscopy," in *Proceedings of the International Workshop "Advanced Techniques for Energy Sources Investigation and Testing"*, 2004, pp. 4–9.
- [216] V. Johnson, A. Pesaran, and T. Sack, "Temperature-Dependent Battery Models for High-Power Lithium-Ion Batteries," *17th Annual Electric Vehicle Symposium, Montreal, Quebec (CA), 10/15/2000–10/18/2000*, 2001.
- [217] J. R. Miller and S. M. Butler, "Reliability of high-voltage electrochemical capacitors: predictions for statistically independent cells from a single distribution," in *12th International Seminar on Double Layer Capacitors and Similar Energy Storage Devices*, 2002.
- [218] D. Linzen, S. Buller, E. Karden, and R. De Doncker, "Analysis and evaluation of charge-balancing circuits on performance, reliability, and lifetime of supercapacitor systems," *IEEE Transactions on Industry Applications*, vol. 41, no. 5, pp. 1135–1141, 2005.
- [219] S. Eom, M. Kim, I. Kim, S. Moon, Y. Sun, and H. Kim, "Life prediction and reliability assessment of lithium secondary batteries," *Journal of Power Sources*, vol. 174, no. 2, pp. 954–958, 2007.
- [220] J. R. Miller and S. M. Butler, "Electrochemical capacitor float-voltage operation: Leakage current influence on cell voltage uniformity," in *11th International Seminar on Double Layer Capacitors and Similar Energy Storage Devices*, 2001.
- [221] J. Francfort, D. Karner, R. Harkins, and J. Tardiolo, "Hybrid Electric Vehicle Fleet and Baseline Performance Testing," 2006.
- [222] R. Kostecky, X. Zhang, P. Ross Jr, F. Kong, S. Sloop, J. Kerr, K. Striebel, E. Cairns, and F. McLarnon, "Failure modes in high-power lithium-ion batteries for use in hybrid electric vehicles," LBNL–48359, Ernest Orlando Lawrence Berkeley National Laboratory, Berkeley, CA (US), Tech. Rep., 2001.
- [223] T. Markel, M. Zolot, K. Wipke, A. Pesaran, and F. Nice, "Energy Storage System Requirements for Hybrid Fuel Cell Vehicles," *Proceedings of the 4th International Advanced Automotive Battery Conference. San Francisco, CA*, 2003.
- [224] M. Verbrugge and R. Conell, "Electrochemical and Thermal Characterization of Battery Modules Commensurate with Electric Vehicle Integration," *Journal of The Electrochemical Society*, vol. 149, p. A45, 2002.
- [225] M. Zolot, "Dual-source energy storage control and performance advantages in advanced vehicles," *Proceedings of the 20th Electric Vehicle Symposium (EVS-20)*, 2003.

- [226] F. Belhachemi, S. Rael, and B. Davat, "A physical based model of power electric double-layer supercapacitors," *IEEE 2000 Industry Applications Conference*, vol. 5, 2000.
- [227] K. Kelly, M. Mihalic, and M. Zolot, "Battery usage and thermal performance of the Toyota Prius and HondaInsight during chassis dynamometer testing," *Battery Conference on Applications and Advances, 2002. The Seventeenth Annual*, pp. 247–252, 2002.
- [228] D. Panigrahi, C. Chiasserini, S. Dey, R. Rao, A. Raghunathan, and K. Lahiri, "Battery life estimation of mobile embedded systems," *Proc. Int. Conf. VLSI Design*, pp. 55–63, 2001.
- [229] P. Rong and M. Pedram, "An analytical model for predicting the remaining battery capacity of lithium-ion batteries," *Proc. 2003 Design, Automation and Test in Europe Conf. and Exposition*, pp. 1148–1149.
- [230] J. Vetter, P. Novák, M. Wagner, C. Veit, K. Möller, J. Besenhard, M. Winter, M. Wohlfahrt-Mehrens, C. Vogler, and A. Hammouche, "Ageing mechanisms in lithium-ion batteries," *Journal of Power Sources*, vol. 147, no. 1-2, pp. 269–281, 2005.
- [231] Y. Kobayashi, "Acceleration coefficient of the molded electric double layer capacitor," *Electronic Components and Technology Conference, 1992. Proceedings., 42nd*, pp. 617–623, 1992.
- [232] P. Sendur, J. L. Stein, H. Peng, and L. Louca, "An algorithm for the assessment of reduced dynamic system models for design," in *International Conference on Simulation and Multimedia in Engineering Education*. Orlando, FL: Society for Modeling and Simulation, 2003.
- [233] —, "An algorithm for the selection of physical system model order based on desired state accuracy and computational efficiency," *Proceedings of the 2003 ASME International Mechanical Engineering Congress and Exposition*, 2003.
- [234] L. Serrao, P. Pisu, and G. Rizzoni, "Analysis and evaluation of a two engine configuration in a series hybrid electric vehicle," *Proceedings of the 2006 ASME International Mechanical Engineering Congress and Exposition*, 2006.
- [235] N. Dembski, Y. Guezennec, and A. Soliman, "Analysis and experimental refinement of real-world driving cycles," *SAE paper 2002-01-0069*, 2002.
- [236] J. B. Ferris, J. L. Stein, and M. M. Bernitsas, "Development of proper models of hybrid systems," *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 120, no. 3, pp. 328–333, 1998.

- [237] P. Sendur, J. L. Stein, H. Peng, and L. Louca, "A model accuracy and validation algorithm," *Proceedings of the 2002 ASME International Mechanical Engineering Congress and Exposition*, 2002.
- [238] P. Pisu, C. J. Hubert, N. Dembski, G. Rizzoni, J. R. Josephson, J. Russell, and M. Carroll, "Modeling and design of heavy duty hybrid electric vehicles," *Proceedings of the 2005 ASME International Mechanical Engineering Congress and Exposition*, 2005.
- [239] L. Louca, J. L. Stein, G. M. Hulbert, and J. Spragues, "Proper model generation: An energy-based methodology," in *1997 International Conference on Bond Graph Modeling*, Phoenix, AZ (USA), 1997.
- [240] T. Miller, G. Rizzoni, and Q. Li, "Simulation-based hybrid-electric vehicle design search," *SAE paper 1999-01-1150*, 1999.
- [241] C. C. Chan, "The state of the art of electric and hybrid vehicles," *Proceedings of the IEEE*, vol. 90, no. 2, pp. 245–275, 2002.